Oracle® Communications

Cloud Native Core, Network Repository Function User Guide

Release 25.1.201
G33631-04
October 2025

Oracle Communications Cloud Native Core, Network Repository Function User Guide, Release 25.1.201

G33631-04

Copyright © 2019, 2025, Oracle and/or its affiliates.

This software and related documentation are provided under a license agreement containing restrictions on use and disclosure and are protected by intellectual property laws. Except as expressly permitted in your license agreement or allowed by law, you may not use, copy, reproduce, translate, broadcast, modify, license, transmit, distribute, exhibit, perform, publish, or display any part, in any form, or by any means. Reverse engineering, disassembly, or decompilation of this software, unless required by law for interoperability, is prohibited.

The information contained herein is subject to change without notice and is not warranted to be error-free. If you find any errors, please report them to us in writing.

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

U.S. GOVERNMENT END USERS: Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs) and Oracle computer documentation or other Oracle data delivered to or accessed by U.S. Government end users are "commercial computer software," "commercial computer software documentation," or "limited rights data" pursuant to the applicable Federal Acquisition Regulation and agency-specific supplemental regulations. As such, the use, reproduction, duplication, release, display, disclosure, modification, preparation of derivative works, and/or adaptation of i) Oracle programs (including any operating system, integrated software, any programs embedded, installed, or activated on delivered hardware, and modifications of such programs), ii) Oracle computer documentation and/or iii) other Oracle data, is subject to the rights and limitations specified in the license contained in the applicable contract. The terms governing the U.S. Government's use of Oracle cloud services are defined by the applicable contract for such services. No other rights are granted to the U.S. Government.

This software or hardware is developed for general use in a variety of information management applications. It is not developed or intended for use in any inherently dangerous applications, including applications that may create a risk of personal injury. If you use this software or hardware in dangerous applications, then you shall be responsible to take all appropriate fail-safe, backup, redundancy, and other measures to ensure its safe use. Oracle Corporation and its affiliates disclaim any liability for any damages caused by use of this software or hardware in dangerous applications.

Oracle®, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.

Intel and Intel Inside are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Epyc, and the AMD logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group.

This software or hardware and documentation may provide access to or information about content, products, and services from third parties. Oracle Corporation and its affiliates are not responsible for and expressly disclaim all warranties of any kind with respect to third-party content, products, and services unless otherwise set forth in an applicable agreement between you and Oracle. Oracle Corporation and its affiliates will not be responsible for any loss, costs, or damages incurred due to your access to or use of third-party content, products, or services, except as set forth in an applicable agreement between you and Oracle.

Contents

1 Introduction

1.1 References 3

2 NRF Supported Services

2.1 Supported NF Types 2

3 NRF Architecture

4 NRF Features

4.1 Pod Protection Using Rate Limiting 1
4.1.1 Egress Gateway Pod Protection Using Rate Limiting 1
4.1.2 Ingress Gateway Pod Protection Using Rate Limiting 9
4.2 NF Profile Size Limit 14
4.3 Egress Gateway Pod Throttling 17
4.4 Traffic Segregation 19
4.5 Support for Server Header 23
4.6 Support for TLS 26
4.7 Error Messages Enhancement 32
4.7.1 Error Response Enhancement for NRF 32
4.7.2 Error Log Messages Enhancement 38
4.8 Limiting Number of NFProfiles in NFDiscover Response 45
4.9 Support for cnDBTier APIs in CNC Console 46
4.10 Enhanced NRF Set Based Deployment (NRF Growth) 49
4.10.1 Impacted Service Operations 59
4.10.2 Interaction with Existing Features 63
4.11 Support for Automated Certificate Lifecycle Management 64
4.12 Egress Gateway Route Configuration for Different Deployments 66
4.13 Routing Egress Messages through SCP 71
4.13.1 SLF Requests 74
4.13.2 Notifications Requests 76
4.13.3 NRF Forwarding Requests 78

Copyright © 2019, 2025, Oracle and/or its affiliates.

4.13.4 Roaming Requests 81
4.14 Ingress Gateway Pod Protection 84
4.15 Network Slice Specific Metrics 88
4.16 CCA Header Validation in NRF for Access Token Service Operation 90
4.17 Monitoring the Availability of SCP Using SCP Health APIs 93
4.18 Controlled Shutdown of NRF 96
4.19 User-Agent Header for Outgoing Requests 101
4.20 Support for Kubernetes Resource 101
4.20.1 Node Selector 101
4.20.2 Kubernetes Probes 103
4.20.3 Pod Disruption Budget 104
4.20.4 Network Policies 104
4.20.5 Tolerations 105
4.20.6 Dual Stack 106
4.21 Pod Protection Support for NRF Subscription Microservice 110
4.22 Pre and Post Install/Upgrade Validations 115
4.22.1 NRF Infrastructure Validation 118
4.22.2 NRF Preinstallation Validation 119
4.22.2.1 For Fresh Installation 119
4.22.2.2 For Fault Recovery 119
4.22.3 NRF Postinstallation Validations 120
4.22.3.1 For Fresh Installation 120
4.22.3.2 For Fault Recovery 120
4.22.4 NRF Preupgrade Validation 120
4.22.5 NRF Postupgrade Validation 121
4.23 Ignore Unknown Attribute in NFDiscover Search Query 122
4.24 NF Authentication using TLS Certificate 122
4.24.1 XFCC Header Validation 122
4.24.2 TLS SNI Header Validation 124
4.25 Subscriber Location Function 125
4.25.1 Static Selection of SLF 128
4.25.2 Dynamic Selection of SLF 129
4.25.3 Support for Configurable Port and Routing Parameter Selection 132
4.25.4 Rerouting SLF Requests Using Alternate SCP and Alternate SLF 133
4.25.5 Discovery Parameter Value Based Skip SLF Lookup 137
4.26 NRF Forwarding 141
4.27 NRF Georedundancy 145
4.27.1 NRF Last Known Response 150
4.28 NF Heartbeat Enhancement 154
4.29 Service mesh for intra-NF communication 155
4.30 NF Screening 155
4.31 REST Based NRF State Data Retrieval 158

Copyright © 2019, 2025, Oracle and/or its affiliates.

4.32 Key-ID for Access Token 158
4.33 NRF Access Token Service Usage Details 160
4.34 Access Token Request Authorization 162
4.35 Preferred Locality Feature Set 164
4.35.1 Preferred Locality 165
4.35.2 Extended Preferred Locality 165
4.35.2.1 Discovery Processing for Extended Preferred Locality Feature 166
4.35.2.2 Configuring Extended Preferred Locality 167
4.35.2.3 Limiting the Number of Producers Based on NF Set Ids 170
4.35.3 NFService Priority Update 173
4.35.3.1 Discovery Processing in NFService Priority Update with preferred-locality 175
4.35.3.2 Configuring NFService Priority Update 176
4.36 Roaming Support 176
4.37 EmptyList in Discovery Response 181
4.38 Overload Control 186
4.39 DNS NAPTR Update 191
4.40 Notification Retry 193
4.41 NRF Message Feed 196
4.42 Subscription Limit 210
4.43 Automated Test Suite Support 212

5 Configuring NRF using CNC Console

5.1 Support for Multiple Cluster Deployment 1
5.2 CNC Console Interface 2
5.3 NRF Configuration 3
5.3.1 cnDBTier APIs 4
5.3.1.1 cnDBTier Backup Status 4
5.3.1.2 cnDBTier Health 4
5.3.1.3 cnDBTier Version 7
5.3.1.4 Backup List 7
5.3.1.5 Database Statistics Report 8
5.3.1.6 Georeplication Status 8
5.3.1.7 Georeplication Recovery 10
5.3.1.8 Replication HeartBeat Status 11
5.3.1.9 Local Cluster Status 11
5.3.1.10 On Demand Backup 12
5.3.2 General Options 13
5.3.3 NF Management Options 15
5.3.4 NF Discovery Options 19
5.3.5 NF Access Token Options 23
5.3.5.1 Access Token Signing Data Status 25

Copyright © 2019, 2025, Oracle and/or its affiliates.

5.3.6 Forwarding Options 25
5.3.7 SLF Options 27
5.3.8 Roaming Options 33
5.3.9 Georedundancy Options 34
5.3.10 NF Authentication Options 35
5.3.11 Logging Level Options 37
5.3.12 NF Screening Options 40
5.3.12.1 Screening Rules 40
5.3.13 DNS NAPTR Update Options 44
5.3.13.1 DNS NAPTR Record Status 45
5.3.13.2 DNS NAPTR Record Retrigger 45
5.3.14 Pod Protection Options 46
5.3.15 Controlled Shutdown 49
5.3.16 NRF Growth Options 50
5.3.16.1 Forwarding Options 50
5.3.17 Ingress Gateway 51
5.3.17.1 CCA Header 51
5.3.17.2 Controlled Shutdown Error Mapping 53
5.3.17.3 Copy Header on Gateway Error 54
5.3.17.4 Error Code Profile 54
5.3.17.5 Overload Control Policy Mapping 56
5.3.17.6 Overload Control Discard Policy 58
5.3.17.7 Error Code Series 59
5.3.17.8 Routes Configuration 60
5.3.17.9 Pod Protection 61
5.3.17.10 Server Header Details 65
5.3.17.11 Congestion Level Configuration 66
5.3.17.12 Pod Protection By Rate Limiting 68
5.3.18 Egress Gateway 70
5.3.18.1 Peer Configuration 71
5.3.18.2 Peer Set Configuration 72
5.3.18.3 Peer Monitoring Configuration 73
5.3.18.4 Routes Configuration 74
5.3.18.5 SBI Error Action Sets 77
5.3.18.6 SBI Error Criteria Sets 78
5.3.18.7 Pod Protection By Rate Limiting 79
5.3.18.8 Error Code Profile 81
5.3.18.9 Congestion Level Configuration 83
5.3.19 Alternate Route Service 84
5.3.19.1 Upstream Dns Config 85

Copyright © 2019, 2025, Oracle and/or its affiliates.

6 NRF Metrics, KPIs, and Alerts

6.1 NRF Metrics 1
6.1.1 NRF Gateways Metrics 13
6.1.1.1 Ingress Gateway Metrics 13
6.1.1.2 Egress Gateway Metrics 21
6.1.1.3 Common Gateway Metrics 28
6.1.2 Platform Metrics 30
6.1.3 Configuring NRF Metrics Dashboard in OCI 31
6.1.4 NRF NF Metrics 31
6.1.5 NF Screening Metrics 42
6.1.6 NF Access token Metrics 44
6.1.7 NRF Configuration Metrics 45
6.1.8 NRF SLF Metrics 46
6.1.9 NRF Forwarding Metrics 48
6.1.10 Roaming Support Metrics 52
6.1.11 Georedundancy Metrics 58
6.1.12 NF Authentication Metrics 60
6.1.13 NF Instance and NF Services Status Change Metrics 60
6.1.14 DNS NAPTR Update Metrics 61
6.1.15 Pod Protection Metrics 65
6.1.16 Network Slice Specific Metrics 67
6.1.17 NRF Cache Data Metrics 80
6.1.18 Overload Control 84
6.2 NRF KPIs 85
6.2.1 NRF Service KPIs 85
6.2.2 Resource Usage KPIs 111
6.2.3 Feature Specific KPIs 115
6.2.3.1 Global Subscription Limit Feature 115
6.2.3.2 NF FQDN Validation KPIs 116
6.2.3.3 Georedundancy KPIs 117
6.2.3.4 Subscription Pod Protection 118
6.2.3.5 Controlled Shutdown of NRF 119
6.2.3.6 Network Slice Specific KPIs 119
6.2.3.7 SCP Health Status 125
6.2.3.8 Ingress Gateway Pod Protection 126
6.2.3.9 NRF Growth Specific KPIs 127
6.2.3.10 Egress Gateway Pod Throttling 132
6.2.3.11 Ingress Gateway Pod Protection Using Rate Limiting 132
6.2.3.12 Egress Gateway Pod Protection Using Rate Limiting 133
6.3 NRF Alerts 134
6.3.1 System Level Alerts 135

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.1.1 OcnrfNfStatusUnavailable 135
6.3.1.2 OcnrfPodsRestart 136
6.3.1.3 NnrfNFManagementServiceDown 137
6.3.1.4 NnrfAccessTokenServiceDown 140
6.3.1.5 NnrfNFDiscoveryServiceDown 141
6.3.1.6 OcnrfRegistrationServiceDown 142
6.3.1.7 OcnrfSubscriptionServiceDown 143
6.3.1.8 OcnrfDiscoveryServiceDown 144
6.3.1.9 OcnrfAccessTokenServiceDown 145
6.3.1.10 OcnrfAuditorServiceDown 146
6.3.1.11 OcnrfConfigurationServiceDown 147
6.3.1.12 OcnrfAppInfoServiceDown 148
6.3.1.13 OcnrfArtisanServiceDown 149
6.3.1.14 OcnrfAlternateRouteServiceDown 150
6.3.1.15 OcnrfPerfInfoServiceDown 151
6.3.1.16 OcnrfIngressGatewayServiceDown 152
6.3.1.17 OcnrfEgressGatewayServiceDown 153
6.3.1.18 OcnrfTotalIngressTrafficRateAboveMinorThreshold 154
6.3.1.19 OcnrfTotalIngressTrafficRateAboveMajorThreshold 155
6.3.1.20 OcnrfTotalIngressTrafficRateAboveCriticalThreshold 156
6.3.1.21 OcnrfTransactionErrorRateAbove0Dot1Percent 157
6.3.1.22 OcnrfTransactionErrorRateAbove1Percent 157
6.3.1.23 OcnrfTransactionErrorRateAbove10Percent 158
6.3.1.24 OcnrfTransactionErrorRateAbove25Percent 159
6.3.1.25 OcnrfTransactionErrorRateAbove50Percent 160
6.3.1.26 OcnrfTotalEgressTrafficRateAboveCriticalThreshold 161
6.3.1.27 OcnrfTotalForwardingTrafficRateAboveCriticalThreshold 162
6.3.1.28 OcnrfHeapUsageCrossedMinorThreshold 162
6.3.1.29 OcnrfHeapUsageCrossedMajorThreshold 163
6.3.1.30 OcnrfHeapUsageCrossedCriticalThreshold 164
6.3.2 Service Level Alerts 164
6.3.2.1 OcnrfAccessTokenRequestsRejected 164
6.3.2.2 OcnrfAuditorMultiplePodUnavailable 165
6.3.2.3 OcnrfAppInfoMultiplePodUnavailable 165
6.3.2.4 OcnrfPerfInfoMultiplePodUnavailable 166
6.3.2.5 OcnrfTotalSLFRateAboveCriticalThreshold 166
6.3.2.6 OcnrfTotalDiscoveryRateAboveCriticalThreshold 167
6.3.2.7 OcnrfAccessTokenRequestsAboveThreshold 167
6.3.2.8 OcnrfNfUpdateRequestsAboveThreshold 168
6.3.2.9 OcnrfRegisteredNfCountAboveThreshold 168
6.3.2.10 OcnrfNfHeartBeatRequestsAboveThreshold 169
6.3.2.11 OcnrfRegisteredNfCountAboveThreshold 170

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.2.12 OcnrfTotalSubscriptionsAboveThreshold 170
6.3.2.13 OcnrfDiscoveryRequestsForUDRAboveThreshold 171
6.3.2.14 OcnrfDiscoveryRequestsForUDMAboveThreshold 171
6.3.2.15 OcnrfDiscoveryRequestsForAMFAboveThreshold 172
6.3.2.16 OcnrfDiscoveryRequestsForSMFAboveThreshold 172
6.3.3 NfProfile Status Change Alerts 173
6.3.3.1 OcnrfRegisteredPCFsBelowCriticalThreshold 173
6.3.3.2 OcnrfRegisteredPCFsBelowMajorThreshold 174
6.3.3.3 OcnrfRegisteredPCFsBelowMinorThreshold 175
6.3.3.4 OcnrfRegisteredPCFsBelowThreshold 176
6.3.3.5 OcnrfTotalNFsRegisteredBelowCriticalThreshold 178
6.3.3.6 OcnrfTotalNFsRegisteredBelowMajorThreshold 178
6.3.3.7 OcnrfTotalNFsRegisteredBelowMinorThreshold 179
6.3.3.8 OcnrfTotalNFsRegisteredApproachingMinorThreshold 180
6.3.3.9 OcnrfNFStatusTransitionToRegistered 181
6.3.3.10 OcnrfNFServiceStatusTransitionToRegistered 181
6.3.3.11 OcnrfNFStatusTransitionToSuspended 182
6.3.3.12 OcnrfNFServiceStatusTransitionToSuspended 183
6.3.3.13 OcnrfNFStatusTransitionToUndiscoverable 184
6.3.3.14 OcnrfNFServiceStatusTransitionToUndiscoverable 185
6.3.3.15 OcnrfNFStatusTransitionToDeregistered 186
6.3.3.16 OcnrfNFServiceStatusTransitionToDeregistered 187
6.3.3.17 OcnrfNfProfileSizeAboveThreshold 188
6.3.3.18 OcnrfDiscoveryResponseSizeAboveThreshold 188
6.3.4 Feature Specific Alerts 189
6.3.4.1 KeyID for AccessToken Feature 189
6.3.4.2 Overload Control Based on Percentage Discards Feature 194
6.3.4.3 DNS NAPTR Update Feature 199
6.3.4.4 Notification Retry Feature 200
6.3.4.5 NRF Message Feed Feature 201
6.3.4.6 Subscription Limit Feature 202
6.3.4.7 Pod Protection Support for NRF Subscription Microservice 205
6.3.4.8 Controlled Shutdown of NRF Feature 211
6.3.4.9 Monitoring the Availability of SCP Using SCP Health APIs Feature 212
6.3.4.10 CCA Header Validation in NRF for Access Token Service Operation Feature 213
6.3.4.11 NRF Georedundancy Feature 215
6.3.4.12 XFCC Header Validation Feature 216
6.3.4.13 Enhanced NRF Set Based Deployment (NRF Growth) Feature 216
6.3.4.14 Ingress Gateway Pod Protection Feature 223
6.3.4.15 Subscriber Location Function Feature 227
6.3.4.16 EmptyList in Discovery Response Feature 228

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.17 Support for TLS 229
6.3.4.18 Egress Gateway Pod Throttling 231
6.3.4.19 Ingress Gateway Pod Protection Using Rate Limiting 232
6.3.4.20 Egress Gateway Pod Protection Using Rate Limiting 233
6.3.5 NRF Alert Configuration 235
6.3.5.1 Disable Alerts 238
6.3.5.2 Configuring SNMP Notifier 239
6.3.6 NRF Alert Configuration in OCI 240

A HTTP Status Code Details

A.1 HTTP Status Code details A-1

B Handling 3GPP Attributes

B.1 vsmf-support-ind B-1
B.2 nrfSupportedFeatures B-1
B.3 DataSetId Enhancements B-5
B.4 Support for servingClientTypes in LmfInfo and GmlcInfo B-6
B.5 Enhancements for dnn NFProfile Attribute and Discovery Query Parameter B-9
B.6 Content-Encoding Header B-11

C Error Response Code Details

C.1 NF Registration Microservice C-1
C.1.1 NFUpdate with Full Replacement (PUT) C-2
C.1.2 NFUpdate with Partial Replacement (PATCH) C-4
C.1.3 NFHeartbeat Service Operation C-5
C.1.4 NFDeregister Service Operation C-6
C.1.5 NFProfileRetrieval Service Operation C-7
C.1.6 NFListRetrieval Service Operation C-8
C.1.7 NFProfileRetrieval Service Operation During Forwarding C-9
C.1.8 UnSupported Media Type C-9
C.2 NF Subscription Microservice C-10
C.2.1 NFStatusSubscribe Service Operation C-10
C.2.1.1 NFStatusSubscribe Update (PATCH) Service Operation C-11
C.2.1.2 NFStatusSubscribe Service Operation During Forwarding C-12
C.2.1.3 NFStatusSubscribe Update (PATCH) Service Operation During Forwarding C-12
C.2.1.4 NFStatusSubscribe Update (PATCH) Service Operation During Roaming C-13
C.2.2 NFStatusUnsubscribe Service Operation C-14
C.2.2.1 NFStatusUnsubscribe Service Operation During Forwarding C-14
C.2.3 NFStatusSubscribe Service Operation During Roaming C-15

Copyright © 2019, 2025, Oracle and/or its affiliates.

C.2.4 NFStatusUnsubscribe Service Operation During Roaming C-16
C.2.5 NFStatusNotify Service Operation C-16
C.2.6 NFStatus UnSupported HTTP Method for Service Operation C-17
C.2.7 NFStatus UnSupported Media Type C-17
C.3 NF Discovery Microservice C-17
C.3.1 NFDiscover Service Operation C-17
C.3.2 NFDiscover Service Operation During Authorization C-18
C.3.3 NFDiscover Service Operation During Forwarding C-18
C.3.4 SLF for NFDiscover C-19
C.3.5 NFDiscover Service Operation During Roaming C-20
C.3.6 Unsupported HTTP Method for NFDiscover Service Operation C-21
C.3.7 Unsupported Media Type for NFDiscover Service Operation C-21
C.4 NF Access Token Microservice C-21
C.4.1 NF Access Token Microservice C-21
C.4.2 Access Token Service Operation During Forwarding C-22
C.4.3 Access Token Service Operation During Roaming C-23
C.4.4 Authentication During Access Token Service Operation C-24
C.4.5 Authorization During Access Token Service Operation C-25
C.5 NRF Cache Data Microservice C-25
C.5.1 GET Subscription C-25
C.5.2 GET Remote Discovery Instance and Remote Instance Data C-26
C.5.3 NFListRetrieval Service Operation C-26
C.5.4 NFProfileRetrieval Service Operation C-26
C.6 NRF Auditor Microservice C-26
C.6.1 Heart Beat for NRF Auditor Service Operation C-27
C.6.2 Remote Heart Beat for NRF Auditor C-27
C.6.3 NRF Auditor Service Operation for NF Profiles C-27
C.6.4 Fetching NRF Configuration C-28
C.6.5 NRF Auditor Service Operation for Reloading Site and Network Version Info C-28
C.6.6 NRF Auditor Service Operation for Remote Subscription C-28
C.6.7 NRF Auditor Service Operation for Subscription C-28
C.7 NRF Artisan Microservice C-29
C.7.1 NRF Artisan for NAPTR Registration C-29
C.7.2 Status API for DNS NAPTR Record C-29
C.7.3 NRF Artisan for Notification Event to DNS NAPTR on Deleting AMF Profile C-29
C.7.4 NRF Artisan for DNS NAPTR Update C-30
C.7.5 Manual Cache Update of DNS NAPTR Records C-30
C.7.6 UnSupported Media Type C-30
C.8 NRF Configuration Microservice C-31
C.8.1 NRF Configuration for General Options C-31
C.8.2 Screening Options Configuration C-32
C.8.3 Management Options Configuration C-32

Copyright © 2019, 2025, Oracle and/or its affiliates.

C.8.4 Discovery Options Configuration C-33
C.8.5 Access Token Configuration C-34
C.8.6 Forwarding Options Configuration C-35
C.8.7 SLF Options Configuration C-36
C.8.8 GeoRedundancy Options Configuration C-37
C.8.9 NAPTR Options Configuration C-38
C.8.10 Pod Protection Options Configuration C-39
C.8.11 Authentication Options Configuration C-40
C.8.12 Logging Options Configuration C-41
C.8.13 Control Shutdown Options Configuration C-42
C.8.14 Roaming Options Configuration C-43
C.8.15 Screening Rules Configuration C-44
C.8.16 Growth Options Configuration C-45
C.8.17 Growth Configuration for Forwarding C-45
C.8.18 Fetching NRF Configuration C-46
C.8.19 Fetching NF State Data C-46
C.8.20 Retrieve Subscription Details Based on Query Attributes for Configuring the Non-Signaling API C-47
C.8.21 Status API for DNS NAPTR Record C-47
C.8.22 Manual Cache Update of DNS NAPTR Records C-47
C.8.23 NRF Operation State Configuration C-48
C.8.24 CNCC Controller Configuration C-49
C.8.25 UnSupported Media Type C-49
C.9 Ingress Gateway Service C-49
C.10 Egress Gateway Service C-51
C.11 Alternate Route Service C-52
C.12 Common Configuration C-53

D Server Header Propagation

D.1 Adding Filters for Server Header Propagation D-1

E Use Cases for Allow Duplicate Subscriptions

F NRF Recommended Features

Copyright © 2019, 2025, Oracle and/or its affiliates.

Preface

- Documentation Accessibility
- Diversity and Inclusion
- Conventions

Documentation Accessibility

For information about Oracle's commitment to accessibility, visit the Oracle Accessibility Program website at http://www.oracle.com/pls/topic/lookup?ctx=acc&id=docacc.

Access to Oracle Support

Oracle customer access to and use of Oracle support services will be pursuant to the terms and conditions specified in their Oracle order for the applicable services.

Diversity and Inclusion

Oracle is fully committed to diversity and inclusion. Oracle respects and values having a diverse workforce that increases thought leadership and innovation. As part of our initiative to build a more inclusive culture that positively impacts our employees, customers, and partners, we are working to remove insensitive terms from our products and documentation. We are also mindful of the necessity to maintain compatibility with our customers' existing technologies and the need to ensure continuity of service as Oracle's offerings and industry standards evolve. Because of these technical constraints, our effort to remove insensitive terms is ongoing and will take time and external cooperation.

Conventions

The following text conventions are used in this document:

Convention	Meaning
boldface	Boldface type indicates graphical user interface elements associated with an action, or terms defined in text or the glossary.
italic	Italic type indicates book titles, emphasis, or placeholder variables for which you supply particular values.
`monospace`	Monospace type indicates commands within a paragraph, URLs, code in examples, text that appears on the screen, or text that you enter.

Copyright © 2019, 2025, Oracle and/or its affiliates.

My Oracle Support

My Oracle Support (https://support.oracle.com) is your initial point of contact for all product support and training needs. A representative at Customer Access Support can assist you with My Oracle Support registration.

Call the Customer Access Support main number at 1-800-223-1711 (toll-free in the US), or call the Oracle Support hotline for your local country from the list at http://www.oracle.com/us/support/contact/index.html. When calling, make the selections in the sequence shown below on the Support telephone menu:

- For Technical issues such as creating a new Service Request (SR), select **1**.
- For Non-technical issues such as registration or assistance with My Oracle Support, select **2**.
- For Hardware, Networking and Solaris Operating System Support, select **3**.

You are connected to a live agent who can assist you with My Oracle Support registration and opening a support ticket.

My Oracle Support is available 24 hours a day, 7 days a week, 365 days a year.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Acronyms

The following table lists the acronyms and the terminologies used in the document:

Table Acronyms and Terminologies

Acronym	Definition
3GPP	3rd Generation Partnership Project
AF	Application Function
AMF	Access and Mobility Management Function
ASM	Aspen Service Mesh
BSF	Oracle Communications Cloud Native Core, Binding Support Function
CA	Certificate Authority
CCA	Client Credentials Assertion
CDS	Cache data Service. Caching Microservice which is responsible for syncing and caching the data
CHF	Charging Function
CM	Configuration Management
CMPv2	Certificate Management Protocol Version 2
CNC	Cloud Native Core
CNC Console	Oracle Communications Cloud Native Configuration Console
CNE	Oracle Communication Cloud Native Core, Cloud Native Environment
CNI	Multus Container Network Interface
CNLB	Cloud Native Load Balancer
CNPCRF	Oracle Communications Cloud Native Core, Policy and Charging Rules Function
CUSTOMER_REPO	Docker registry address including the port number, if the docker registry has an associated port.
cnDBTier	Oracle Communications Cloud Native Core, cnDBTier
GUAMI	Globally Unique AMF Identifier
HTTPS	Hypertext Transfer Protocol Secure
IMAGE_TAG	Image tag from release tar file. You can use any tag number. However, make sure that you use that specific tag number while pushing docker image to the docker registry.
IMS	IP Multimedia Subsystem
JOSE	Javascript Object Signing and Encryption
JSON	JavaScript Object Notation
JWT	JSON Web Token
LBVM	Load Balancing of Virtual Machine
LCM	Life Cycle Management
MCC	Mobile Country Code
MCPTT	Mission-critical push-to-talk
METALLB_ADDRESS_POOL	Address pool configured on metallb to provide external IPs
MNC	Mobile Network Code
NAD	Network Attachment Definitions
NEF	Oracle Communications Cloud Native Core, Network Exposure Function

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table (Cont.) Acronyms and Terminologies

Acronym	Definition
NF	Network Function
NPLI	Network Provided Location Information
NRF	Oracle Communications Cloud Native Core, Network Repository Function
NSI	Network Slice Instances
NWDAF	Oracle Communications Networks Data Analytics Function
OAM	Operations Administration and Maintenance
OCI	Oracle Cloud Infrastructure
OCCM	Oracle Communications Cloud Native Core, Certificate Management
OCNADD	Oracle Communications Networks Analytics Data Director
PDB	Pod Disruption Budget
PCF	Oracle Communications Cloud Native Core, Policy Control Function
PCRF	Oracle Communications Cloud Native Core, Policy and Charging Rules Function
PCEF	Policy and Charging Enforcement Function
PDS	Policy Data Service
PRA	Presence Reporting Area
PRE	Policy Runtime Engine
PDU	Protocol Data Unit
PVC	Persistent Volume Chain
Policy	Oracle Communications Cloud Native Core, Converged Policy
QoS	Quality of Service
RAN	Radio Access Network
SAN	Subject Alternate Name
SBI	Service Based Interface
SNSSAI	Single Network Slice Selection Assistance Information
SMF	Session Management Function
SNAT	Source Network Address Translation
SRA	Successful Resource Allocation
STR	Session Termination Request (Rx)
UDR	Oracle Communications Cloud Native Core, Unified Data Repository
UE	User Equipment
UPF	User Plane Function
UPSI	UE Policy Section Identifier
URSP	UE Route Selection Policies
Local NRF set state data	NF instances and NF subscription data that is stored in the local cnDBTier of an NRF instance. It is the state data from the local cnDBTier and replicated state data from the georedundant NRF.
Remote NRF set state data	NF registrations and NF subscriptions data that is stored in remote NRF set instances within the same segment. It is the state data from peer NRF set in the same segment
Periodic syncing	It refreshes the local cache periodically.
State Data	The state data is the dynamic data that contains the NFInstances and NFSubscriptions of the NFs.

Copyright © 2019, 2025, Oracle and/or its affiliates.

What's New in This Guide

This section lists the documentation updates for release 25.1.2xx.

Release 25.1.201-G33631-04, October 2025

Feature Updates

- **New Features:**
  - **Egress Gateway Pod Protection By Rate Limiting**
    - Added the Egress Gateway Pod Protection Using Rate Limiting section to describe the functionality of the feature.
    - Added the following pages in the CNC Console section:
      - Pod Protection By Rate Limiting
      - Error Code Profile
      - Congestion Level Configuration
    - Added the following metrics in the Egress Gateway Metrics section:
      - `oc_egressgateway_http_request_ratelimit_values_total`
      - `oc_egressgateway_http_request_ratelimit_reject_chain_duration_histogram_seconds_bucket`
      - `oc_egressgateway_http_request_ratelimit_reject_chain_length_histogram_bucket`
      - `oc_egressgateway_http_request_ratelimit_denied_count_total`
      - `oc_egressgateway_congestion_cpu_state`
      - `oc_egressgateway_congestion_system_state`
      - `oc_egressgateway_system_state_duration_percentage`
      - `oc_egressgateway_congestion_level_total`
      - `oc_egressgateway_congestion_level_bucket_total`
      - `oc_egressgateway_congestion_cpu_percentage_bucket`
    - Added the feature related KPIs in the Egress Gateway Pod Protection Using Rate Limiting section.
    - Added the following alerts in the Egress Gateway Pod Protection Using Rate Limiting section:
      - OcnrfEgressDiscardDueToRateLimitMajorThreshold
      - OcnrfEgressDiscardDueToRateLimitCriticalThreshold

General Updates:

- Added the procedure to disable the feature in the Egress Gateway Pod Throttling section.
- Updated the maximum number of supported NF Profiles to 600 in the NRF Supported Services section.
- Added the `log4j2_events_total` metrics in the Platform Metrics section.
- Added the `level` dimension in the NRF Metrics section.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- Updated the dimensions of metric `oc_ingressgateway_connection_failure_total` in the Ingress Gateway Metrics section.
- Updated the error response details for the following error codes:
  - `ONRF-REG-PARTUPD-E0010` in the NFUpdate with Partial Replacement (PATCH) section.
  - `ONRF-DIS-DISC-E0010` in the NFDiscover Service Operation section.
  - `ONRF-CFG-GENOPT-E0010` in the NRF Configuration for General Options section.
  - `ONRF-CFG-SCRNOPT-E0010` in the Screening Options Configuration section.
  - `ONRF-CFG-MGMTOPT-E0010` in the Management Options Configuration section.
  - `ONRF-CFG-DISCOPT-E0010` in the Discovery Options Configuration section.
  - `ONRF-CFG-ACCOPT-E0010` in the Access Token Configuration section.
  - `ONRF-CFG-FWDOPT-E0010` in the Forwarding Options Configuration section.
  - `ONRF-CFG-SLFOPT-E0010` in the SLF Options Configuration section.
  - `ONRF-CFG-GROPT-E0010` in the GeoRedundancy Options Configuration section.
  - `ONRF-CFG-NPTROPT-E0010` in the NAPTR Options Configuration section.
  - `ONRF-CFG-POPROPT-E0010` in the Pod Protection Options Configuration section.
  - `ONRF-CFG-AUTHOPT-E0010` in the Authentication Options Configuration section.
  - `ONRF-CFG-LOGOPT-E0010` in the Logging Options Configuration section.
  - `ONRF-CFG-CSHTOPT-E0010` in the Control Shutdown Options Configuration section.
  - `ONRF-CFG-ROAMOPT-E0010` in the Roaming Options Configuration section.
  - `ONRF-CFG-GWTHOPT-E0010` in the Growth Options Configuration section.
  - `ONRF-CFG-GWTHFOR-E0010` in the Growth Configuration for Forwarding section.

Release 25.1.200-G33631-03, August 2025

- Updated the metric name in the KPI expressions from `container_memory_usage_seconds_total` to `container_memory_usage_bytes` in the document for the following services in the Resource Usage KPIs section:
  - nfregistration
  - nfsubscription
  - nfdiscovery
  - nrfauditor
  - nfaccesstoken
  - nrfconfiguration
  - ingress-gateway
  - egress-gateway
- Updated the following metric names in the document in the NRF Message Feed section:
  - Updated the metric name from `oc_ingressgateway_msgcopy_response_total` to `oc_ingressgateway_msgcopy_responses_total`.
  - Updated the metric name from `oc_egressgateway_msgcopy_response_total` to `oc_egressgateway_msgcopy_responses_total`.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- Updated the metric name in the document from `oc_egressgateway_msgcopy_response_total` to `oc_egressgateway_msgcopy_responses_total` in the Egress Gateway Metrics section.

Release 25.1.200- G33631-02, July 2025

General Updates:

- Added a note in the NRF Message Feed section to mention that the feature works in non-ASM deployment.
- Removed the upgrade and rollback impact section from the Enhanced NRF Set Based Deployment (NRF Growth) section as this section is not applicable.
- Updated the alert name in the document from `OcnrfHeapUsageCrossedMajorThreshold` to `OcnrfHeapUsageCrossedCriticalThreshold` in the OcnrfHeapUsageCrossedCriticalThreshold section.

Release 25.1.200- G33631-01, July 2025

Feature Updates

- **New Features:**
  - **Ingress Gateway Pod Protection By Rate Limiting**
    - Added the Ingress Gateway Pod Protection Using Rate Limiting section to describe the functionality of the feature.
    - Added the following pages in the CNC Console section:
      - Congestion Level Configuration
      - Pod Protection By Rate Limiting
    - Added the following dimensions for the metrics in the NRF Metrics section:
      - `Allowed`
      - `Id`
      - `Action`
      - `RouteLevelColor`
      - `ActionId`
      - `CongestionLevel`
      - `Priority`
      - `ErrorProfile`
    - Added the following metrics in the Ingress Gateway Metrics section:
      - `oc_ingressgateway_http_request_ratelimit_values_total`
      - `oc_ingressgateway_http_request_ratelimit_reject_chain_duration_histogram_seconds`
      - `oc_ingressgateway_http_request_ratelimit_reject_chain_length_histogram`
      - `oc_ingressgateway_http_request_ratelimit_denied_count_total`
      - `oc_ingressgateway_congestion_cpu_state`
      - `oc_ingressgateway_congestion_system_state`

Copyright © 2019, 2025, Oracle and/or its affiliates.

        - `oc_ingressgateway_system_state_duration_percentage`
        - `oc_ingressgateway_congestion_level_total`
        - `oc_ingressgateway_congestion_level_bucket_total`
        - `oc_ingressgateway_congestion_cpu_percentage_bucket`
    - Added the feature related KPIs in the Ingress Gateway Pod Protection Using Rate Limiting section.
    - Added the following alerts in the Ingress Gateway Pod Protection Using Rate Limiting section:
        - OcnrfIngressDiscardDueToRateLimitMajorThreshold
        - OcnrfIngressDiscardDueToRateLimitCriticalThreshold
  - **NF Profile Size Limit**
    - Added the NF Profile Size Limit section to describe the functionality of the feature.
    - Added the following parameters in the NF Management Options section to enable and configure the feature using CNC Console:
        - **NF Profile Limit**
        - **Feature Status**
        - **NF Profile Size Limits**
        - **Error Responses**
    - Added the `ocnrf_nfProfile_size_limit_breached` metric in the NRF NF Metrics section.
    - Added the NF Profile Size Limit Breached KPI in the NRF Service KPIs section.
    - Added the following error response codes for the feature:
        - `ONRF-REG-REGN-E0023` in the NF Registration Microservice section.
        - `ONRF-REG-FULLUPD-E0023` in the NFUpdate with Full Replacement (PUT) section.
        - `ONRF-REG-PARTUPD-E0023` in the NFUpdate with Partial Replacement (PATCH) section.
  - **Writing Messages of the Same Transaction in the Same Kafka Partition**
    Updated the NRF Message Feed section to describe the functionality of the feature.
- **Enhancements:**
  - **Feature Enhancements**
    - The following enhancements are made for the "Support for cnDBTier APIs in CNC Console" feature:
        - Added the cnDBTier Backup Status API in the Support for cnDBTier APIs in CNC Console section.
        - Added the cnDBTier Backup Status CNC Console parameters in the cnDBTier Backup Status section.
    - Added the feature overview of "Pod Level Traffic Rejections" in the Overload Control section.
    - Added the additional metadata sent from NRF to Data Director as part of message feed in the NRF Message Feed section.

Copyright © 2019, 2025, Oracle and/or its affiliates.

        - Added the **Log Rate Control** and **Rate** parameters for controlling the log rates using CNC Console in the Logging Level Options section.
    - **Documentation Enhancements**
        - Updated the Overload Control section with the following:
            - Enhanced the existing feature description.
            - Added the `service_resource_stress` and `service_resource_overload_level` metrics in the Overload Control section.
            - Added the `service` and `isLeaderPod` dimensions for the `load_level` metric in the Overload Control section.
            - Added the descriptions for the `service` and `isLeaderPod` dimensions in the NRF Metrics section.
            - Updated the description of the `type` dimension in the NRF Metrics section.

General Updates:

- Removed the NF Profile Size KPI from the NRF Service KPIs section.
- Added the NF Profile Size Avg and NF Profile Size Max KPIs in the NRF Service KPIs section.
- Added the `oc_ingressgateway_route_overloadcontrol_discard_total` metric in the Ingress Gateway Metrics section.
- Added the `LoadLevel` and `DiscardAction` dimensions in the NRF Metrics section.
- Renamed the `ocnrf_nf_profile_size_bytes` metric to `ocnrf_nf_profile_size` in the NRF NF Metrics section.
- Added the `CUSTOM_NRF_AUDIT_NF_SIZE` as a possible value for the `serviceOperation` dimension in the NRF Cache Data Metrics section for the following metrics:
    - `ocnrf_query_cds_requests_total`
    - `ocnrf_query_cds_responses_total`
    - `ocnrf_query_cds_round_trip_time_seconds`
- Added a note for handling the discovery query for AMF with `guami` in the EmptyList in Discovery Response section.
- Updated the NRF Recommended Features section with the following mandatory features to be enabled during the NRF deployment at the production-level:
    - Pod Level Traffic Discards
    - Ingress Gateway Pod Protection Using Rate Limiting
- Updated the NRF Supported Services section with the maximum number of NFProfiles and NFSubscriptions.
- Updated the description of the Discovery Parameter Value Based Skip SLF Lookup section.
- Added a note in the DataSetId Enhancements section to indicate that DataSetId parameter is ignored for certain scenarios for the NF Profiles.
- Updated the Support for Server Header section to mention that in NRF Roaming scenarios, NRF propagates the server header received from SEPP or peer NRFs to the Consumer NF without any changes.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- Added a note in the DNS NAPTR Update section to mention that the feature works in non-ASM deployment.
- Updated the NRF Last Known Response section to mention that in a NfProfileRetrieval request, NRF sends a response with the requested NfProfile by changing the status of NF from PROBABLE_SUSPENDED to UNDISCOVERABLE.
- Added the `ONRF-REG-PARTUPD-E0008` error code in the NFUpdate with Partial Replacement (PATCH) section.
- Added the `ONRF-CFG-UNKSROP-E0902` error code in the UnSupported Media Type section.
- Removed the following error codes from the NFUpdate with Full Replacement (PUT) section:
  - `ONRF-REG-FULLUPD-E0002`
  - `ONRF-REG-FULLUPD-E0007`
  - `ONRF-REG-FULLUPD-E0008`
  - `ONRF-REG-FULLUPD-E0009`
  - `ONRF-REG-FULLUPD-E0104`
  - `ONRF-REG-FULLUPD-E0112`

Copyright © 2019, 2025, Oracle and/or its affiliates.

1 Introduction

This document provides information about the role of Oracle Communications Cloud Native Core, Network Repository Function (NRF) in 5G Service Based Architecture, how to configure, and use NRF.

NRF is a key component of the 5G Service Based Architecture. NRF maintains an updated repository of all the Network Functions (NFs) available in the operator's network. It also maintains the services provided by each of the NFs in the 5G core that is expected to be instantiated, scaled, and terminated with minimal to no manual intervention. In addition to serving as a repository of the services, NRF also supports discovery mechanisms that allow NFs to discover each other and get the updated status of the desired NFs.

> **Note**
>
> The performance and capacity of the NRF system may vary based on the call model, Feature or Interface configuration, and underlying CNE and hardware environment.

The NRF installation is supported over the following platforms:

- *Oracle Communications Cloud Native Core, Cloud Native Environment (CNE)*
- *Oracle Communications Cloud Native Core OCI Adaptor, NF Deployment in OCI*

For more information, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

NRF supports the following functions:

- Maintains the profiles of the available NF instances and their supported services in the 5G core network.
- Allows consumer NF instances to discover other provider's NF instances in the 5G core network.
- Allows NF instances to track the status of other NF instances.
- Provides Oauth2 based Access Token service for consumer NF authorization.
- Provides specific NF Type selection based on subscriber identity like SUPI, GPSI.
- Supports forwarding of messages from one NRF to another NRF.
- Supports georedundancy to ensure service availability.
- Supports roaming to serve 5G subscribers roaming in a non-home network also known as visited or serving networks, consumer network functions in visited or serving networks need to access the NF Profiles located in the home network of 5G subscribers.

NRF interacts with every other NF in the 5G core network and it supports the above functions through the following services:

- Management Service
- Discovery Service

Copyright © 2019, 2025, Oracle and/or its affiliates.

- AccessToken Service

NRF Availability

NRF availability is dependent on many factors. NRF applications are designed to achieve 99.999% availability, according to the applicable Telecommunications Industry Association TL9000 standards, with the following deployment requirements:

- Deploy on a Cloud Native Environment with at least 99.999% Availability.
- Deploy with n + k application redundancy, where k is greater than or equal to one.
- Maintain production software within n-3 software releases, where n is the current general availability release.
- Apply bug fixes, critical patches, and configuration recommendations provided by Oracle promptly.
- Maintain disaster recovery procedures external to the applications for the reconstruction of lost or altered files, data, programs, or Cloud Native environment.
- Install, configure, operate, and maintain NRF as per Oracle's applicable installation, operation, administration, and maintenance specifications.
- Maintain an active support contract and provide access to the deployed NRF and your personnel to assist Oracle in addressing any outage.

NRF availability is measured for each calendar year and is calculated as follows:

Table 1-1 Measuring NRF Availability

Availability	Description
Planned Product Availability	(Product available time in each month) less (**Excluded Time** (defined below) in each month).
Actual Product Availability	(Planned Product Availability) less (any Unscheduled Outage)
Product Availability Level	(Actual Product Availability across all Production instances divided by Planned Product Availability across all Production instances) x 100

Note

Excluded Time means:

- Scheduled maintenance time.
- Lack of power or backhaul connectivity, except to the extent that such lack of backhaul connectivity was caused directly by the CNC NF.
- Hardware failure.
- Issues arising out of configuration errors or omissions.
- Failures caused by third-party equipment or software not provided by Oracle.
- Occurrence of any event under Force Majeure.
- Any time associated with failure to maintain the recommended architecture and redundancy model requirements above.

Copyright © 2019, 2025, Oracle and/or its affiliates.

1.1 References

Following are the reference documents:

- *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*
- *Oracle Communications Cloud Native Core, Network Repository Function Network Impact Report*
- *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*
- *Oracle Communications Cloud Native Core, Cloud Native Environment User Guide*
- *Oracle Communications Cloud Native Core, cnDBTier User Guide*
- *Oracle Communications Cloud Native Configuration Console User Guide*
- *Oracle Communications Cloud Native Core Automated Test Suite User Guide*
- *Oracle Communications Cloud Native Core, OCI Deployment Guide*
- *Oracle Communications Cloud Native Core, OCI Adaptor User Guide*
- *Oracle Communications Cloud Native Core, Certificate Management User Guide*
- *Oracle Communications Network Analytics Data Director User Guide*
- *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*
- *Oracle Communications Cloud Native Core Release Notes*
- *Oracle Communications Cloud Native Core Licensing Information User Guide*
- *Oracle Communications Cloud Native Core Solution Upgrade Guide*
- *Oracle Communications Cloud Native Core Security Guide*

Copyright © 2019, 2025, Oracle and/or its affiliates.

2 NRF Supported Services

This section includes information about the services supported by NRF.

NRF supports the following services:

NRF Management Service

The NRF Management service is identified by the service name `Nnrf_NFManagement`.

NRF supports the following management services:

> **Note**
>
> The respective service name is mentioned next to each service.

- **Register NF instance** (`NFRegister`): Allows the NF instance to register its NF profile in the NRF along with the list of services provided by the NF instance.
- **Update NF instance** (`NFUpdate`): Enables the NF instance to partially update or replace the parameters of its NF profile in the NRF. It also allows to add or delete services provided by the NF instance.
  This operation supports the following:
  - Complete replacement of NF profile
  - Add, remove or update attributes of NF Profile
  - Heart beat and load information of NF
- **Deregister NF instance** (`NFDeregister`): Enables the NF instance to deregister its NF profile and the services provided by the NF instance from the 5G network.
- **Subscribe to Status** (`NFStatusSubscribe`): Enables the NF instance to subscribe to the status changes of other NF instances registered in the NRF.
- **Unsubscribe to Status** (`NFStatusUnsubscribe`): Enables the NF instance to unsubscribe the status changes of other NF instances.
- **Notifications of Status** (`NFStatusNotify`): Sends status notifications to subscribed NFs.
- **Retrieval of NF list** (`NFListRetrieval`): Allows the retrieval of a list of NF Instances that are currently in the registered state in NRF. This service operation is not allowed to be invoked from the NRF in a different PLMN.
- **Retrieval of NF Profiles** (`NFProfileRetrieval`): Allows the retrieval of the NF profile of a given NF instance that are currently in the registered state in NRF. This service operation is not allowed to be invoked from the NRF in a different PLMN.

NRF Discovery Service

The NRF discovery service is identified by the service name `Nnrf_NFDiscovery Service`.

NRF supports the following discovery service:

Copyright © 2019, 2025, Oracle and/or its affiliates.

- **Discover NF instance** (`NFDiscover`): NRF supports the discovery of the NRF profile of the NF instances or NF services that match certain input criteria.

NRF Access Token Service

The NRF Access Token service handles 3GPP defined Access Token service operations. Oauth2.0 based token is provided by NRF according to inputs provided by consumer network function in access token request. The NRF Access Token service is identified by the service name `Nnrf_AccessToken`.

NRF supports the following access token service:

- Access Token (`Nnrf_AccessToken`): NRF supports issuing OAuth2 token to consumer NFs for accessing specific Producer Services.

> **Note**
>
> The performance and capacity of the NRF system may vary based on the call model, Feature or Interface configuration, and underlying CNE and hardware environment.

Maximum Supported NFProfiles and NFSubscriptions Count

NRF supports maximum of 600 NFProfiles.

NRF supports maximum of 1000 NFSubscriptions

2.1 Supported NF Types

NRF supports the following NF Types. All NRF service operations are supported for these NF Types.

Table 2-1 Supported NF Types

Network Function	Definition
5G_EIR	5G Equipment Identity Register
AF	Application Function
AMF	Access and Mobility Management Function
AUSF	Authentication Server Function
BSF	Binding Support Function
CBCF	Cell Broadcast Center Function
CHF	Charging Function
DRA	Diameter Routing Agent
GMLC	Gateway Mobile Location Center
ICSCF	Interrogating Call Session Control Function
IMS_AS	IP Multimedia Subsystem Application Server
LMF	Location Management Function
MME	Mobility Management Entity
N3IWF	non-3GPP Interworking Function
NEF	Network Exposure Function
NRF	Network Repository Function
NSSF	Network Slice Selection Function

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 2-1 (Cont.) Supported NF Types

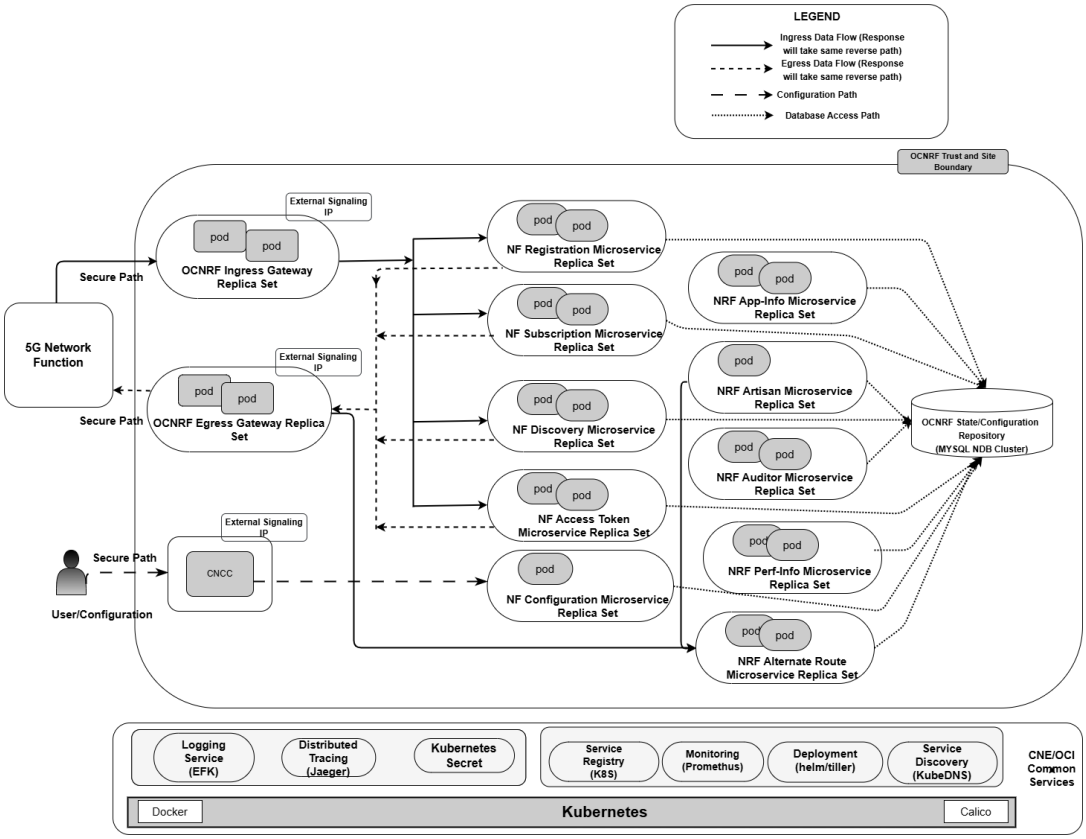
Network Function	Definition
NSSAAF	Network Slice Specific Authentication and Authorization Function
NWDAF	Network Data Analytic Function
PCF	Policy Control Function
SCP	Service Communication Proxy
SCEF	Service Capability Exposure Function
SCSAS	Security Assurance Specification
SEPP	Security Edge Protection Proxy
SMF	Session Management Function
SMSF	Short Message Service Function
SPAF	Secured Packet Application Function
SOR_AF	Steering of Roaming Application Function
UCMF	UE Capability Management Function
UDR	Unified Data Repository
UDM	Unified Data Management
UDSF	Unstructured Data Storage Function
UPF	User Plane Function

Copyright © 2019, 2025, Oracle and/or its affiliates.

3
NRF Architecture

NRF comprises various microservices deployed in Kubernetes based Cloud Native Environment (CNE) and Oracle Cloud Infrastructure (OCI). Some common services like logs or metrics data collection, analysis, and graphs or charts visualization, and so on are provided by the environment. The microservices integrate with the environment and provide the necessary data.

Figure 3-1 NRF Architecture



Following are the components of the NRF product:

- **NF Registration Microservice**
  This microservice receives and handles the following service operations:
  - NFRegister service requests from the NFs
  - NFUpdate service requests from the NFs
  - NFDeregister service requests from the NFs
  - NFListRetrieval service requests from the NFs
  - NFProfileRetrieval service requests from the NFs

Copyright © 2019, 2025, Oracle and/or its affiliates.

  - Heart-beat messages from the NFs
- **NF Subscription Microservice**
  This microservice performs the following service operations:
  - receives and handles NFStatusSubscribe service requests from the NFs
  - receives and handles NFStatusUnsubscribe service requests from the NFs
  - sends NFStatusNotify service requests towards the subscribed NFs
- **NF Discover Microservice**
  This microservice receives and handles the following service operations:
  - NFDiscover service requests from the NFs
- **NF Access Token Microservice**
  This microservice handles 3GPP-defined AccessToken service operations. Oauth2.0 based token is provided by NRF according to inputs provided by consumer network function in access token request.
- **NRF Auditor Microservice**
  This microservice is internal to NRF. This microservice performs the following tasks:
  - finds and deletes the expired subscription records
  - finds and deletes the profile records which have been SUSPENDED for a very long time
  - monitors the heart-beat expiry, mark the NF profiles as suspended and act appropriately on the suspended NF profiles
- **NRF Configuration Microservice**
  This microservice is used to configure NRF. These configurations can be changed dynamically by an operator or user using REST based interface. This configuration data is managed by the NRF configuration service and is stored in a separate data store.
- **NRF Ingress Gateway Microservice**
  This microservice is an entry point for accessing NRF supported service operations.
- **NRF Egress Gateway Microservice**
  This microservice is responsible to route NRF initiated egress messages to other NFs.
- **App-Info Microservice**
  This microservice is responsible to get the status of microservices related to NFManagement Service operations (that is, NF Registration microservice, NF Subscription microservice, NRF Auditor microservice).

  NF Registration, NF Subscription, NRF Auditor microservices uses this microservice to get the status of other two microservices. In case any of the services is found as down, status asking microservice will reject the incoming messages. This microservice is also responsible to fetch DB replication status whether it is active or inactive.
- **NRF Artisan Microservice**
  This is an optional utility microservice which gets deployed to perform feature specific tasks. For example, building candidate SLFs to which NRF will send SLF Query.
- **Perf-Info Microservice**
  This is an optional microservice which gets deployed to enable overload control in NRF. It helps in computing CPU or Memory utilization of NRF core services and performs Overload Control Manager role.
- **Alternate-Route Microservice**
  This is an optional microservice which gets deployed when NRF needs to perform DNS SRV based queries to the DNS server to resolve Virtual FQDNs or update NAPTR records.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- **Cache Data Microservice**
  The microservice is responsible for caching the state data in memory from the database. The NfRegistration, NfSubscription, NfDiscovery, NfAccesstoken, NrfConfiguration, and NrfArtisan microservices query the Cache Data Service (CDS) to get the state data to process NRF supported service operations.

  CDS builds and maintains the state data of the local NRF set by reading the state data from cnDBTier. If the NRF Growth feature is enabled, CDS maintains the state data of the remote NRF sets by synching with the remote NRF sets periodically. In case of CDS failure, NRF microservices rely on the cnDBTier to obtain the NF profile information local data and fulfill the service operation.

Ingress and Egress Gateway Traffic Management

For more information on Ingress and Egress Gateway Traffic Management, see *Oracle Communications Cloud Native Core, Cloud Native Environment User Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

4
NRF Features

This section explains the NRF features.

> **Note**
>
> The performance and capacity of the NRF system may vary based on the call model, Feature or Interface configuration, and underlying CNE and hardware environment.

Maximum Supported NFProfiles and NFSubscriptions Count

NRF supports maximum of 600 NFProfiles. NRF supports maximum of 1000 NFSubscriptions.

NRF Capacity

NRF supports up to 52K TPS. For the specific call model and detailed resource requirements, see *Oracle Communications Cloud Native Core, Network Repository Function Benchmarking Guide*.

4.1 Pod Protection Using Rate Limiting

Pod Protection Using Rate Limiting comprises of the following features:

- Ingress Gateway Pod Protection Using Rate Limiting
- Egress Gateway Pod Protection Using Rate Limiting

For more information about these features and their processing, see the following sections:

- Ingress Gateway Pod Protection Using Rate Limiting
- Egress Gateway Pod Protection Using Rate Limiting

4.1.1 Egress Gateway Pod Protection Using Rate Limiting

The Egress Gateway microservice manages all outgoing traffic from Network Functions (NFs). The Egress Gateway microservice may become vulnerable due to overload conditions during unexpected traffic spikes, uneven traffic distribution, or bursts caused by network fluctuations. To ensure system stability and prevent potential network outages, a protection mechanism for the Egress Gateway microservice is necessary.

With the implementation of this feature, rate-limiting mechanism is applied for Egress Gateway pods. This mechanism allows to set a maximum number of requests that a pod can process. When the request rate exceeds the threshold, the pods take action to protect themselves. They either reject additional requests with a custom error code or allow them, depending on the configuration. Each Egress Gateway pod has its own Pod Protection Policer, which is applied individually.

The Egress Gateway's configuration for this feature includes:

Copyright © 2019, 2025, Oracle and/or its affiliates.

- **Fill Rate**: This sets the maximum number of requests a pod can handle in a defined time interval. It is also possible to allocate the specific percentage for various path or method combinations.
- **Error Code Profiles**: This allows to reject the requests with a predefined error code profile.
- **Congestion Configuration**: This allows to configure the congestion levels based on CPU resource utilization.
- **Default Priority**: This allows to configure the priority for different routes.
- **Denied Request Actions**: This defines the action to be taken when the fill rate is exceeded. This action is based on the priority defined in the denied request actions and the action configured. The possible options for action are 'CONTINUE', which allows requests to be processed, or 'REJECT', which rejects requests with a specified error code. Egress Gateway will reject all the requests which exceeds the fill rate unless specific 'CONTINUE' action is defined in the denied request actions.

The fill rate represents the maximum number of requests a pod can handle when traffic is uniformly distributed within a span of one second interval. However, in bursty traffic scenarios where the traffic is not uniformly distributed within a span of one second interval, even if the average traffic received is within the fill rate, still traffic failure may be expected.

Congestion level is checked in a fixed interval (`congestionconfig.refreshInterval`) and the action will be decided based on the congestion level in that interval. It is possible that the CPU levels might toggle between these intervals causing the congestion levels to switch frequently. To know the detected congestion level, see the metric `oc_egressgateway_congestion_level_bucket_total`.

The following table describes NRF's configuration for different congestion levels for Egress Gateway pods:

Table 4-1 Congestion Levels of Pods

Level Value	Abatement Threshold	Onset Threshold	Level Name	Description
1	58	65	Normal	The default congestion level of pod is "0". The pod moves from Normal to default when the CPU is less than the abatement of Normal (58). The pod moves from default to Normal state when the CPU exceeds onset of Normal (65). The pod moves from Danger of Congestion to Normal state, when the CPU is less than abatement of Danger of Congestion (70).
2	70	75	Danger of Congestion	The pod moves from Danger of Congestion to Normal, when the CPU is less than abatement of Danger of Congestion (70). The pod moves from Normal to Danger of Congestion state when the CPU exceeds onset of Danger of Congestion (75). The pod moves from Congested to Danger of Congestion when the CPU is less than abatement of Congested (80).

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-1 (Cont.) Congestion Levels of Pods

Level Value	Abatement Threshold	Onset Threshold	Level Name	Description
3	80	85	Congested	The pod moves from Congested to Danger of Congestion when the CPU is less than abatement of Congested (80). The pod moves from Danger of Congestion to Congested state when the CPU exceeds onset of Congested (85).

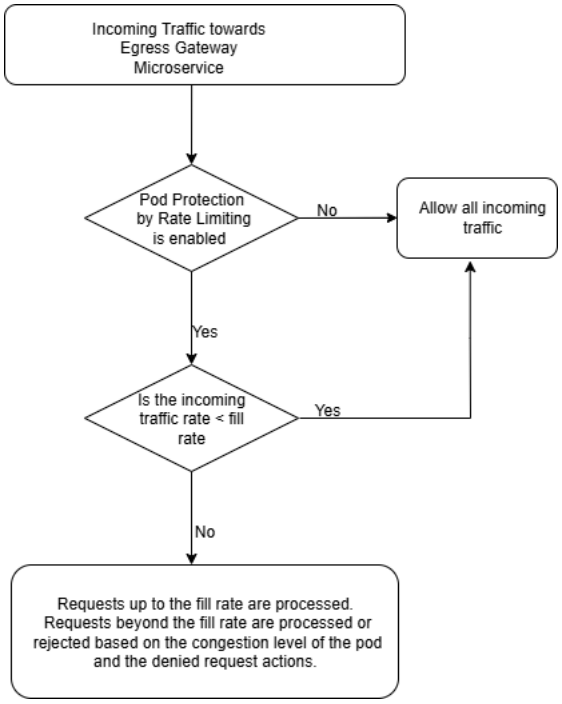
Note

- If the current level is at L3 (Congested with onset value of 85) and the CPU value is 66, the level would be marked as L1 (Normal). The abatement value is considered only when the switching is between the adjacent levels (L2 (Danger of Congestion) to L1 (Normal) and L3 (Congested) to L2 (Danger of Congestion)).
- When NRF Discovery microservice receives an overload response from Egress Gateway microservice, it does not retry or re-reroute this request. An error response will be generated based on the specific feature configurations like NRF Forwarding, Subscriber Location Function, and Roaming.
- Egress Gateway microservice performs routes validation before applying any other gateway service filters. Consequently, requests that don't match a valid route in `routesConfig` in REST are not subject to rate limiting. NRF has always a default route configured. Unless the routes are not explicitly removed, all the EGW messages will go through rate limiting. For more details on routesConfig, see *Oracle Communications Cloud Native Core, Network Repository Function REST API Guide.*

The following diagram explains the Egress Gateway Pod Protection Using Rate Limiting mechanism:

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-1 Egress Gateway Pod Protection Using Rate Limiting



1. NRF send traffic to the Egress Gateway microservice.
2. Egress Gateway pod checks if the Pod Protection Using Rate Limiting feature is enabled or not.
   a. If the feature is enabled, Egress Gateway compares the outgoing traffic rate with the fill rate that is defined for the rate limiting feature.
      - If the outgoing request rate is below the fill rate, all the outgoing traffic are processed.
      - If the outgoing request rate exceeds the fill rate, requests up to the fill rate are processed. The requests beyond the fill rate are processed or rejected based on the congestion level of the pod and the denied request actions (priority and action defined). If there is no priority defined in the denied request actions all the requests will be rejected.
        – At default or Normal congestion levels all the requests are processed.
        – At Danger of Congestion, the requests with the defined priority (1-2) traffic are processed.
        – At Congested level, only the highest priority traffic (priority 1) is processed; lower-priority traffic is rejected.

Examples

Consider the following configuration for the `{apiRoot}/nrf/nf-common-component/v1/egw/podProtectionByRateLimiting` API:

```
{
    "enabled": true,
    "fillRate": 2700,
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
"routes": [
    {
        "id": 1,
        "methods": [
            "GET"
        ],
        "path": "/nudr-group-id-map/v1/nf-group-ids",
        "percentage": 70,
        "defaultPriority": 3
    },
    {
        "id": 2,
        "methods": [
            "GET"
        ],
        "path": "/nnrf-disc/v1/nf-instances",
        "percentage": 4,
        "defaultPriority": 2
    },
    {
        "id": 3,
        "methods": [
            "POST"
        ],
        "path": "/nnrf-nfm/v1/subscriptions",
        "percentage": 3,
        "defaultPriority": 2
    },
    {
        "id": 4,
        "methods": [
            "POST"
        ],
        "path": "/oauth2/token",
        "percentage": 3,
        "defaultPriority": 2
    },
    {
        "id": 5,
        "methods": [
            "GET"
        ],
        "path": "/nnrf-nfm/v1/nf-instances/**",
        "percentage": 10,
        "defaultPriority": 2
    },
    {
        "id": 6,
        "methods": [
            "POST"
        ],
        "path": "/**",
        "percentage": 10,
        "defaultPriority": 1
    }
],
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
    "errorCodeProfile": "error429",
    "deniedRequestActions": [
        {
            "id": 1,
            "congestionLevel": 0,
            "action": "CONTINUE"
        },
        {
            "id": 2,
            "congestionLevel": 1,
            "action": "CONTINUE"
        },
        {
            "id": 3,
            "priority": "1-2",
            "congestionLevel": 2,
            "action": "CONTINUE"
        },
        {
            "id": 4,
            "priority": "1",
            "congestionLevel": 3,
            "action": "CONTINUE"
        }
    ],
    "priorityHeaderName": "3gpp-Sbi-Message-Priority",
    "defaultPriority": 24
}
```

Any requests which are not matching with the action configured and the priority level configured in the denied request actions are rejected.

For example, in congestion level 2, requests other than priority 1 and 2 are candidates for rejection. Similarly for congestion level 3, requests other than priority 1 are candidates for rejection. There is no need for defining rejection explicitly for these actions.

In the above configuration, the total fill rate is set to 2,700. The individual route definitions are as follows:

- Route Id 1: 70% allocation (1890 out of 2700), priority 3
- Route Id 2: 4% allocation (108 out of 2700), priority 2
- Route Id 3: 3% allocation (81 out of 2700), priority 2
- Route Id 4: 3% allocation (81 out of 2700), priority 2
- Route Id 5: 10% allocation (270 out of 2700), priority 2
- Route Id 6: 10% allocation (270 out of 2700), priority 1

Example 1-Handling TPS of 2835 in Danger of Congestion Mode

If the incoming Transactions Per Second (TPS) is 2835, this exceeds the configured fill rate by 135 requests. As a result, the system enters a "Danger of Congestion" state.

Request distribution details:

- Route Ids 2, 3, 4, and 6 receive incoming TPS equal to their allocated fill rates. Therefore, all requests for these routes are accepted and processed:

Copyright © 2019, 2025, Oracle and/or its affiliates.

  - Route Id 2: 108 requests (4%)
  - Route Id 3: 81 requests (3%)
  - Route Id 4: 81 requests (3%)
  - Route Id 6: 270 requests (10%)
- Route Id 5 receives 135 requests, which is less than its allocated fill rate of 270 requests (10%). As such, all 135 requests are accepted for processing.
- Route Id 1 receives 2160 requests, exceeding its allocation of 1890 requests (70%). The system initially accepts up to the allocated 1,890 requests for processing. Since Route Id 5 processed fewer requests than its allocation (135 instead of 270), the unused capacity (135 requests) is applied to Route Id 1, allowing a total of 2025 requests for processing on this route.
- The remaining 135 requests from Route Id 1 are rejected, as the configured fill rate is already met and Route Id 1 has the lowest priority (priority 3). According to the specified congestion management policy, requests with priority 3 are rejected at congestion level 2.

Example 2- Handling TPS of 2970 in Congested Mode

When the incoming Transactions Per Second (TPS) reaches 2970, it exceeds the fill rate of 2700 by 270 requests. The system enters Congested mode.

Route processing details

- Route Id 2 receives 108 requests, corresponding to the defined 4%. All 108 requests are accepted for processing.
- Route Id 3 receives 81 requests, which meets the defined 3%. All 81 requests are processed.
- Route Id 4 also receives 81 requests, equaling the defined 3%. All requests are processed.
- Route Id 5 receives 270 requests, aligning with the defined 10%. All 270 requests are allowed.
- Route Id 6 receives 540 requests, exceeding the defined 10%. However, since Route Id 6 has the highest default priority (priority 1) and the denied action request is set to CONTINUE for this priority, all 540 requests are processed.
- Route ID 1 receives 1890 requests, meeting the defined 70% allocation. Due to lower priority (priority 3) under congestion, only 1620 requests are accepted, and the remaining 270 requests are rejected.

Managing the feature

Prerequisite

> **Warning**
>
> This feature is disabled by default. It is recommended to enable the Egress Gateway Pod Protection Using Rate Limiting feature over the Egress Gateway Pod Throttling feature. The Egress Gateway Pod Throttling feature will be deprecated in the upcoming releases.You must disable the Egress Gateway Pod Throttling feature if enabled by performing a Helm upgrade before proceeding to enable this feature.

For more information about disabling the existing Egress Gateway Pod Throttling feature, see Egress Gateway Pod Throttling.

Copyright © 2019, 2025, Oracle and/or its affiliates.

This feature uses the following Helm parameters which are preconfigured with the required details:

- `global.egwGeneratedErrorCheckDuringOverLoad`
- `global.egwResponseCodesDuringOverLoad`
- `global.noRetryDuringEgwOverload`

For more information about the above parameters, see the "Global Parameters" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Configure

- **Configure using REST API:** Perform the following configurations for this feature:
  - Configure the Error code profile using the `{apiRoot}/nrf/nf-common-component/v1/egw/errorcodeprofiles` API.
  - Configure the congestion level of the pods based on CPU using the `{apiRoot}/nrf/nf-common-component/v1/egw/congestionConfig` API.

For more information about these APIs, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

- **Configure using CNC Console**:
  - Configure the Error code Profiles, as described in Error Code Profile.
  - Create or update the Congestion Level Configuration, as in Congestion Level Configuration.

Enable

- **Enable using REST API**: Configure the attributes in the `{apiRoot}/nrf/nf-common-component/v1/egw/podProtectionByRateLimiting` API to enable the feature.
  For more information about this API, see "Pod Protection By Rate Limiting" in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Configure the parameters as mentioned in Pod Protection By Rate Limiting.

> **Note**
>
> Use the recommended values in the above configuration.

Observability

Metrics

The following metrics are added for this feature:

- `oc_egressgateway_http_request_ratelimit_values_total`
- `oc_egressgateway_http_request_ratelimit_reject_chain_duration_histogram_seconds_bucket`
- `oc_egressgateway_http_request_ratelimit_reject_chain_length_histogram_bucket`
- `oc_egressgateway_http_request_ratelimit_denied_count_total`
- `oc_egressgateway_congestion_cpu_state`

Copyright © 2019, 2025, Oracle and/or its affiliates.

- `oc_egressgateway_congestion_system_state`
- `oc_egressgateway_system_state_duration_percentage`
- `oc_egressgateway_congestion_level_total`
- `oc_egressgateway_congestion_level_bucket_total`
- `oc_egressgateway_congestion_cpu_percentage_bucket`

For more information about the metrics, see Egress Gateway Metrics.

Alerts

The following alerts are added in the Egress Gateway Pod Protection Using Rate Limiting section:

- OcnrfEgressDiscardDueToRateLimitMajorThreshold
- OcnrfEgressDiscardDueToRateLimitCriticalThreshold

KPIs

The following KPIs are added for this feature:

- Allowed Requests
- Reject Chain Length
- Discard Action
- Congestion Level

For more information about KPIs, see Egress Gateway Pod Protection Using Rate Limiting.

Maintain

If you encounter alerts at system or application levels, see the NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

1. Collect the logs: For more information on collecting logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. Raise a service request: See My Oracle Support for more information on how to raise a service request.

4.1.2 Ingress Gateway Pod Protection Using Rate Limiting

The Ingress Gateway microservice manages all incoming traffic from Network Functions (NFs). The Ingress Gateway microservice may become vulnerable due to overload conditions during unexpected traffic spikes, uneven traffic distribution, or bursts caused by network fluctuations. To ensure system stability and prevent potential network outages, a protection mechanism for the Ingress Gateway microservice is necessary.

With the implementation of this feature, rate-limiting mechanism is applied for Ingress Gateway pods. This mechanism allows to set a maximum number of requests that a pod can process. When the request rate exceeds the threshold, the pods take action to protect themselves. They either reject additional requests with a custom error code or allow them, depending on the configuration. Each Ingress Gateway pod has its own Pod Protection Policer, which is applied individually.

The Ingress Gateway's configuration for this feature includes:

Copyright © 2019, 2025, Oracle and/or its affiliates.

- **Fill Rate**: This sets the maximum number of requests a pod can handle in a defined time interval. It is also possible to allocate the specific percentage for various path or method combinations.
- **Error Code Profiles**: This allows to reject the requests with a predefined error code profile.
- **Congestion Configuration**: This allows to configure the congestion levels based on CPU resource utilization.
- **Denied Request Actions**: This defines the action to be taken when the fill rate is exceeded. The options are 'CONTINUE', which allows requests to be processed, or 'REJECT', which rejects requests with a specified error code.
  Ingress Gateway will reject all the requests which exceeds the fill rate unless specific 'CONTINUE' action is defined in the denied request actions.

The fill rate represents the maximum number of requests a pod can handle when traffic is uniformly distributed within a span of one second interval. However, in bursty traffic scenarios where the traffic is not uniformly distributed within a span of one second interval, even if the average traffic received is within the fill rate, still traffic failure may be expected.

Congestion level is checked in a fixed interval (`congestionconfig.refreshInterval`) and the action will be decided based on the congestion level in that interval. It is possible that the CPU levels might toggle between these intervals causing the congestion levels to switch frequently. To know the detected congestion level, see the metric `oc_ingressgateway_congestion_level_bucket_total`.

The following table describes NRF's configuration for different congestion levels for Ingress Gateway pods:

Table 4-2 Congestion Levels of Pods

Level Value	Abatement Threshold	Onset Threshold	Level Name	Description
1	58	65	Normal	The default congestion level of pod is "0". The pod moves from Normal to default when the CPU is less than the abatement of Normal (58). The pod moves from default to Normal state when the CPU exceeds onset of Normal (65). The pod moves from Danger of Congestion to Normal state, when the CPU is less than abatement of Danger of Congestion (70).
2	70	75	Danger of Congestion	The pod moves from Danger of Congestion to Normal, when the CPU is less than abatement of Danger of Congestion (70). The pod moves from Normal to Danger of Congestion state when the CPU exceeds onset of Danger of Congestion (75). The pod moves from Congested to Danger of Congestion when the CPU is less than abatement of Congested (80).

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-2 (Cont.) Congestion Levels of Pods

Level Value	Abatement Threshold	Onset Threshold	Level Name	Description
3	80	85	Congested	The pod moves from Congested to Danger of Congestion when the CPU is less than abatement of Congested (80). The pod moves from Danger of Congestion to Congested state when the CPU exceeds onset of Congested (85).

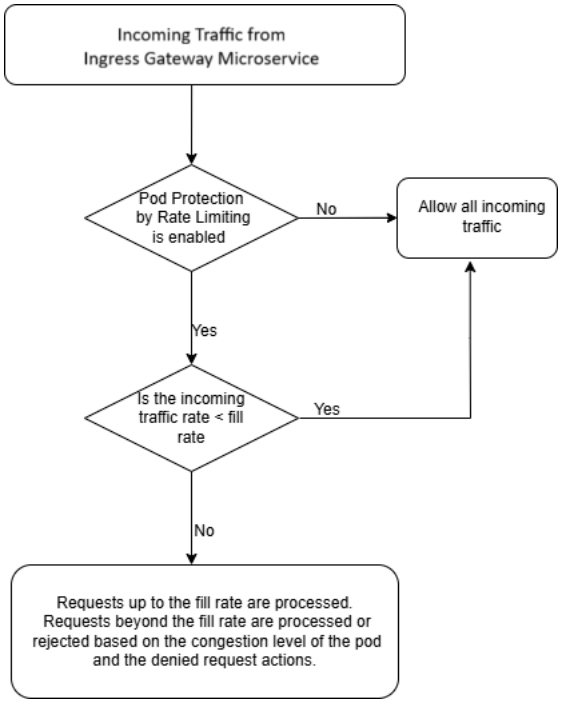
Note

- If the current level is at L3 (Congested with onset value of 85) and the CPU value is 66, the level would be marked as L1 (Normal). The abatement value is considered only when the switching is between the adjacent levels (L2 (Danger of Congestion) to L1 (Normal) and L3 (Congested) to L2 (Danger of Congestion)).
- When NRF rejects the requests, it is expected that the consumer NFs reroute to alternate NRFs.
- Ingress Gateway microservice rate limiting only applies to signaling messages and does not impact internal messages, such as requests from perf-info to Ingress Gateway microservice, message copy requests.
- Ingress Gateway microservice performs routes validation before applying any other gateway service filters. Consequently, requests that don't match a valid route in `routesConfig` in Helm are not subject to rate limiting. For more details on routesConfig, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide* .

The following diagram explains the Ingress Gateway Pod Protection Using Rate Limiting mechanism:

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-2 Ingress Gateway Pod Protection Using Rate Limiting



1. Consumer NFs send traffic to the Ingress Gateway microservice.
2. Ingress Gateway pod checks if the Pod Protection Using Rate Limiting feature is enabled or not.
   a. If the feature is enabled, Ingress Gateway compares the incoming traffic rate with the fill rate that is defined for the rate limiting feature.
      i. If the incoming request rate is below the fill rate, all the incoming traffic are processed.
      ii. If the incoming request rate exceeds the fill rate, requests up to the fill rate are processed. The requests beyond the fill rate are processed or rejected based on the congestion level of the pod and the denied request actions.

Consider the following examples:

- if fillRate is configured as 1000,
  - if the incoming Transaction Per Second (TPS) is 1100, congestionLevel is Normal, and action is configured as CONTINUE, all the 1100 requests are processed.
  - if the incoming TPS is 1300, congestionLevel is Danger of Congestion, and action is configured as CONTINUE, all the 1300 requests are processed.
  - if the incoming TPS is 1500, congestionLevel is Congested, and action is configured as REJECT, 1000 requests are processed successfully and the 500 requests are rejected with the configured error code.
- if two routes, id1 configured with 30 percentage and id2 with 70 percentage, and fillRate is configured as 1000,
  - if the incoming TPS is 1100, congestionLevel is Normal, and action is configured as CONTINUE, all the 1100 requests are processed.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- if the incoming TPS is 1300, fill rate is 1000, congestionLevel is Danger of Congestion, and action is configured as CONTINUE, all requests are processed successfully.
- if the incoming TPS is 1500, congestionLevel is Congested, and action is configured as REJECT, 300 requests from route id1 is processed and 700 requests from route id2 is processed, and 500 remaining TPS will be rejected.

Managing the feature

Prerequisites

> **Note**
>
> It is recommended to enable the Ingress Gateway Pod Protection Using Rate Limiting feature over the Ingress Gateway pod protection feature. The Ingress Gateway Pod Protection feature will be deprecated in the upcoming releases. You must disable the Ingress Gateway Pod Protection before proceeding to this feature. For more information about disabling the existing pod protection feature, see Ingress Gateway Pod Protection.

Enable

- **Configure using REST API**: Perform the following configurations for this feature:
  - Configure the Error code profile using the `{apiRoot}/nrf/nf-common-component/v1/igw/errorcodeprofiles` API.
    Configure the `ingressgateway.errorCodeProfiles` in Helm. Perform a Helm upgrade. This `ingressgateway.errorCodeProfiles` value will be used when request is rejected at Ingress Gateway.
  - Configure the congestion level of the pods based on CPU using the `{apiRoot}/nrf/nf-common-component/v1/igw/congestionConfig` API.
    For more information about these APIs, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure using CNC Console**:
  - Configure the Error code Profiles, as described in Error Code Profile.
    Configure the `ingressgateway.errorCodeProfiles` in Helm. Perform a Helm upgrade. This `ingressgateway.errorCodeProfiles` value will be used when request is rejected at Ingress Gateway.
  - Create or update the Congestion Level Configuration, as in Congestion Level Configuration.
- **Enable using REST API**: Configure the attributes in the `{apiRoot}/nrf/nf-common-component/v1/igw/podProtectionByRateLimiting}` API to enable the feature.
  For more information about this API, see "Pod Protection By Rate Limiting" in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Configure the parameters as mentioned in Pod Protection By Rate Limiting.

Observability

Metrics

The following metrics are added for this feature:

Copyright © 2019, 2025, Oracle and/or its affiliates.

- `oc_ingressgateway_http_request_ratelimit_values_total`
- `oc_ingressgateway_http_request_ratelimit_reject_chain_duration_histogram_seconds`
- `oc_ingressgateway_http_request_ratelimit_reject_chain_length_histogram`
- `oc_ingressgateway_http_request_ratelimit_denied_count_total`
- `oc_ingressgateway_congestion_cpu_state`
- `oc_ingressgateway_congestion_system_state`
- `oc_ingressgateway_system_state_duration_percentage`
- `oc_ingressgateway_congestion_level_total`
- `oc_ingressgateway_congestion_level_bucket_total`
- `oc_ingressgateway_congestion_cpu_percentage_bucket`

For more information about the metrics, see Ingress Gateway Metrics.

Alerts

The following alerts are added in the Ingress Gateway Pod Protection Using Rate Limiting section.

- OcnrfIngressDiscardDueToRateLimitMajorThreshold
- OcnrfIngressDiscardDueToRateLimitCriticalThreshold

KPIs

The following KPIs are added for this feature:

- Allowed Request Rate Per Route Id
- Total Rejections Chain Length
- Discard Request Action Traffic Rate
- Pod Congestion Level

For more information about KPIs, see Ingress Gateway Pod Protection Using Rate Limiting.

Maintain

If you encounter alerts at system or application levels, see the NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

1. **Collect the logs**: For more information on collecting logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.2 NF Profile Size Limit

The Nnrf_NFManagement service allows an NF to register, update, or deregister its profile in the NRF. The NF Profile consists of general parameters of the NF Instance and also the parameters of the different NF service instances exposed by the NF Instance. NRF uses the registered profiles for service operations like NfListRetrieval, NfProfileRetrieval, NfDiscover, and NFAccessToken. It is observed that when the size of the NF Profile registered is large, the

Copyright © 2019, 2025, Oracle and/or its affiliates.

performance of these service operations degrades due to increased processing time of larger profiles and can cause higher resource utilization and latency.

This feature allows to specify the maximum limit of the NF Profile size that can be registered with NRF. The NF Profile size is evaluated during the NFRegister or NFUpdate service operation, and if the size is within the configured maximum limit, the service operation is allowed. If the profile size breaches the configured thresholds, the service operation gets rejected.

NRF behavior for each of the service operations is defined as follows:

- NFRegister or NFUpdate (PUT):
  - NRF receives the NF Profile from the NF and evaluates the size of the NF Profile against the configured maximum allowed size.
  - The NFRegister or NFUpdate service operation succeeds if the NF Profile size is below the configured size.
  - If the size of the NF Profile has breached the configured size, the NFRegister or NFUpdate service operation fails with a configurable error response.
- NFUpdate (Partial Update):
  NRF receives the update that the NF wishes to apply on NF Profiles registered at NRF. NRF evaluates the size of the NF Profile after applying the updates on registered nfProfile.

  If the size of the NF Profile is below the configured size, the NFUpdate service operation succeeds.

  If the size of the NF Profile has breached the configured size, the NFUpdate service operation fails with a configurable error response.
- NFHeartBeat:
  NfProfile size is not evaluated for NFHeartbeat service operation. The NFs can continue to Heartbeat even if their profile size has breached the threshold. This can occur only if the bigger profile was registered before enabling the feature.

The `ocnrf_nfProfile_size_limit` metric shows the size of the nfProfile registered at NRF. The size of the nfProfile is calculated as per the payload received while registering or updating the nfProfile.

Calculating NF Profile Size

To calculate the size of the nfProfiles payload before registering or updating it, use the following steps:

1. Save the profile payload in a text file without spaces.
2. Verify the size of the nfProfile payload using Linux tools like wc.

For example, consider the below payload:

```
{"nfInstanceId":"ae137ab7-740a-46ee-
aa5c-951806d77b0d","nfType":"BSF","nfStatus":"REGISTERED","heartBeatTimer":30,
"plmnList":[{"mcc":"310","mnc":"14"}],"sNssais":[{"sd":"4eaaab","sst":124},
{"sd":"daaac8","sst":54},{"sd":"f4aaa6","sst":73}],"nsiList":
["slice-1","slice-2"],"fqdn":"BSF.oracle.com","interPlmnFqdn":"BSF.oracle.com"
,"ipv4Addresses":
["192.168.2.100","192.168.3.100","192.168.2.110","192.168.3.110"],"ipv6Address
es":
["2001:0db8:85a3:0000:0000:8a2e:0370:7334"],"capacity":2000,"load":0,"locality
":"US
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
      East","bsfInfo":{"ipv4AddressRanges":
[{"start":"192.168.0.0","end":"192.168.0.100"}],"ipv6PrefixRanges":
[{"start":"2001:db8:8513::/48","end":"2001:db8:8513::/96"}],"dnnList":
["abc","DnN-OrAcLe-32","ggsn-cluster-
A.mnc012.mcc345.gprs","lemon"]},"nfServiceList":{"ae137ab7-740a-46ee-
aa5c-951806d77b0d":{"serviceInstanceId":"ae137ab7-740a-46ee-
aa5c-951806d77b0d","serviceName":"nbsf-management","versions":
[{"apiVersionInUri":"v1","apiFullVersion":"1.0.0","expiry":"2018-12-03T18:55:0
8.871Z"}],"scheme":"http","nfServiceStatus":"REGISTERED","fqdn":"BSF.oracle.co
m","interPlmnFqdn":"BSF.oracle.com","apiPrefix":"","capacity":500,"load":0,"su
pportedFeatures":"80000000"}}}
```


For the above nfProfile, the size is calculated as 1177 bytes.

```
$ cat bsf.txt |wc -c
1177
```

> **Note**
>
> - The `{apiRoot}/nrf-configuration/v1/nfManagementOptions` API allows to configure the maximum size of the profile per nfType. If not configured for a specific nfType, the evaluation is performed using "`ALL_NF_TYPES`". This is a mandatory configuration.
> - The calculated total size of the `NfProfile` may vary due to updates made by the NRF. The size is determined based on the profile after the NRF has applied its modifications. For instance, the NRF may update certain attributes such as `expiry`, which follow a timestamp format.

Managing the feature

Enable

- **Enable Using REST API**: Set the value of `featureStatus` to *ENABLED* in the `{apiRoot}/nrf-configuration/v1/nfManagementOptions` URI. For more details, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable Using CNC Console**: Set the value of **Feature Status** to **ENABLED** in the NF Management Options.

Observability

Metrics

The `ocnrf_nfProfile_size_limit_breached` metric is added in the NRF NF Metrics section.

Alerts

There are no alerts introduced for this feature.

KPIs

The "NF Profile Size Limit Breached" KPI is added in the NRF Service KPIs section.

Maintain

Copyright © 2019, 2025, Oracle and/or its affiliates.

If you encounter alerts at system or application levels, see the NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

1. **Collect the logs**: For more information on collecting logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.3 Egress Gateway Pod Throttling

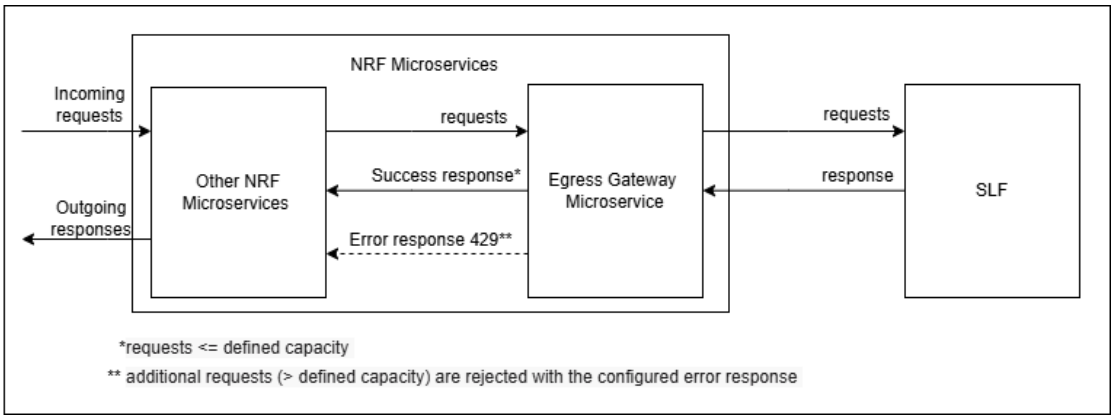
Egress Gateway microservice handles all the outgoing traffic from NRF to other NFs. In the event of unexpected high traffic, uneven traffic distribution, or traffic bursts due to network fluctuations, the Egress Gateway pods may get overloaded. This further may impact the system latency and performance. It is required to protect the Egress Gateway microservice from these conditions to prevent high impact on the outgoing traffic.

With the implementation of this feature, each Egress Gateway pod monitors its incoming traffic and if the traffic exceeds the defined capacity, the excess traffic is not processed and gets rejected. The traffic capacity is applied at each pod and applicable to all the incoming requests irrespective of the message type.

Egress Gateway Pod Throttling Mechanism

The following image describes the pod throttling mechanism when the requests received from NRF microservices like NFDiscovery, NFRegistration, NFSubscription, and NFAccessToken exceeds the defined capacity in Egress Gateway microservice.

Figure 4-3 Egress Gateway Pod Throttling Mechanism



1. NRF microservice sends the requests to the Egress Gateway microservice.
2. Egress Gateway microservice evaluates the number of requests received from the other NRF microservices:
   a. If the number of incoming requests are 2500 and the maximum defined limit is 2700, then Egress Gateway microservice processes all the 2500 requests.
   b. If the number of incoming requests are 3000 and the maximum defined limit is 2700, then Egress Gateway microservice processes only 2700 requests, and rejects the additional 300 requests with response code 429 to the NRF microservice.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Note: When NRF Discovery microservice receives an overload response from Egress Gateway microservice, it does not retry or re-reroute this request. An error response will be generated based on the specific feature configurations like NRF Forwarding, Subscriber Location Function, and Roaming.

Managing the Egress Gateway Pod Throttling

Enable

This feature can be enabled or disabled at the time of NRF deployment using the following Helm Parameters:

Perform the following configuration to enable this feature using the Helm:

1. Open the `ocnrf_custom_values_25.1.201.yaml` file.
2. Set the value of `egressgateway.podLevelMessageThrottling.enabled` to *true* under Egress Gateway Microservice parameters section.

   Note

   - This feature is enabled by default.
   - This feature also uses the following parameters which are read-only:
     - `egressgateway.podLevelMessageThrottling.duration`
     - `egressgateway.podLevelMessageThrottling.requestLimit`
     - `global.egwGeneratedErrorCheckDuringOverLoad`
     - `global.egwResponseCodesDuringOverLoad`
     - `global.noRetryDuringEgwOverload`

   For more information about the above parameters, see the "Egress Gateway Microservice" and "Global Parameters" sections in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
3. Save the file.
4. Install NRF. For more information about installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
5. Run Helm upgrade, if you are enabling this feature after NRF deployment. For more information about upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Disable

This feature can be disabled using the following Helm Parameter:

Perform the following configuration to disable this feature using Helm:

1. Open the `ocnrf_custom_values_25.1.201.yaml` file.
2. Set the value of `egressgateway.podLevelMessageThrottling.enabled` to *false* under the Egress Gateway Microservice parameters section. For more information about the parameter, see the "Egress Gateway Microservice" and "Global Parameters" sections in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

3. Save the file.
4. Install NRF. For more information about installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
5. Run Helm upgrade, if you are enabling this feature after NRF deployment. For more information about upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Observability

Metrics

Following metrics are added in the Egress Gateway Metrics section:

- `oc_egressgateway_podlevel_throttling_allowed_total`
- `oc_egressgateway_podlevel_throttling_discarded_total`

The `route_id` dimension is added in NRF Metrics.

Alerts

Following alerts are added in the Egress Gateway Pod Throttling section:

- OcnrfEgressPerPodDiscardRateAboveMajorThreshold
- OcnrfEgressPerPodDiscardRateAboveCriticalThreshold

For more information about alerts, see NRF Alerts section.

KPIs

Added the `Egress Gateway Pod Throttling` KPI in the Egress Gateway Pod Throttling section.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alert persists, perform the following:

1. **Collect the logs and Troubleshooting Scenarios**: For more information on how to collect logs and troubleshooting information, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.

2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.4 Traffic Segregation

This feature provides an option for traffic segregation at NRF based on traffic types. Within a Kubernetes cluster, traffic segregation can divide applications or workloads into distinct sections such as Operations Administration and Maintenance (OAM), Service Based Interface (SBI), Kubernetes control traffic, and so on. The Multus Container Network Interface (CNI) plugin for Kubernetes enables attaching multiple network interfaces to pods to help segregate traffic to/from NRF edge microservices like Ingress and Egress Gateway microservices.

This feature addresses the challenge of logically separating IP traffic of different traffic types, which are typically handled through a single network (Kubernetes overlay). The new

Copyright © 2019, 2025, Oracle and/or its affiliates.

functionality ensures that critical networks are not cross-connected or sharing the same routes, thereby preventing network congestion.

The feature requires configuration of separate networks, Network Attachment Definitions (NADs), and the Cloud Native Load Balancer (CNLB). These configurations are crucial for facilitating Ingress/Egress gateway traffic segregation, and optimizing load distribution in NRF.

> **Note**
>
> The Traffic Segregation feature is only available in NRF if OCCNE is installed with CNLB.

Cloud Native Load Balancer (CNLB)

CNE provides CNLB for managing the ingress and egress network as an alternate to the existing Load Balancing of Virtual Machine (LBVM), lb-controller, and egress-controller solutions. You can enable or disable this feature only during a fresh CNE installation. When this feature is enabled, CNE automatically uses CNLB to control ingress traffic. To manage the egress traffic, you must preconfigure the egress network details in the `cnlb.ini` file before installing CNE.

For more information about enabling and configuring CNLB, see *Oracle Communications Cloud Native Core, Cloud Native Environment User Guide, and Oracle Communications Cloud Native Core, Cloud Native Environment Installation, Upgrade, and Fault Recovery Guide*.

Network Attachment Definitions for CNLB

A Network Attachment Definition (NAD) is a resource used to set up a network attachment, in this case, a secondary network interface to a pod. NRF supports two types of CNLB NADs:

1. Ingress Network Attachment Definitions
   Ingress NADs are used to handle inbound traffic only. This traffic enters the CNLB application through an external interface service IP address and is routed internally using interfaces within CNLB networks.

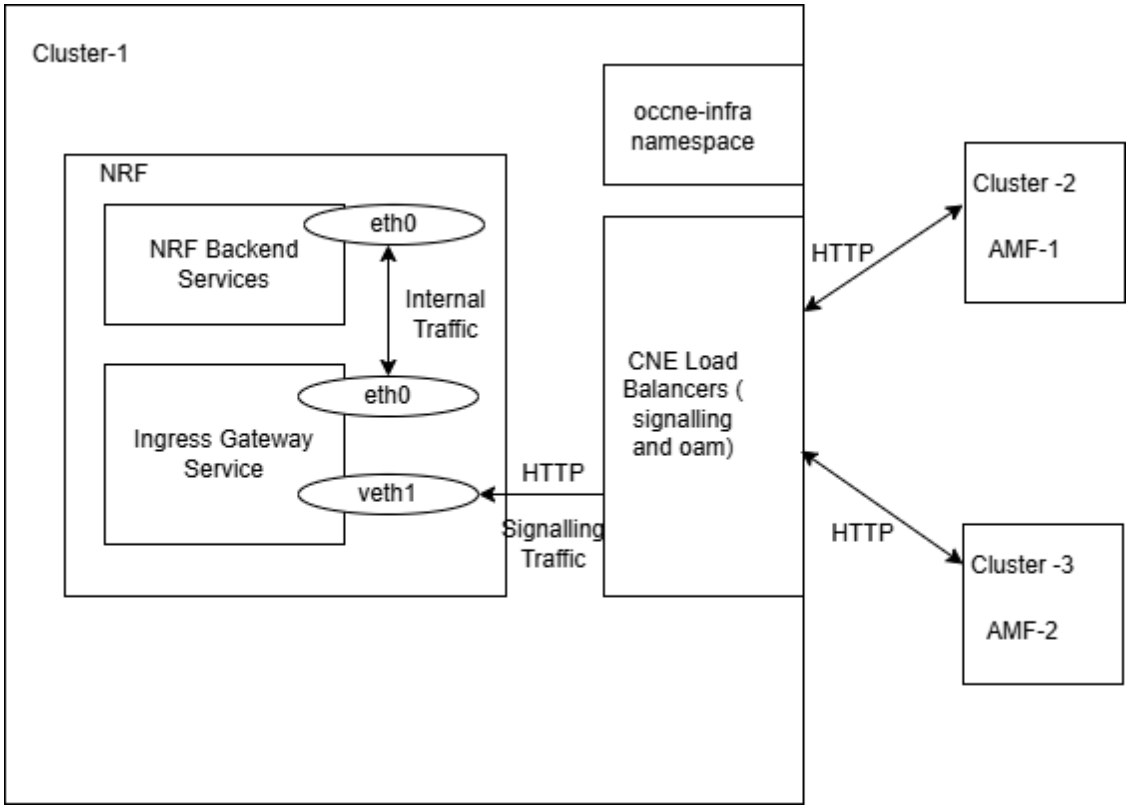
   Naming Convention: `nf-<service_network_name>-int`
2. Egress Only Network Attachment Definitions
   Egress Only NADs enable outbound traffic only. An NRF Egress Gateway pod can initiate traffic and route it through a CNLB application, translating the source IP address to an external egress IP address. An egress NAD contains network information to create interfaces for NRF Egress Gateway pods and routes to external subnets.
   - Prerequisites:
     - Ingress NADs are already created for the desired internal networks.
     - Destination (egress) subnet addresses are known beforehand and defined under the `cnlb.ini` file's `egress_dest` variable to generate NADs.
     - The use of an Egress NAD on a deployment can be combined with Ingress NADs to route traffic through specific CNLB apps.
   - Naming Convention: `nf-<service_network_name>-egr`

Ingress Traffic Segregation

The following image describes the traffic segregation at Ingress Gateway.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-4 Ingress Traffic Segregation



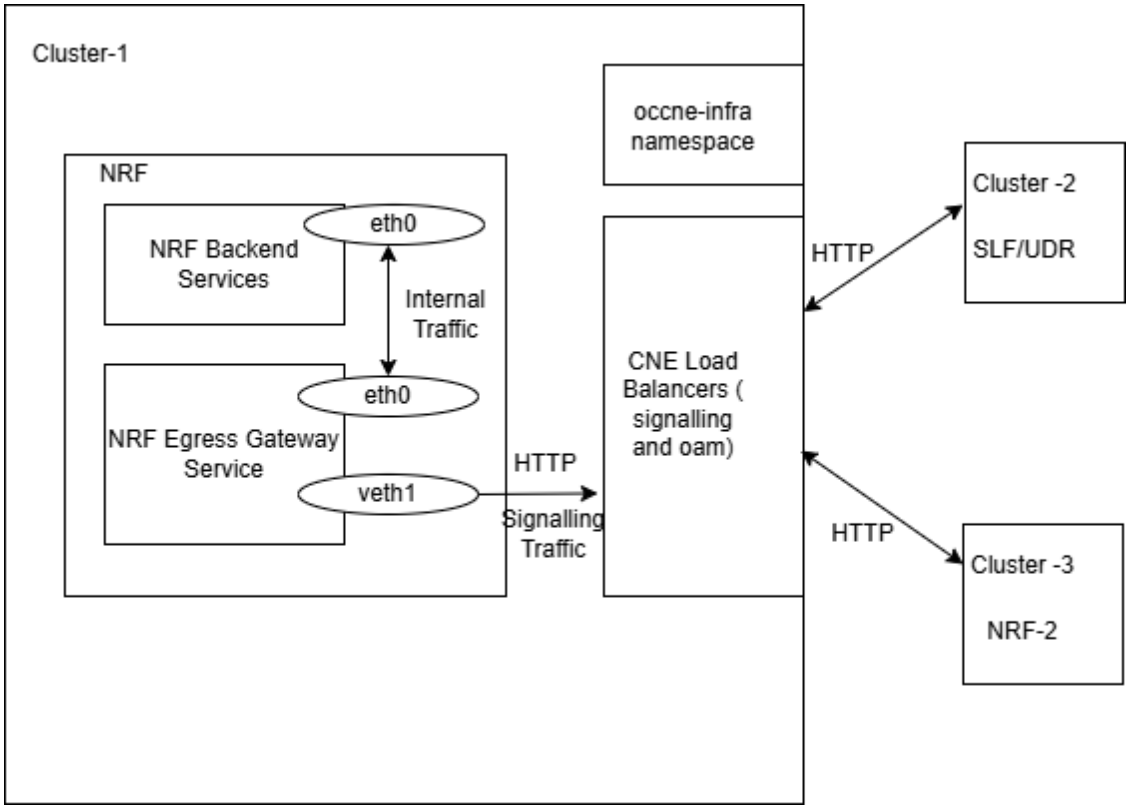
Here AMF-1 and AMF-2 send messages to the Ingress Gateway of the NRF. Internal traffic between NRF Backend and the Ingress Gateway travels through the default interface, eth0, which is connected to the pods. The signaling traffic between AMF-1 and NRF, as well as between AMF-2 and NRF, travels through a new interface, veth1, created using the Multus plugin. The Ingress service IPs and ports will be set up in Cloud Native Load Balancers (CNLBs), which will direct traffic to the Ingress Gateway. This gateway will then handle the requests and route them to the NRF.

Egress Traffic Segregation

The following image describes the traffic segregation at Egress Gateway.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-5 Egress Traffic Segregation



Here NRF is connected to SLF and another NRF, which are on different PLMNs and different networks. Internal traffic between NRF Backend and the NRF Egress Gateway passes through the default interface, eth0, which is connected to the pods. The signaling traffic between NRF and SLF, as well as between NRF and NRF-2, travels through a new interface, veth1, created using the Multus plugin. Egress IPs will be defined at the CNLBs, which will perform Source Network Address Translation (SNAT) on the IP addresses in requests sent to SLF and NRF-2.

Managing Traffic Segregation

Enable

This feature is disabled by default. To enable this feature, you need to configure the network attachment annotations in the `ocnrf_custom_values_25.1.201.yaml` file.

Configuration

For more information about Traffic Segregation configuration, see " Configuring Traffic Segregation" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Observe

There are no Metrics, KPIs, or Alerts available for this feature.

Maintain

To resolve any alerts at the system or application level, see NRF Alerts section. If the alerts persist, perform the following:

Copyright © 2019, 2025, Oracle and/or its affiliates.

1. Collect the logs: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. Raise a service request: See My Oracle Support for more information on how to raise a service request.

4.5 Support for Server Header

NRF handles various requests from consumer Network Functions (NFs) and other network entities over HTTP protocol. On receiving these requests, NRF validates and processes them before responding to these requests. When NRF sends an error response, it includes the source of the error to troubleshoot and take corrective measures.

This feature offers the support for Server Header in NRF responses, which contain information about the origin of an error response.

With this feature, NRF adds or passes the server header in the error responses as described in the following scenarios:

- When NRF acts as a standalone server.
- In NRF Forwarding scenarios, NRF propagates the server header received from peer NRF to the Consumer NF without any changes.
- In NRF Roaming scenarios, NRF propagates the server header received from SEPP or peer NRFs to the Consumer NF without any changes.

This enhancement improves NRF's error-handling scenarios by determining the originator of the error response for better troubleshooting.

When the server header feature is enabled, NRF adds the server header while generating the error responses.

The server header format is as follows:

```
<NRF>-<NrfInstanceId>
```

Where,

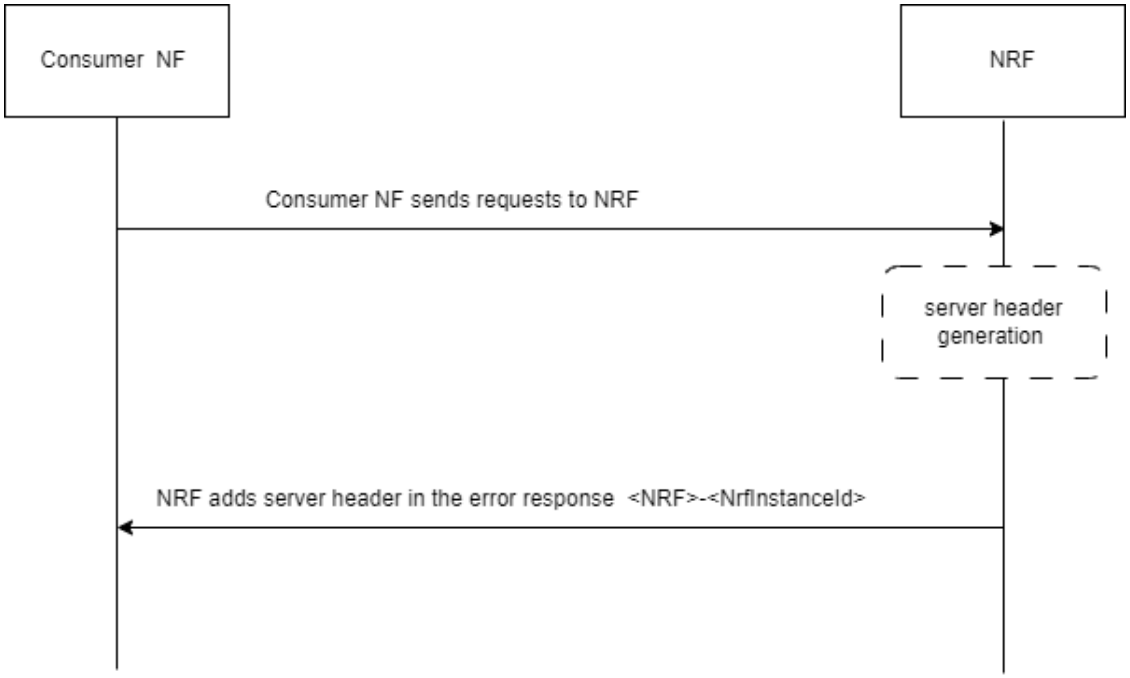
- `<NRF>` is the NF type.
- `<NrfInstanceId>` is the unique identifier of the NRF instance.

For example: `NRF-6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c`

The following image illustrates the NRF behavior while processing the error responses:

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-6 Server Header Details



1. Consumer or Client NF sends requests to NRF.
2. NRF receives the requests from the Consumer or Client NF and validates them for further processing.
3. In case, NRF sends an error response while processing the incoming requests, it adds the server header *{NRF-<NrfInstanceId>}* in the error response and sends it back to the Consumer or Client NF.

Managing Support for Server Header

Enable

This feature is disabled by default. This feature can be enabled at global level or at per-route level.

- **Enable using REST API**:
  - **Global**: Set the value of the `enabled` attribute to `true` in the *{apiRoot}/nrf/nf-common-component/v1/igw/serverheaderdetails* configuration API. For more information about the API, see "Server Header Details" section in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
  - **Route level**: Set the value of the `enabled` attribute to `true` in the *{apiRoot}/nrf/nf-common-component/v1/igw/routesconfiguration* configuration API. For more information about the API, see "Routes Configuration" section in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**:
  - **Global**: Switch **Enabled** on the Server Header Details page to enable the feature. For more information about enabling the feature using CNC Console, see Server Header Details.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- **Route level**: Switch **Enabled** on the Routes Configuration page to enable the feature. For more information about enabling the feature using CNC Console, see Routes Configuration.

For more information about server header propagation in service-mesh deployment, see Adding Filters for Server Header Propagation.

Configure

Perform the configurations in the following sequence to configure this feature using REST API or CNC Console:

1. Configure *{apiRoot}/nrf/nf-common-component/v1/igw/errorcodeserieslist* to update the errorcodeserieslist that is used to list the configurable exception or error for an error scenario in Ingress Gateway.
2. Perform the following global configuration in *{apiRoot}/nrf/nf-common-component/v1/igw/serverheaderdetails* API.
3. (Optional) Perform the following configuration at route level based on the `id` attribute in *{apiRoot}/nrf/nf-common-component/v1/igw/routesconfiguration* API.

> **Note**
>
> - If the feature is not configured at route level, global configuration is taken into consideration.
> - Route-level configuration will take precedence over the global configuration.

Observability

Metrics

There are no metrics added to this feature.

Alerts

There are no alerts added or updated for this feature.

KPIs

There are no KPIs related to this feature.

Maintain

If you encounter alerts at system or application levels, see the NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

1. **Collect the logs**: For more information on collecting logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.

2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

Copyright © 2019, 2025, Oracle and/or its affiliates.

4.6 Support for TLS

NRF use Hypertext Transfer Protocol Secure (HTTPS) to establish secured connections with Consumer NFs and Producer NFs, respectively. These communication protocols are encrypted using Transport Layer Security (TLS). TLS comprises the following components:

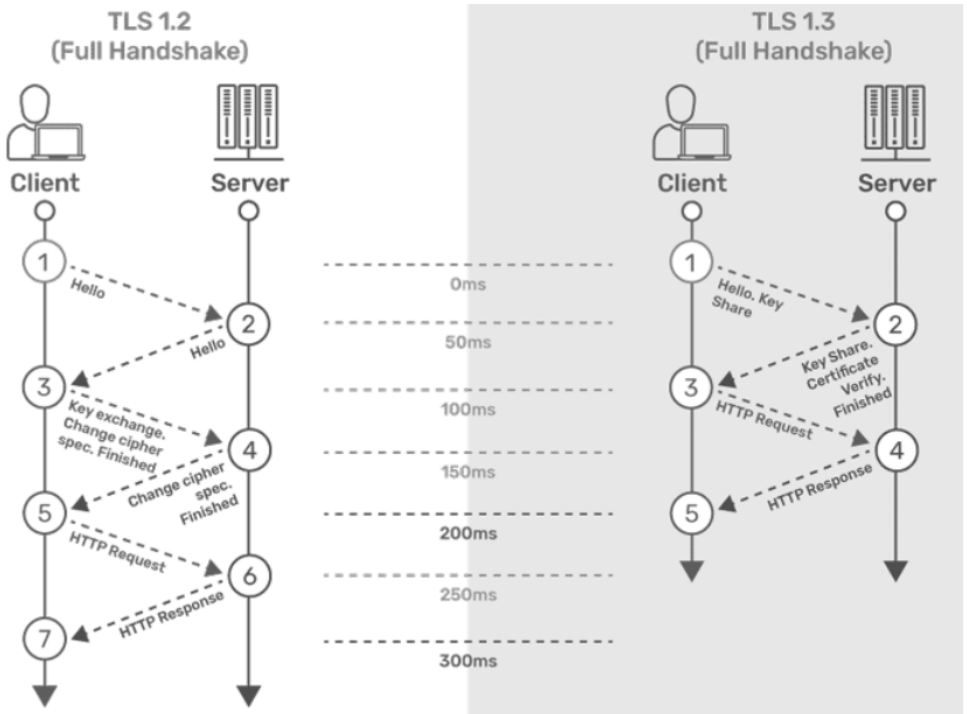
- **Handshake Protocol**: Exchanges the security parameters of a connection. Handshake messages are supplied to the TLS record layer.
- **Record Protocol**: Receives the messages to be transmitted, fragments the data into multiple blocks, secures the records, and then transmits the result. Received data is delivered to higher-level peers.

From Release 24.2.0 onwards, NRF supports TLS 1.3 for all Consumer NFs, Producer NFs, Data Director, SBI Interfaces, and interfaces where TLS 1.2 was supported. TLS 1.2 will continue to be supported.

TLS Handshake

This section describes the differences between TLS 1.2 and TLS 1.3 and the advantages of TLS 1.3 over TLS 1.2 and earlier versions.

Figure 4-7 TLS Handshake



TLS 1.2
Step 1: The connection or handshake starts when the client sends a “client hello” message to the server. This message consists of cryptographic information such as supported protocols and supported cipher suites. It also contains a random value or random byte string.

Step 2: To respond to the “client hello” message, the server sends the “server hello” message. This message contains the CipherSuite that the server has selected from the options provided

Copyright © 2019, 2025, Oracle and/or its affiliates.

by the client. The server also sends its certificate along with the session ID and another random value.

Step 3: The client verifies the certificate sent by the server. When the verification is complete, it sends a byte string and encrypts it using the public key of the server certificate.

Step 4: When the server receives the secret, both the client and server generate a master key along with session keys (ephemeral keys). These session keys are used to symmetrically encrypt the data.

Step 5: The client sends an “HTTP Request” message to the server to enable the server to switch to symmetric encryption using the session keys.

Step 6: To respond to the client’s “HTTP Request” message, the server does the same and switches its security state to symmetric encryption. The server concludes the handshake by sending an HTTP response.

Step 7: The client-server handshake is completed in two round-trips.

TLS 1.3
Step 1: The connection or handshake starts when the client sends a “client hello” message to the server. The client sends the list of supported cipher suites. The client also sends its key share for that particular key agreement protocol.

Step 2: To respond to the “client hello” message, the server sends the key agreement protocol that it has chosen. The “Server Hello” message comprises the server key share, server certificate, and the “Server Finished” message.

Step 3: The client verifies the server certificate, generates keys as it has the key share of the server, and sends the “Client Finished” message along with an HTTP request.

Step 4: The server completes the handshake by sending an HTTP response.

The following digital signature algorithms are supported in TLS handshake:

Table 4-3 Digital Signature Algorithms

Algorithm	Key Size (Bits)	Elliptic Curve (EC)
RS256 (RSA)	2048	NA
	4096 This is the recommended value.	NA
ES256 (ECDSA)	NA	SECP384r1 This is the recommended value.

Comparison Between TLS 1.2 and TLS 1.3

The following table provides a comparison of TLS 1.2 and TLS 1.3:

Table 4-4 Comparison of TLS 1.2 and TLS 1.3

Feature	TLS 1.2	TLS 1.3
TLS Handshake	• The initial handshake was carried out in clear text. • A typical handshake in TLS 1.2 involves the exchange of 5 to 7 packets.	• The initial handshake is carried out along with the key share. • A typical handshake IN TLS 1.3 involves the exchange of up to 3 packets.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-4 (Cont.) Comparison of TLS 1.2 and TLS 1.3

Feature	TLS 1.2	TLS 1.3
Cipher Suites	• Less secure Cipher suites. • Use SHA-256 and SHA-384 hashing – TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 – TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 – TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 – TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 – TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256	• More secure Cipher suites. • Apart from all the ciphers supported for TLS 1.2 , the following additional ciphers are supported for only TLS 1.3 : – TLS_AES_128_GCM_SHA256 – TLS_AES_256_GCM_SHA384 – TLS_CHACHA20_POLY1305_SHA256
Round-Trip Time (RTT)	This has a high RTT during the TLS handshake.	This has low RTT during the TLS handshake.
Perfect Forward Secrecy (PFS)	This doesn't support PFS.	TLS 1.3 supports PFS. PFS ensures that each session key is completely independent of long-term private keys, which are keys that are used for an extended period to decrypt encrypted data.
Privacy	This is less secure, as the ciphers used are weak.	This is more secure, as the ciphers used are strong.
Performance	This has high latency and a less responsive connection.	This has low latency and a more responsive connection.

Advantages of TLS 1.3

TLS 1.3 handshake offers the following improvements over earlier versions:

- all handshake messages after the ServerHello are encrypted.
- improves efficiency in the handshake process by requiring fewer round trips than TLS 1.2. It also uses cryptographic algorithms that are faster.
- better security than TLS 1.2. It addresses known vulnerabilities in the handshake process.
- got rid of data compression.

The following table describes the TLS versions supported in the client and server side. The last column indicates which version will be used.

TLS Version Used
When NRF is acting as a client or a server, it can have different TLS versions.

The following table gives information about which TLS version will be used when you have various combinations of TLS versions between the server and the client.

Table 4-5 TLS Version Used

Client Support	Server Support	TLS Version Used
TLS 1.2, TLS 1.3	TLS 1.2, TLS 1.3	TLS 1.3
TLS 1.3	TLS 1.3	TLS 1.3
TLS 1.3	TLS 1.2, TLS 1.3	TLS 1.3

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-5 (Cont.) TLS Version Used

Client Support	Server Support	TLS Version Used
TLS 1.2, TLS 1.3	TLS v1.3	TLS 1.3
TLS 1.2	TLS 1.2, TLS 1.3	TLS 1.2
TLS 1.2, TLS 1.3	TLS 1.2	TLS 1.2
TLS 1.3	TLS 1.2	Sends an error message. For more information about the error message, see "Troubleshooting TLS Version Compatibilities" section in *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
TLS 1.2	TLS 1.3	Sends an error message. For more information about the error message, see "Troubleshooting TLS Version Compatibilities" section in *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.

Note

- If Egress Gateway is deployed with both the versions of TLS that is TLS 1.2 and TLS 1.3, then Egress Gateway as client will send both versions of TLS in the client hello message during the handshake and the server needs to decide which version to be used.
- If Ingress Gateway is deployed with both the version of TLS that is with TLS 1.2 and TLS 1.3, then Ingress Gateway as the server will use the TLS version received from the client in the server hello message during the handshake.
- This feature does not work in ASM deployment.

Managing Support for TLS 1.2 and TLS 1.3

Enable

This feature can be enabled or disabled at the time of NRF deployment using the following Helm Parameters:

- **enableIncomingHttps** - This flag is used for enabling/disabling HTTPS/2.0 (secured TLS) in the Ingress Gateway. If the value is set to false, NRF will not accept any HTTPS/2.0 (secured) traffic. If the value is set to true, NRF will accept HTTPS/2.0 (secured) Traffic. **Note:** Do not change `&enableIncomingHttpsRef` reference variable.

  For more information on enabling this flag, see the "Enabling HTTPS at Ingress Gateway" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
- **enableOutgoingHttps** -This flag is used for enabling/disabling HTTPS/2.0 (secured TLS) in the Egress Gateway. If the value is set to false, NRF will not accept any HTTPS/2.0 (secured) traffic. If the value is set to true, NRF will accept HTTPS/2.0 (secured) traffic.

Copyright © 2019, 2025, Oracle and/or its affiliates.

For more information on enabling this flag, see the "Enabling HTTPS at Egress Gateway" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide.*

Configure

You can configure this feature using Helm parameters.

The following parameters in the Ingress Gateway and Egress Gateway microservices must be customized to support TLS 1.2 or TLS 1.3.

1. Generate HTTPS certificates for both the Ingress and Egress Gateways. Ensure that the certificates are correctly configured for secure communication. After generating the certificates, create a Kubernetes secret for each Gateway (Ingress and Egress). Then, configure these secrets to be used by the respective Gateways. For more information about HTTPS configuration, generating certificates, and creating secrets, see the "Configuring Secrets for Enabling HTTPS" section in the *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
2. Open the `ocnrf_custom_values_25.1.201.yaml` file.
3. Configure the following parameters under Ingress Gateway and Egress Gateway parameters section.
   - Parameters required to support TLS 1.2:
     - `service.ssl.tlsVersion` indicates the TLS version.
     - `cipherSuites` indicates supported cipher suites.
     - `allowedCipherSuites` indicates allowed cipher suites.
     - `messageCopy.security.tlsVersion` indicates the TLS version for establishing communication between Kafka and NF when security is enabled.
   - Parameters required to support TLS 1.3:
     - `service.ssl.tlsVersion` indicates the TLS version.
     - `cipherSuites` indicates the supported cipher suites.
     - `allowedCipherSuites` indicates the allowed cipher suites.
     - `messageCopy.security.tlsVersion` indicates the TLS version for establishing communication between Kafka and NF when security is enabled.
     - `clientDisabledExtension` is used to disable the extension sent by messages originated by clients during the TLS handshake with the server.
     - `serverDisabledExtension` is used to disable the extension sent by messages originated by servers during the TLS handshake with the client.
     - `tlsNamedGroups` is used to provide a list of values sent in the supported_groups extension. These are comma-separated values.
     - `clientSignatureSchemes` is used to provide a list of values sent in the signature_algorithms extension.

   For more information about configuring the values of the above-mentioned parameters, see the "Ingress Gateway Microservice" and "Egress Gateway Microservice" sections in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
4. Save the file.

Copyright © 2019, 2025, Oracle and/or its affiliates.

5. Install NRF. For more information about installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
6. Run Helm upgrade, if you are enabling this feature after NRF deployment. For more information about upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

> **Note**
>
> - NRF does not prioritize cipher suites based on priorities. To select a cipher based on priorities, you must write the cipher suites in decreasing order of priority.
> - NRF does not prioritize supported groups based on priorities. To select a supported group based on priorities, you must write the supported group values in the decreasing order of priority.
> - If you want to provide values for the signature_algorithms extension using the **clientSignatureSchemes** parameter, the following comma-separated values must be provided to deploy the services:
>   - rsa_pkcs1_sha512
>   - rsa_pkcs1_sha384
>   - rsa_pkcs1_sha256
> - The mandatory extensions as listed in RFC 8446 cannot be disabled using the **clientDisabledExtension** attribute on the client or using the **serverDisabledExtension** attribute on the server side. The following is the list of the extensions that cannot be disabled:
>   - supported_versions
>   - key_share
>   - supported_groups
>   - signature_algorithms
>   - pre_shared_key

Observe
Metrics

Following metrics are added in the NRF Gateways Metrics section:

- `oc_ingressgateway_incoming_tls_connections`
- `oc_egressgateway_outgoing_tls_connections`
- `security_cert_x509_expiration_seconds`

For more information about metrics, see NRF Metrics section.

Alerts

Following alerts are added in the NRF Alerts section:

- OcnrfTLSCertificateExpireMinor
- OcnrfTLSCertificateExpireMajor
- OcnrfTLSCertificateExpireCritical

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> Alert gets raised for every certificate that will expire in the above time frame. For example, NRF supports both RSA and ECDSA. So, we have configured two certificates. Accordingly, let us suppose RSA certificate is about to expire in 6 months in this situation only one alert will be raised and if both are about to expire then two alerts will be raised. Moreover, alerts can be differentiated using "serialNumber" tag. For example, serialNumber=4661 is used for RSA and serialNumber =4662 is used for ECDSA.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alert persists, perform the following:

1. **Collect the logs and Troubleshooting Scenarios**: For more information on how to collect logs and troubleshooting information, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.7 Error Messages Enhancement

NRF sends error messages when there is an error scenario while performing any of the service operations. In order to identify the cause of these errors and fix them quickly, NRF supports the following enhancements:

- Error Response Enhancement for NRF
- Error Log Messages Enhancement

4.7.1 Error Response Enhancement for NRF

NRF receives the message requests from Consumer NFs and may send error responses while processing them. The error response generated by NRF included the error description in the `detail` attribute of the `ProblemDetails`. This description is used to identify the issue and troubleshoot them.

The sample `ProblemDetails` before the error details enhancement is as follows:

```
{
 “title”: “Not Found”,
 “status”: 404,
 “detail”: “NRF Forwarding Error Occurred”,
 “cause”: “UNSPECIFIED_MSG_FAILURE”
}
```

With the enhanced error response mechanism, NRF sends additional information such as server FQDN, NF service name, vendor NF name, and application error Id in the `detail` attribute of the `ProblemDetails`. This enhancement provides more information about the error and troubleshoot them.

Copyright © 2019, 2025, Oracle and/or its affiliates.

The new format of the ProblemDetails is as follows:

```
{
“title”:“<title of Problem Details>”,
“status”:“<status code of Problem Details>”,
“detail”:“<Server-FQDN>:<NF Service>:<Application specific problem
detail>:<VendorNF>-<App-Error-Id>“,
“cause”:“<cause value of Problem Details>”
}
```

Example:

```
{
 “title”: “Not Found”,
 “status”: 404,
 “detail”: “NRF-d5g.oracle.com: Nnrf_NFManagement: NRF Forwarding Error
Occurred: ONRF-REG-NFPRFWD-E0299; 404; NRF-d5g.oracle.com: Nnrf_NFManagement:
Could not find Profile records for NfInstanceId=8cfc6828-
bd5d-4a3a-93d4-6bdd848d6bab: ONRF-REG-NFPR-E1004”,
 “cause”: “UNSPECIFIED_MSG_FAILURE”
}
```

The updated format for the detail attribute will be sent only for the error responses generated by NRF application. For all other scenarios (like response generated by the underlying stack) the detail attribute will not provide the information in the updated format.

> **Note**
>
> All responses sent by NRF application towards the peer follows this format. All internal microservice responses and OAM responses may or may not follow this format.

The following table describes the detail attribute of the ProblemDetails parameter:

Table 4-6 Detail Attributes of the ProblemDetails

Attribute	Description
<Server-FQDN>	Indicates the NRF FQDN. It is obtained from the nfFqdn Helm parameter. **Sample Value**: NRF-d5g.oracle.com

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-6 (Cont.) Detail Attributes of the ProblemDetails

Attribute	Description
<NF Service>	Indicates the service name. This service name can be 3GPP defined NRF service name or an internal one. **Possible Values:** • Nnrf_NFManagement: This is used by nfregistration, nfsubscription, and nrfauditor microservices. • Nnrf_NFDiscovery: This is used by nfdiscovery microservice. • Nnrf_AccessToken: This is used by nfaccesstoken microservice. • Nnrf_Internal_CDS: This is used by nrfcachedata microservice. • Nnrf_Internal_Artisan: This is used by nrfauditor microservice. • Nnrf_Internal_Config: This is used by nrfconfiguration microservice for NRF specific configurations. • ingressgateway: This is used by ingressgateway. • egressgateway: This is used by egressgateway. • alternateroute: This is used by alternate-route. • commonconfig: This is used by nrfconfiguration microservice for NRF common configurations such as ingressgateway, egressgateway.
<Application specific problem detail>	Indicates a short description of the error. **Sample Value:** NRF Forwarding Error occurred
<VendorNF>	Indicates the Oracle NF Type. For NRF, this value is always ONRF. Here, O is the prefixed value. NF type is obtained from the nfType Helm parameter.
<App-Error-Id>	Indicates the application specific error Id. This Id is a unique error Id which includes the Microservice ID, Category, and the Error Code. The format of <App-Error-Id> is as follows: <Microservice ID>-<Category>-<Error Code> Where, • <Microservice ID>: Indicates the NRF microservice IDs. For more information of NRF Microservice IDs see the Table 4-7. **Sample Value:** REG is the Microservice ID for nfregistration. • <Category>: Indicates an unique grouping of errors. The presence of category is optional in application error code. For detailed information about the different categories available, see the Table 4-8. **Sample Value:** NFPRFWD for NFProfileRetrieval service operation during forwarding. • <Error Code>: Indicates the error code in the Exxxx format. **Sample Value**: E0299 **Sample Value**: • With category: REG-NFPRFWD-E0299 where, REG is the Microservice ID, NFPRFWD is the Category, and E0299 is the Error Code. • Without category: ONRF-IGW-E002 where, ONRF is the vendor NF, IGW is the Microservice ID, and E002 is the Error Code.

The following table describes the list of available microservices in NRF.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-7 Microservice Ids List

Microservice Ids	Microservice Name
REG	nfregistration
SUB	nfsubscription
DIS	nfdiscovery
ACC	nfaccesstoken
CDS	nrfcachedata
AUD	nrfauditor
CFG	nrfconfiguration
ART	nrfartisan
IGW	ingressgateway
EGW	egressgateway
ARS	alternate-route

The following table describes the list of available categories in NRF.

Table 4-8 List of Categories

Category Id	Category Description
ACAUTHN	Authentication during Access Token service operation.
ACAUTHZ	Authorization during Access Token service operation.
ACCOPT	Access Token configuration.
ACFWD	Access Token service operation during forwarding.
ACROAM	Access Token service operation during roaming.
ACTOK	Access Token service operation.
AUTHOPT	Authentication options configuration.
CNCCDTN	CNCC Controller configuration.
CSHTOPT	Control shutdown options configuration.
DEREG	NFDeregister service operation.
DISAUTH	NFDiscover service operation during authorization.
DISC	NFDiscover service operation.
DISCOPT	Discovery options configuration.
DISCSLF	SLF for NFDiscover.
DISCFWD	NFDiscover service operation during forwarding.
DISROAM	NFDiscover service operation during roaming.
FULLUPD	NFUpdate with Full Replacement (PUT) service operation.
FWDOPT	Forwarding options configuration.
GENOPT	General options configuration.
GROPT	GeoRedundancy options configuration.
GTSUB	Get subscription.
GWTHFOR	Growth configuration for forwarding.
GWTHOPT	Growth options configuration.
HB	NFHeartbeat service operation.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-8 (Cont.) List of Categories

Category Id	Category Description
HBAUD	Heart beat for NRF Auditor service operation.
HBRMAUD	Remote heart beat for NRF Auditor.
INVSROP	UnSupported HTTP Method for service operation.
LOGOPT	Logging options configuration.
MGMTOPT	Management options configuration.
NFDTLS	Fetching NF State data
NFINSTS	Get remote discovery instance and remote instance data.
NFLR	NFListRetrieval service operation.
NFPR	NFProfileRetrieval service operation.
NFPRAUD	NRF Auditor service operation for NF profiles.
NFPRFWD	NFProfileRetrieval service operation during forwarding.
NPTRDRG	NRF Artisan for notification event to DNS NAPTR on deleting AMF profile.
NPTROPT	NAPTR options configuration.
NPTRREC	Status API for DNS NAPTR record.
NPTRREG	NRF Artisan for NAPTR registration.
NPTRUPD	NRF Artisan for DNS NAPTR update.
NRFCFG	Fetching NRF Configuration.
NRFOPER	NRF operation state configuration.
PARTUPD	NFUpdate with Partial Replacement (PATCH) service operation.
POPROPT	Pod protection options configuration.
REGN	NFRegister service operation.
RELDVER	NRF Auditor service operation for reloading site and network version info.
ROAMOPT	Roaming options configuration.
RTNPTR	Manual cache update of DNS NAPTR records.
SCRNOPT	Screening options configuration.
SCRNRUL	Screening rules configuration.
SLFOPT	SLF options configuration.
SNOTIF	NFStatusNotify service operation.
SUBAUD	NRF Auditor service operation for subscription.
SUBDTLS	Retrieve subscription details based on query attributes for configuring the non-signaling API.
SUBFWD	NFStatusSubscribe service operation during forwarding.
SUBRAUD	NRF Auditor service operation for remote subscription.
SUBROAM	NFStatusSubscribe service operation during roaming.
SUBSCR	NFStatusSubscribe service operation.
UNKSROP	UnSupported media type.
UNSFWD	NFStatusUnsubscribe service operation during forwarding.
UNSROAM	NFStatusUnsubscribe service operation during roaming.
UNSUBSC	NFStatusUnsubscribe service operation.
UPDSUBS	NFStatusSubscribe update (PATCH) service operation.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-8 (Cont.) List of Categories

Category Id	Category Description
UPSFWD	NFStatusSubscribe update (PATCH) service operation during forwarding.
UPSROAM	NFStatusSubscribe update (PATCH) service operation during roaming.

The error response codes are available in the Error Response Code Detailssection.

Managing the Error Response Enhancements for NRF

This section explains the procedure to enable and configure the feature.

Enable

This feature is an enhancement to the existing `detail` attribute in the `ProblemDetails` and hence there is no specific flag to enable or disable this feature.

Configure

You can configure the feature using Helm.

Helm

Perform the following configuration for this feature using the Helm:

1. Open the `ocnrf_custom_values_25.1.201.yaml` file.
2. Configure the following parameters under global parameters:
   - `global.nfFqdn` parameter to indicate the NRF FQDN.
   - `global.nfType` parameter to indicate the NF type.
   - `global.maxDetailsLength` is used to determine the maximum length limit of compiled error string by NRF in the details field.
3. Save the file.
4. Install NRF. For more information about installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
5. Run Helm upgrade, if you are enabling this feature after NRF deployment. For more information about upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Observability

Metrics

There are no new metrics added to this feature.

Alerts

There are no alerts added or updated for this feature.

KPIs

There are no new KPIs related to this feature.

Maintain

Copyright © 2019, 2025, Oracle and/or its affiliates.

If you encounter alerts at system or application levels, see the NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

1. Collect the logs: For more information on collecting logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.

2. Raise a service request: See My Oracle Support for more information on how to raise a service request.

4.7.2 Error Log Messages Enhancement

NRF uses the logs to register the system events along with their date and time of occurrence. They also provide important details about the chain of events that could have led to an error or problem. These log details are further used to identify the source of the error and troubleshoot them. NRF supports various log levels that can be set for a microservice with any of the following values:

- TRACE
- DEBUG
- INFO
- WARN
- ERROR

For more information on details of these log levels, see the “Log Levels” section in *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.

With this feature, NRF adds additional information to the existing “ERROR” log messages. This additional information can provide more details about the error which can help to identify the problem details, error generating entity and subscriber information.

The Error Log Messages Enhancement feature adds the following additional attributes to the existing "ERROR" logs with appropriate values during the failure scenarios:

- `errorStatus`
- `errorTitle`
- `errorDetails`
- `errorCause`
- `sender`
- `receiver`
- `subscriberId`

For more information about the new attributes, see Table 4-9.

Sample enhanced log details for the NRF Subscription Microservice:

```
{
  "instant": {
    "epochSecond": 1717077558,
    "nanoOfSecond": 256932930
  },
  "thread": "boundedElastic-8",
  "level": "ERROR",
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
  "loggerName": "com.oracle.cgbu.cne.nrf.routes.SubscriptionHandler",
  "message": "Response sent: 500 Subscription global limit breached for uri :
http://ocnrf-ingressgateway.nrf1-ns/nnrf-nfm/v1/subscriptions with the
problem cause : INSUFFICIENT_RESOURCES and problem details : NRF-
d5g.oracle.com: Nnrf_NFManagement: Subscription global limit breached: ONRF-
SUB-SUBSCR-E2003",
  "endOfBatch": false,
  "loggerFqcn": "org.apache.logging.log4j.spi.AbstractLogger",
  "threadId": 39547,
  "threadPriority": 5,
  "messageTimestamp": "2024-05-30T13:59:18.256+0000",
  "configuredLevel": "WARN",
  "subsystem": "NfSubscribe",
  "processId": "1",
  "nrfTxId": "nrf-tx-2103278974",
  "ocLogId": "1717077558225_77_ocnrf-
ingressgateway-854464d548-426bz:1717077558237_110_ocnrf-
nfsubscription-766d45f5c7-t4xp8",
  "xRequestId": "",
  "numberOfRetriesAttempted": "",
  "hostname": "ocnrf-nfsubscription-766d45f5c7-t4xp8",
  "errorStatus": "500",
  "errorTitle": "Subscription global limit breached",
  "errorDetails": "NRF-d5g.oracle.com: Nnrf_NFManagement: Subscription global
limit breached: ONRF-SUB-SUBSCR-E2003",
  "errorCause": "INSUFFICIENT_RESOURCES",
  "sender": "NRF-6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c",
  "subscriberId": "imsi-345012123123126"
}
```

In the above example, the following new attributes are added to the error logs.

```
  "errorStatus": "500",
  "errorTitle": "Subscription global limit breached",
  "errorDetails": "NRF-d5g.oracle.com: Nnrf_NFManagement: Subscription global
limit breached: ONRF-SUB-SUBSCR-E2003",
  "errorCause": "INSUFFICIENT_RESOURCES",
  "sender": "NRF-6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c",
  "subscriberId": "imsi-345012123123126"
```

The `additionalErrorLogging` feature flag is added to the logging API for the supported microservices. When the feature flag is enabled, it adds the `errorStatus`, `errorTitle`, `errorDetails`, `errorCause`, `sender`, `receiver` attributes to the ERROR logs.

Additionally, the `logSubscriberInfo` flag is added to the logging API for the supported microservices. When `additionalErrorLogging` and `logSubscriberInfo` are enabled, the subscriber information is also added to the ERROR logs.

When the `logSubscriberInfo` flag is enabled and if the `subscriberId` information is available, then NRF adds the `subscriberId` value in the log messages. If `logSubscriberInfo` is disabled, and the `subscriberId` information is still available NRF adds the `subscriberId` value as XXXX in the log messages.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> The `subscriberId` value can be any of the following:
>
> - 3GPP-Sbi-Correlation-Info header present in the request header received at NRF.
> - 3GPP-Sbi-Correlation-Info header present in the response header received from the peer in the case of NRF acting as a client.
> - Either SUPI or GPSI values or both SUPI and GPSI values.
>
> For more information about the `subscriberId` attribute, see Table 4-9.

Example with both SUPI and GPSI values:

```
{
  "instant": {
    "epochSecond": 1717489820,
    "nanoOfSecond": 998846286
  },
  "thread": "@608f310a-49",
  "level": "ERROR",
  "loggerName": "com.oracle.cgbu.cne.nrf.service.NFDiscoveryService",
  "message": "Response sent: 503 null for uri : /nnrf-disc/v1/nf-instances?
{target-nf-type=[UDR], requester-nf-type=[UDM], supi=[imsi-10000005],
gpsi=[msisdn-77195225555]} with the problem cause : null and problem
details : NRF-d5g.oracle.com: Nnrf_NFDiscovery: SLF Lookup Error Occurred:
ONRF-DIS-DISCSLF-E0499; 503; null; 503; null",
  "endOfBatch": false,
  "loggerFqcn": "org.apache.logging.log4j.spi.AbstractLogger",
  "threadId": 49,
  "threadPriority": 5,
  "messageTimestamp": "2024-06-04T08:30:20.998+0000",
  "configuredLevel": "WARN",
  "processId": "1",
  "nrfTxId": "nrf-tx-2050587612",
  "ocLogId": "1717489820750_308_ocnrf-
ingressgateway-54594dfd87-9fq2k:1717489820763_73_ocnrf-
nfdiscovery-744ff58c64-5cdh9",
  "serviceOperation": "NFDiscover",
  "xRequestId": "",
  "requesterNfType": "UDM",
  "targetNfType": "UDR",
  "discoveryQuery": "/nnrf-disc/v1/nf-instances?{target-nf-type=[UDR],
requester-nf-type=[UDM], supi=[imsi-10000005], gpsi=[msisdn-77195225555]}",
  "hostname": "ocnrf-nfdiscovery-744ff58c64-5cdh9",
  "subsystem": "discoveryNfInstances",
  "errorStatus": "503",
  "errorDetails": "NRF-d5g.oracle.com: Nnrf_NFDiscovery: SLF Lookup Error
Occurred: ONRF-DIS-DISCSLF-E0499; 503; null; 503; null",
  "sender": "NRF-6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c",
  "subscriberId": "imsi-10000005;msisdn-77195225555"
}
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> The log with updates attributes will be printed only for the error responses generated by NRF application. For all other scenarios (like error responses generated by the underlying stack) the log with updated attributes will not be printed.

Mapping Error Log Attributes

The following table explains the mapping of the new attributes with the existing attributes of `ProblemDetails`, `subscriberId`, `sender`, and `receiver` in the error logs.

Table 4-9 Mapping Error Log Attributes

New Attribute in Logs	Value when NRF acting as Standalone Server (Example: NFRegister, NFUpdate, NFListRetrieval, NFProfileRetrieval service operations and so on)	Value when NRF is acting as Standalone Client (Example: NfStatusNotify service operation)	Value when NRF acting as both Server and Client (Example: NRF Forwarding, Roaming, SLF)	
			As Server	As Client
`errorStatus`	`status` sent by NRF in `ProblemDetails` of HTTP response	`status` received byNRF in `ProblemDetails` of HTTP response	`status` sent by NRF in `ProblemDetails` of HTTP response	`status` received by NRF in `ProblemDetails` of HTTP response
`errorTitle`	`title` sent by NRF in `ProblemDetails` of HTTP response	`title` received by NRF in `ProblemDetails` of HTTP response	`title` sent by NRF in `ProblemDetails` of HTTP response	`title` received by NRF in `ProblemDetails` of HTTP response
`errorDetails`	`detail` sent by NRF in `ProblemDetails` of HTTP response	`detail` received by NRF in `ProblemDetails` of HTTP response	`detail` sent by NRF in `ProblemDetails` of HTTP response	`detail` received by NRF in `ProblemDetails` of HTTP response
`errorCause`	`cause` sent by NRF in `ProblemDetails` of HTTP response	`cause` received by NRF in `ProblemDetails` of HTTP response	`cause` sent by NRF in `ProblemDetails` of HTTP response	`cause` received by NRF in `ProblemDetails` of HTTP response
`sender`	NRF-NRF Instance Id	Server header value as received in the error response.	NRF-NRF Instance Id	Server header value as received in the error response
`receiver`	Not applicable	NRF-NRF Instance Id	Not applicable	NRF-NRF Instance Id

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-9 (Cont.) Mapping Error Log Attributes

New Attribute in Logs	Value when NRF acting as Standalone Server (Example: NFRegister, NFUpdate, NFListRetrieval, NFProfileRetrieval service operations and so on)	Value when NRF is acting as Standalone Client (Example: NfStatusNotify service operation)	Value when NRF acting as both Server and Client (Example: NRF Forwarding, Roaming, SLF)	
			As Server	As Client
subscriberId	The priority is considered as follows: 1. if the value of *3GPP-Sbi-Correlation-Info* header is present in the request. 2. else, only in case of NFDiscover Service operation request, the derived values from SUPI/GPSI query parameters if present.	The priority is considered as follows: 1. if the value of *3GPP-Sbi-correlation-Info* header is present in the response received when NRF is acting as a client. 2. else, if the value of *3GPP-Sbi-correlation-Info* header is present in the request. 3. else, only in case of NFDiscover Service operation request, the derived values from SUPI/GPSI query parameters if present.	The priority is considered as follows: 1. if the value of *3GPP-Sbi-correlation-Info* header is present in the response received when NRF is acting as a client. 2. else, if the value of *3GPP-Sbi-correlation-Info* header is present in the request. 3. else, only in case of NFDiscover Service operation request, the derived values from SUPI/GPSI query parameters if present.	The priority is considered as follows: 1. if the value of *3GPP-Sbi-correlation-Info* header is present in the response received when NRF is acting as a client. 2. else, if the value of *3GPP-Sbi-correlation-Info* header is present in the request. 3. else, only in case of NFDiscover Service operation request, the derived values from SUPI/GPSI query parameters if present.

> **Note**
>
> In case, if any of the above newly added attributes does not have a value, then these attributes will not be present in the log.

Access Token Error Mappings

In case of ERROR generated for Access Token service operation, the failure responses generated can be an instance of `problemDetails` or an `AccessTokenErr`. In case of `problemDetails`, the mappings are mentioned as Table 4-9 table. In case of `AccessTokenErr` the mapping is as follows:

Table 4-10 Access Token Err

New Attributes in Logs	AccessTokenErr
errorStatus	400

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-10 (Cont.) Access Token Err

New Attributes in Logs	AccessTokenErr
errorTitle	Bad Request
errorDetails	error_description
errorCause	error

Sample AccessTokenErr message

```
{
    "error": "unauthorized_client",
    "error_description": "NRF-d5g.oracle.com: Nnrf_AccessToken: NfType in
oAuth request is different from the registered profile: ONRF-ACC-ACTOK-E4004",
    "error_uri": None
}
```

Sample Error Log Messages for Access Token Error

```
{
  "instant": {
    "epochSecond": 1717078943,
    "nanoOfSecond": 335260003
  },
  "thread": "XNIO-1 task-2",
  "level": "ERROR",
  "loggerName": "com.oracle.cgbu.cne.nrf.rest.NFAccessTokenController",
  "message": "Response sent: 400 Bad Request for uri : http://ocnrf-
ingressgateway.nrf1-ns/oauth2/token with the problem cause : Bad Request and
problem details : Bad Request",
  "endOfBatch": false,
  "loggerFqcn": "org.apache.logging.log4j.spi.AbstractLogger",
  "threadId": 71,
  "threadPriority": 5,
  "messageTimestamp": "2024-05-30T14:22:23.335+0000",
  "configuredLevel": "WARN",
  "subsystem": "accessToken",
  "processId": "1",
  "nrfTxId": "nrf-tx-1682094534",
  "ocLogId": "1717078942906_77_ocnrf-
ingressgateway-854464d548-426bz:1717078942947_71_ocnrf-
nfaccesstoken-79b55d9b45-qm4gn",
  "xRequestId": "",
  "hostname": "ocnrf-nfaccesstoken-79b55d9b45-qm4gn",
  "errorStatus": "400",
  "errorTitle": "Bad Request",
  "errorDetails": "NRF-d5g.oracle.com: Nnrf_AccessToken: NfType in oAuth
request is different from the registered profile: ONRF-ACC-ACTOK-E4004",
  "errorCause": "unauthorized_client",
  "sender": "NRF-6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c",
  "subscriberId": "imsi-345012123123129"
}
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

Managing the Error Log Messages Enhancements

This section explains the procedure to enable and configure the feature.

Enable

You can enable the Error Log Messages Enhancements feature using the REST API or Console:

- **Enable Using REST API**: Set the value of the feature flags `additionalErrorLogging` and `logSubscriberInfo` as ENABLED for each of the supported microservices as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
  Sample API

```
{
    "appLogLevel": "WARN",
    "packageLogLevel": [
        {
            "packageName": "root",
            "logLevelForPackage": "WARN"
        }
    ],
    "additionalErrorLogging": "ENABLED",
    "logSubscriberInfo": "ENABLED"
}
```

  - *{apiRoot}/nrf-configuration/v1/nfAccessToken/logging*
  - *{apiRoot}/nrf-configuration/v1/nfDiscovery/logging*
  - *{apiRoot}/nrf-configuration/v1/nfRegistration/logging*
  - *{apiRoot}/nrf-configuration/v1/nfSubscription/logging*
  - *{apiRoot}/nrf-configuration/v1/nrfArtisan/logging*
  - *{apiRoot}/nrf-configuration/v1/nrfAuditor/logging*
  - *{apiRoot}/nrf-configuration/v1/nrfConfiguration/logging*
  - *{apiRoot}/nrf-configuration/v1/nrfCacheData/logging*
- **Enable Using CNC Console**: Set the value of the feature flags **Additional Error Logging** and **Log Subscriber Info** as ENABLED for each of the supported microservices in **Logging Level Options**. For more information, see Logging Level Options.

Configure

You can configure the feature using Helm.

Perform the following steps to configure this feature using Helm:

1. Open the `ocnrf_custom_values_25.1.201.yaml` file.
2. Configure the following existing parameters:
   - `global.nrfInstanceId` to determine the NRF NfInstanceId.
   - `global.nfType` to determine the NF type.
3. Save the file.

Copyright © 2019, 2025, Oracle and/or its affiliates.

4. Install NRF. For more information about the installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide.*
5. Run Helm upgrade, if you are enabling this feature after NRF deployment. For more information about the upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide.*

Observability

Metrics

There are no new metrics added for this feature.

Alerts

There are no alerts added or updated for this feature.

KPIs

There are no new KPIs related for this feature.

Maintain

If you encounter alerts at system or application levels, see the NRF Alerts section for the resolution steps.

In case the alerts still persist, perform the following:

1. **Collect the logs**: For more information on collecting logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.

2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.8 Limiting Number of NFProfiles in NFDiscover Response

When NFDiscover service operation request is received, NRF returns NFProfiles based on the query attributes. As per 3GPP TS 29.510, the following two attributes play a key role to limit the number of NFProfiles to be returned in the response:

- `limit`: Maximum number of NFProfiles to be returned in the response.
  Minimum: 1
- `max-payload-size`: Maximum payload size of the response, expressed in kilo octets. When present, the NRF limits the number of NFProfiles returned in the response so as to not to exceed the maximum payload size indicated in the request.
  Default: 124 kilo octets

  Maximum: 2000 (that is, 2 mega octet)

`limit` Attribute

While returning the NFProfiles in the NFDiscover service operation response, NRF limits the number of NFProfiles that can be returned after processing the NFDiscover service operation request based on the value of the `limit` attribute.

In case the `limit` attribute is not present in the NFDiscover search query, then NRF limits the number of NFProfiles based on the value of the `profilesCountInDiscoveryResponse` attribute in the *{apiRoot}/nrf-configuration/v1/nfDiscoveryOptions* API.

Copyright © 2019, 2025, Oracle and/or its affiliates.

`max-payload-size` Attribute

While returning the NFProfiles in the NFDiscover service operation response, NRF limits the number of NFProfiles whose size is within the value of `max-payload-size` attribute, excluding the size of the following attributes defined by 3GPP of the NFDiscover service operation response:

- validityPeriod (Mandatory)
- nfInstances (Mandatory)
- preferredSearch (Conditional)
- nrfSupportedFeatures (Conditional)

In case this attribute is not present in the NFDiscover search query, then NRF considers the default value (124 kilo octets) as defined by 3GPP TS 29.510.

4.9 Support for cnDBTier APIs in CNC Console

NRF earlier fetched the status of cnDBTier using the CLI-based mechanism.

With the implementation of this feature, cnDBTier APIs are integrated into the CNC Console. NRF users can view the specific cnDBTier APIs directly on the CNC Console as mentioned in the following tables:

Table 4-11 cnDBTier APIs

Console Parameter	cnDBTier API name	Description
cnDBTier Backup Status	`http://<base-uri>/ocdbtier/backup/status`	This is a read-only API. This API displays the current backup status of the cnDBTier. It displays the following: • current system time • current cnDBTier backup status • next scheduled backup time
cnDBTier Health	`http://<base-uri>/{unique_prefix}/ocdbtier/health-info`	This API lists the health info of cnDBTier pods.
Backup Manager Health Status	`http://<base-uri>/{unique_prefix}/ocdbtier/health-info/backup-mgr-svc/status`	This is a read-only API. This API displays the health status of the backup manager service. It checks the following: • if the backup manager service is up or not • if the service can connect to database or not

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-11 (Cont.) cnDBTier APIs

Console Parameter	cnDBTier API name	Description
Monitor Health Status	`http://<base-uri>/{unique_prefix}/ocdbtier/health-info/monitor-svc/status`	This is a read-only API. This API displays the health status of the monitor service. It checks the following: • if the monitor service is up or not • if the service can connect to database or not • if the metrics are fetched or not (the metrics are fetched when the service is up and vice versa)
NDB Health Status	`http://<base-uri>/{unique_prefix}/ocdbtier/health-info/ndb-svc/status`	This is a read-only API. This API displays the health status of the NDB service pods such as (data pods, sql pods, app-my-sql pods, mgmt pods). It checks the following: • if the pod is connected to PVC or not • if the pods status is up or not **Note:** PVC Health Status attribute is set to NA when some of the database pods are not connected to the PVC.
Replication Health Status	`http://<base-uri>/{unique_prefix}/ocdbtier/health-info/replication-svc/status`	This is a read-only API. This API displays the health status of the replication service. It checks the following: • if the replication service is up or not • if the replication service can connect to database or not
cnDBTier Version	`http://<base-uri>/ocdbtier/version`	This is a read-only API. This API displays the cnDBTier version.
Backup List	`http://<base-uri>/ocdbtier/list/backups`	This is a read-only API. This API displays the details of completed backups along with backup ID, backup creation timestamp, and backup size.
Database Statistics Report	`http://<base-uri>/ocdbtier/statistics/report/dbinfo`	This is a read-only API. This API displays the number of available database.
Georeplication Status	`dbconfig-geoReplicationStatus`	This API displays the georeplication status.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-11 (Cont.) cnDBTier APIs

Console Parameter	cnDBTier API name	Description
Real Time Overall Replication Status	`http://<base-uri>/ocdbtier/status/replication/realtime`	This is a read-only API. This API displays the overall replication status in multisite deployments. For example, in a four-site deployment, it provides the replication status between the following sites: site1-site2, site1-site3, site1-site4. This is applicable for all other sites, such as, site 2, site 3, and site 4.
Site Specific Real Time Replication Status	`http://<base-uri>/ocdbtier/status/replication/realtime/{remoteSiteName}`	This is a read-only API. This API displays the site-specific replication status.
Replication HeartBeat Status	`http://<base-uri>/ocdbtier/heartbeat/status`	This is a read-only API. This API displays the connectivity status between the local site and the remote site name to which NRF is connected.
Local Cluster Status	`http://<base-uri>/ocdbtier/status/cluster/local/realtime`	This is a read-only API. This API displays the status of the local cluster.
On Demand Backup	`http://<base-uri>/ocdbtier/on-demand/backup/initiate`	This API is used to initiate a new backup. This API displays the status of initiated on-demand backups and helps to create a new backup.
Update Cluster As Failed	`http://<base-uri>/ocdbtier/markcluster/failed`	This API is used to mark a disrupted cnDBTier cluster as failed.
Start Georeplication Recovery	`http://<base-uri>/ocdbtier/faultrecovery/start`	This API is used to start the georeplication recovery process.
Georeplication Recovery Status	`http://<base-uri>/ocdbtier/faultrecovery/status`	This is a read-only API. This API is used to monitor the recovery status of georeplication for both FAILED and ACTIVE cnDBTier sites.

For more information about the above-mentioned cnDBTier APIs, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.

Managing cnDBTier APIs in CNC Console

Enable

The CNC console section for accessing the cnDBTier APIs is available in NRF if the cnDBTier is configured as an instance during the CNC Console deployment. For more information about integrating cnDBTier APIs in CNC Console, see the "NF Single Cluster Configuration With cnDBTier Menu Enabled" section in *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

There is no option to disable this feature.

Configure

cnDBTier APIs can be accessed or configured using the CNC Console. For more information, see cnDBTier APIs in the CNC Console.

Maintain
If you encounter alerts at system or application levels, see the NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

1. Collect the logs: For more information on collecting logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. Raise a service request: See My Oracle Support for more information on how to raise a service request.

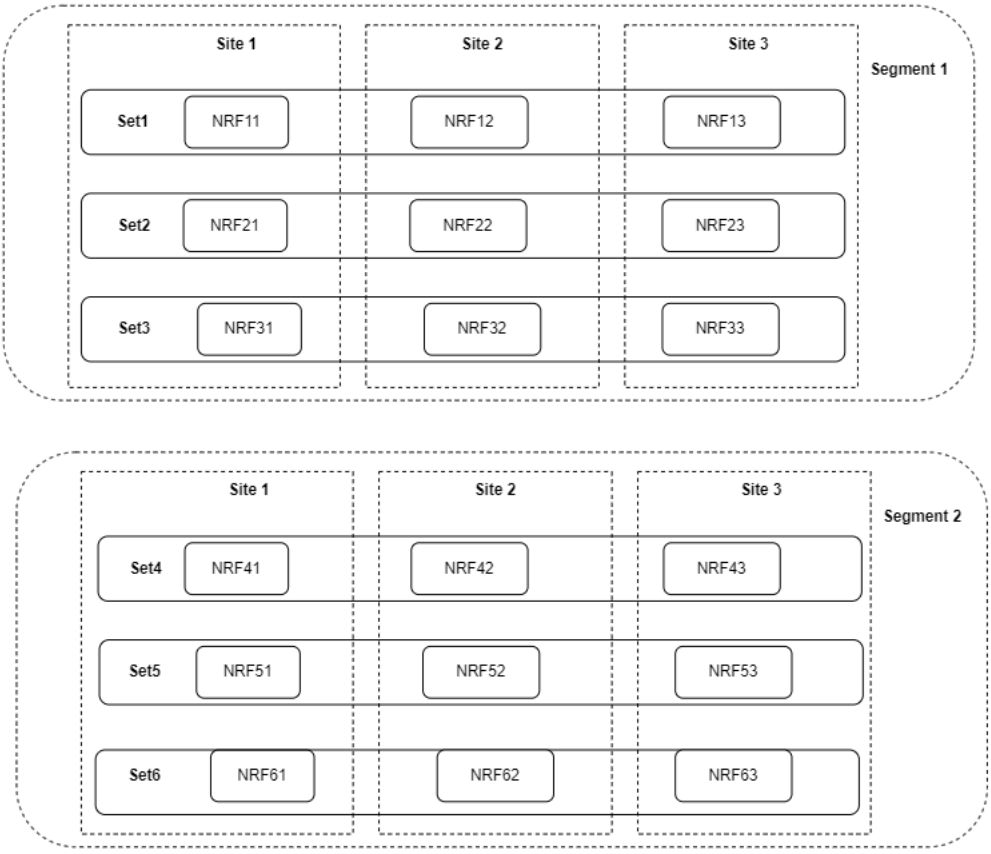
4.10 Enhanced NRF Set Based Deployment (NRF Growth)

NRF was supporting the deployment of two segments of NRF in a network. In the case of capacity expansion of the network to support increased traffic, a single NRF set cannot be scaled beyond a certain capacity.

With the implementation of this feature, NRF supports the deployment of multiple NRF set on the network instead of scaling the capacity of each NRF instance. Each segment in this network can have a single georedundant set and each set can have up to four georedundant NRFs. The georedundant NRFs in the set share the state data with each other.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-8 NRF Segments



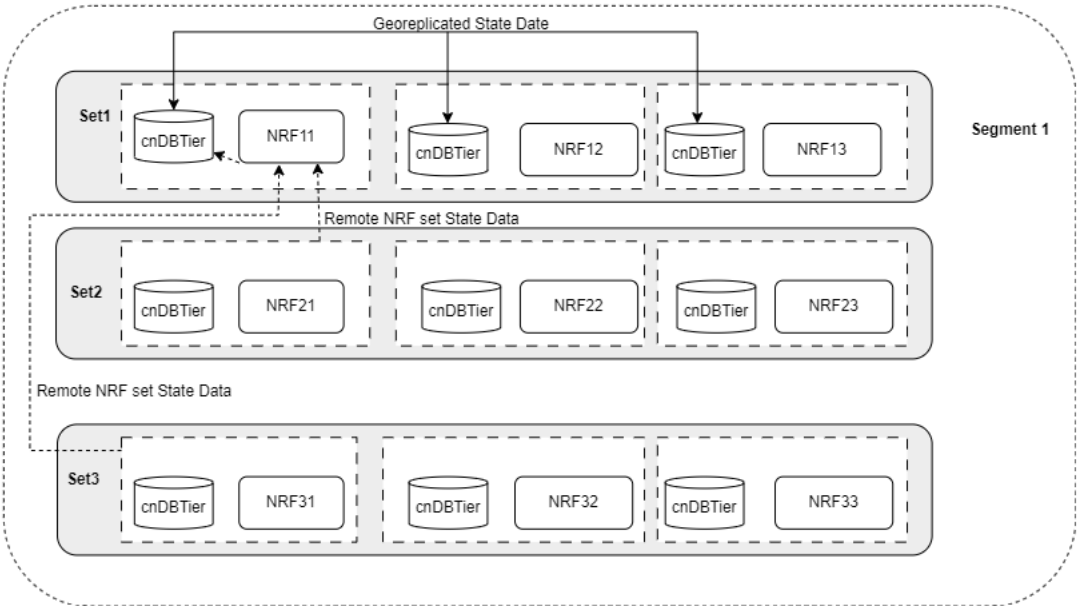
A dedicated cnDBTier instance is deployed for each NRF that is georeplicated within the specific NRF set. Each NRF in the set synchronizes with its most preferred NRFs in the other sets within the segment to retrieve the state data of that set. Thus, every NRF has the complete segment-level view of all the sets in a specific segment.

The state data in each NRF comprises of:

- local data
- georeplicated data of the specific set
- remote data from the other sets.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-9 State Data in NRF



The Cache Data Service (CDS), which is multipod deployment, builds and maintains the state data of the local NRF set and the remote NRF sets in the in-memory cache. Each pod of CDS maintains its cache independently. CDS maintains the segment-level view of the state data. This segment-level view is not pushed to cnDBTier. For more information about CDS, see NRF Architecture.

The cache of local NRF set data:

- is built or updated by querying the cnDBTier periodically
- uses the last known information about the NFs when CDS is not updated due to a database error

The cache of remote set data:

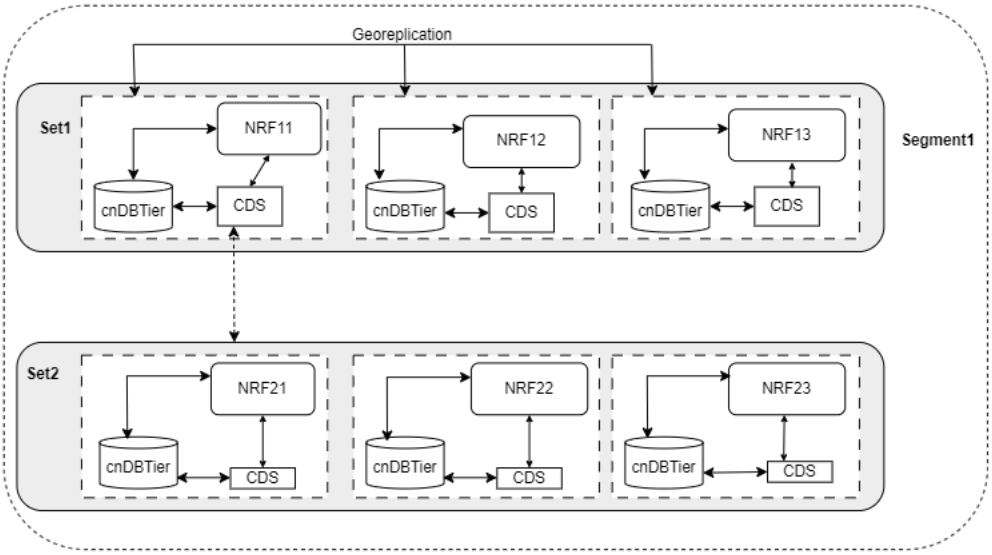
- is built or updated by synchronizing the state data of remote NRF sets periodically.

> **Note**
>
> The periodicity at which the remote NRF set data is synchronized with CDS in-memory cache is 2 seconds. Remote NRF set data synchronization is not performed for each request as it is performed over the WAN network. This may cause higher latency as it involves retries also.

- uses the last known information about the NFs in case remote NRF set sync is failed.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-10 Cache Data Service



During pod initialization, the CDS service marks itself as available only after the local and remote set state data is loaded in the in-memory cache. CDS is only used for reading local and remote state data. The NRF microservices continue to write state data directly to the cnDBTier.

> **Note**
>
> CDS tries to fetch the state data from all the NRFs in remote sets. In case all the NRFs in any of the specific remote sets are unavailable, then CDS will come up with the state data of the remaining NRF sets (if any) and the local data.

The following NRF core microservices read data from CDS for various service operation requests as explained below. The write operations are directed to cnDBTier.

- **NF Registration Microservice (nfregistration)**: The registration microservice queries the CDS for service operations such as NFListRetrieval and NFProfileRetrieval.
- **NF Subscription Microservice (nfsubscription)**: The subscription microservice queries the CDS for service operations such as NFStatusSubscribe, NfStatusSubscribe(Patch), NfStatusUnsubscribe, NfStatusNotify.
- **NRF Auditor Microservice (nrfauditor)**: The auditor microservice queries the CDS to retrieve the total number of NFs per NfType in the segment.
- **NF Access Token Microservice (nfaccesstoken)**: The AccessToken microservice queries the CDS for processing the access token service operations.
- **NRF Configuration Microservice (nrfconfiguration)**: The configuration microservice queries CDS for processing the state data API requests.
- **NF Discovery Microservice (nfdiscovery)**: The discovery microservice queries the CDS periodically to update its local cache.

For more information about the microservices, see the Impacted Service Operations section.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> All the remaining NRF microservices don't query CDS.

This feature interacts with some of the existing NRF features, for more information about this, see the Interaction with Existing Features section.

Managing NRF Growth feature

Enable

Prerequisites

A minimum of two NRF sets are required to enable the feature.

Steps to Enable

1. Upgrade all existing sites to NRF 24.1.0 version or above. Ensure that all the sites are on the same version.
2. Configure a unique `nfSetId` for each NRF set using the `/nrf-configuration/v1/nrfGrowth/featureOptions`. Ensure that the same `nfSetId` must be configured for all the NRFs in the set.
   For example: Consider a case of three NRF instances in a set and the `nfSetId` is ***set1***. Set1 has NRF11, NRF12, and NRF13. Configure ***set1*** as `nfSetId` for NRF11, NRF12, and NRF13. See Figure 4-11 for more details.
3. Install new NRF set(s) or upgrade an existing NRF set(s) to the NRF 24.1.0 version or above.
   For example: Consider the new NRF set as Set2 with three NRF instances, NRF21, NRF 22, and NRF23. See Figure 4-11 for more details.
4. Using the following state data API, retrieve the nf-instances and subscription details in each NRF of each set. Save the output for validation later.
   a. Use `/nrf-state-data/v1/nf-details` to retrieve the nf-instances details.
   b. Use `/nrf-state-data/v1/subscription-details` to retrieve the subscription details.

   For more information about the query parameters, see the REST Based NRF State Data Retrieval section.
5. Configure the `nfSetId` of the new NRF set by setting the attribute `nfSetId` using the API `/nrf-configuration/v1/nrfGrowth/featureOptions`. Ensure that the same `nfSetId` must be configured for all the NRFs in the set.
   For example: `nfSetId` of Set2 is ***set2***. Configure ***set2*** as `nfSetId` for NRF21, NRF 22, and NRF23. See Figure 4-11 for more details.
6. Load the NRF Growth feature specific alerts in both the sets. For more information about the alert configuration, see the NRF Alert Configuration section.
7. Configure the NRFs of the remote NRF sets in each NRF by setting the attribute `nrfHostConfigList` using the API `/nrf-configuration/v1/nrfGrowth/featureOptions`.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> - Once configured, the NRFs will start syncing with the remote NRF sets. The remote set data will not be used for service operations until the feature is enabled.
> - One of the entries in the `staticNrfConfigList` attribute must contain the details of the local NRF set.

For example: Configure the host details of NRFs in Set2 as the remote NRF sets in each NRFs of Set1. For instance, configure the `nrfHostConfigList` attribute in NRF11 of Set1 as remote NRFs host details of NRF21, NRF 22, and NRF23. Similarly, configure each NRFs in Set1 and Set2.

8. Ensure that the following alerts are not present in any of the sets in the network:
   a. OcnrfRemoteSetNrfSyncFailed
   b. OcnrfSyncFailureFromAllNrfsOfAllRemoteSets
   c. OcnrfSyncFailureFromAllNrfsOfAnyRemoteSet
      If present, wait for 30 seconds to 1 minute and retry till the alerts are cleared. If the alerts are not cleared, see alerts for resolution steps.
9. Use the state data API to validate the nf-instances and subscriptions as below to ensure that the remote synching is successful, and all the NFs and subscriptions are synced.
   a. Use `/nrf-state-data/v1/nf-details?testSegmentData=true` to validate the nf-instances.
   b. Use `/nrf-state-data/v1/subscription-details?testSegmentData=true` to validate the subscriptions.

   Where, `testSegmentData` query is used to validate the state data of the segment before enabling the NRF Growth feature.

   > **Note**
   >
   > Ensure that the state data of the NRFs in Set1 is available in the NRFs of Set2. In case you are upgrading the existing set, then ensure that the state data of the NRFs in Set2 is available in the NRFs of Set1.

   For more information about the query parameters, see the REST Based NRF State Data Retrieval section.
10. Enable NRF Growth Forwarding feature, if forwarding is applicable.

    > **Note**
    >
    > When NRF Growth feature is enabled, the existing forwarding options will not be used.

11. Enable the NRF Growth feature in all the NRFs. Enable the feature using the API `/nrf-configuration/v1/nrfGrowth/featureOptions`.
12. Ensure that the following alerts are not present in any of the sets in the network:

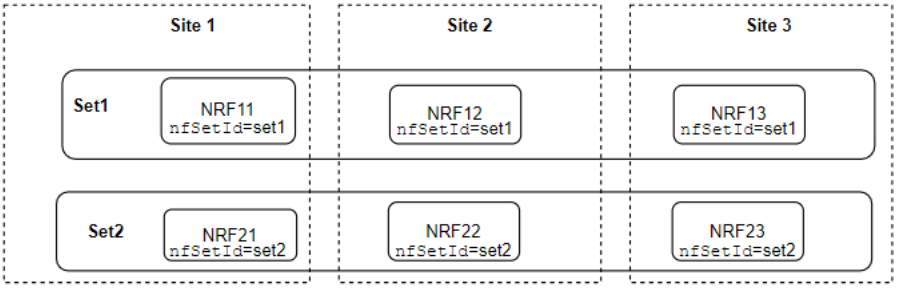
Copyright © 2019, 2025, Oracle and/or its affiliates.

a. OcnrfRemoteSetNrfSyncFailed
b. OcnrfSyncFailureFromAllNrfsOfAllRemoteSets
c. OcnrfSyncFailureFromAllNrfsOfAnyRemoteSet
d. OcnrfDatabaseFallbackUsed

If present, wait for 30 seconds to 1 minute and retry till the alerts are cleared. If the alerts are not cleared, see alerts for resolution steps.

13. After the above configurations are complete, all service operations consider complete segment data for processing the request.
You can migrate the NFs from Set1 to Set2 or vice-versa. For more information about the steps to migrate NFs, see the Migration of NFs section.

Figure 4-11 NRF Growth nfSetId Configuration



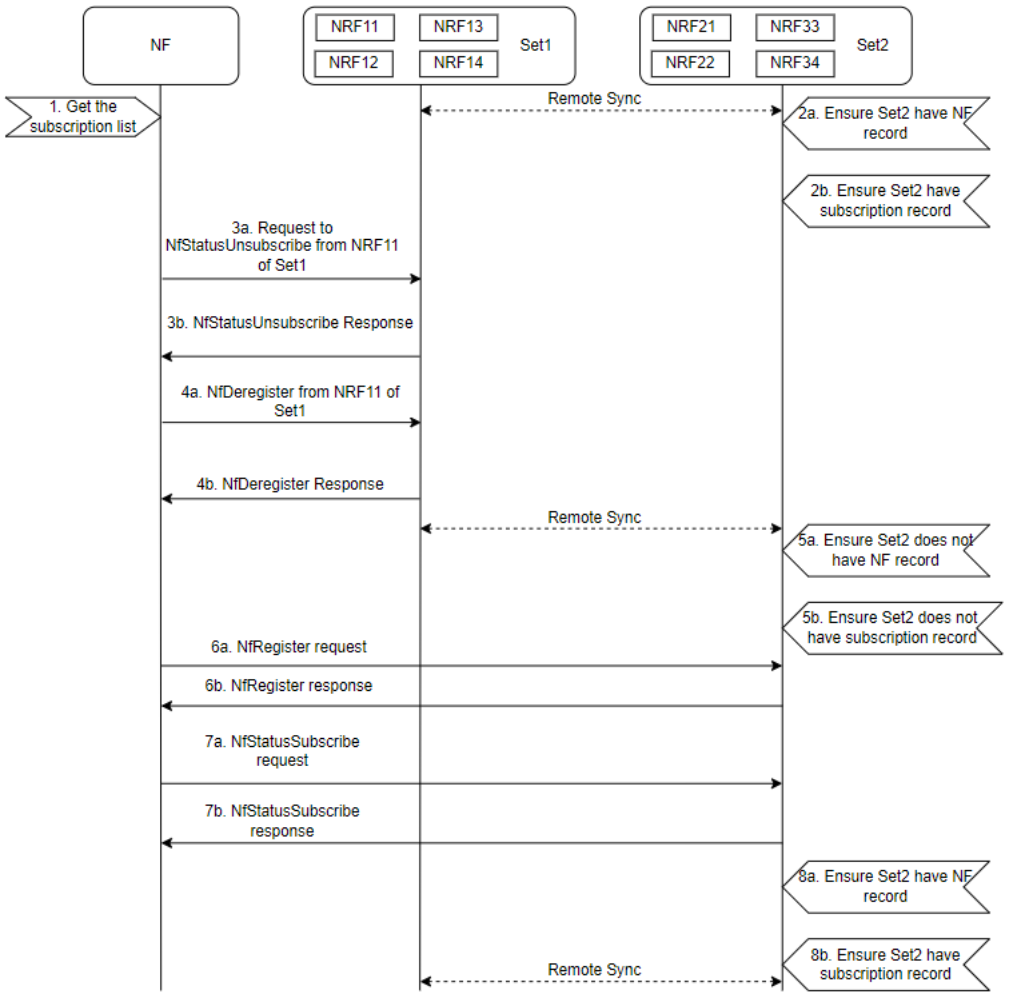
Migration of NFs

1. Make sure that the NF subscription list is available.
2. Use the following state data API at the remote NRF set to validate the NRF has record of this NF record as remote host details. If present, wait for 5-10 seconds and retry the API till the nfInstanceId is not present in the response.
   a. Use `/nrf-state-data/v1/nf-details?nf-instance-id=<the nfInstanceId of the NF>` to validate the nf-instances.
   b. Use `/nrf-state-data/v1/subscription-details?subscription-id= <subscription ID of the subscription>` to validate the subscriptions.
3. Unsubscribe the subscriptions of the NF from the NRF.
   a. NF sends NfStatusUnsubscribe request to NRF.
   b. NRF sends NfStatusUnsubscribe response to NF.
4. Deregister the NF from the NRF.
   a. NF sends NfDeregister request to NRF.
   b. NRF sends NfDeregister response to NF.
5. Use the following state data API at the target NRF to validate the NRF does not have the NF record. If present, wait for 5-10 seconds and retry the API till the nfInstanceId is not present in the response.
   a. Use `/nrf-state-data/v1/nf-details?nf-instance-id=<the nfInstanceId of the NF>` to validate the nf-instances.
   b. Use `/nrf-state-data/v1/subscription-details?subscription-id= <subscription ID of the subscription>` to validate the subscriptions.

Copyright © 2019, 2025, Oracle and/or its affiliates.

6. Register the NF with the NRF at the target NRF set.
   a. NF sends NfRegister request to NRF.
   b. NRF sends NfRegister response to NF.
7. Create the new subscription as required.
   a. NF sends NfStatusSubscribe request to NRF.
   b. NRF sends NfStatusSubscribe response to NF.
8. Use the following state data API at the target NRF to validate that the NF is registered and the subscription is created. The NF record is stored as local registered NF. If not present, wait for 5-10 seconds and retry the API till the nfInstanceId is present in the response.
   a. Use `/nrf-state-data/v1/nf-details?nf-instance-id=<the nfInstanceId of the NF>` to validate the nf-instances.
   b. Use `/nrf-state-data/v1/subscription-details?subscription-id= <subscription ID of the subscription>` to validate the subscription.

Figure 4-12 Migration of NF from one set to another



Copyright © 2019, 2025, Oracle and/or its affiliates.

Configure

Configure the NRF Growth feature using REST API or CNC Console:

- **Configure Forwarding Options for NRF Growth feature using REST API**: Perform the configurations as described in the "Forwarding Options for NRF Growth" section of *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure Forwarding Options for NRF Growth feature using CNC Console**: Perform the configurations as described in the Forwarding Options section.
- **Configure NRF Growth feature using REST API**: Perform the feature configurations as described in the "NRF Growth Options" section of *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure NRF Growth feature using CNC Console**: Perform the feature configurations as described in NRF Growth Options.

> **Note**
>
> The forwarding rules for nfDiscover service requests are based on following 3GPP discovery request parameters:
>
> - target-nf-type ***(Mandatory Parameter)***
> - service-names ***(Optional Parameter)***
>
> The forwarding rules for access token service requests are based on the following 3GPP access token request parameter:
>
> - scope ***(Mandatory Parameter)***

Observe

Metrics

The following metrics are added for the NRF Growth feature.

- `ocnrf_cds_rx_requests_total`
- `ocnrf_cds_tx_responses_total`
- `ocnrf_cds_round_trip_time_seconds`
- `ocnrf_query_remote_cds_requests_total`
- `ocnrf_query_remote_cds_responses_total`
- `ocnrf_query_remote_cds_round_trip_time_seconds`
- `ocnrf_query_remote_cds_message_size_bytes`
- `ocnrf_cache_fallback_total`
- `ocnrf_db_fallback_total`
- `ocnrf_query_cds_requests_total`
- `ocnrf_query_cds_responses_total`
- `ocnrf_query_cds_round_trip_time_seconds`
- `ocnrf_dbmetrics_total`

Copyright © 2019, 2025, Oracle and/or its affiliates.

- `ocnrf_nf_registered_count`
- `ocnrf_cache_sync_count_total`
- `ocnrf_remote_set_unavailable_total`
- `ocnrf_all_remote_sets_unavailable_total`

For more information on the above metrics, see the NRF Cache Data Metrics section.

Alerts

Following are the NRF Growth feature specific alerts:

- OcnrfRemoteSetNrfSyncFailed
- OcnrfSyncFailureFromAllNrfsOfAnyRemoteSet
- OcnrfSyncFailureFromAllNrfsOfAllRemoteSets
- OcnrfCacheDataServiceDown
- OcnrfDatabaseFallbackUsed
- OcnrfTotalNFsRegisteredAtSegmentBelowMinorThreshold
- OcnrfTotalNFsRegisteredAtSegmentBelowMajorThreshold
- OcnrfTotalNFsRegisteredAtSegmentBelowCriticalThreshold

For more information on the above alerts, see the Feature Specific Alerts section.

KPIs

Following are the NRF Growth feature specific KPIs:

- Cache Sync
- Total Number of CDS Requests
- Total Number of CDS Responses
- Total Number of CDS Requests per Service Operation
- Total Number of CDS Responses per Service Operation
- CDS Latency 50%
- CDS Latency 90%
- CDS Latency 95%
- CDS Latency 99%
- Total Number of CDS Requests per Request Type
- Total Number of CDS Responses per Request Type
- Total Number of Remote CDS Requests
- Total Number of Remote CDS Responses
- Remote CDS Query Latency 50%
- Remote CDS Query Latency 90%
- Remote CDS Query Latency 95%
- Remote CDS Query Latency 99%
- Database Fallback
- CDS Cache Sync

Copyright © 2019, 2025, Oracle and/or its affiliates.

For more information on the above metrics, see the NRF Growth Specific KPIs section.

Maintain
If you encounter alerts at system or application levels, see the NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

1. **Collect the logs**: For more information on collecting logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.10.1 Impacted Service Operations

This section explains the change in the functionality of the service operation when multiple NRF sets are deployed.

nfRegistration Service Operation

NFRegister, NFUpdate, or NFDeregister

The NF registers and heartbeats to the Primary NRF. When the *nfRegistration* service receives an NFRegister, NFUpdate, or NFDeregister request, it processes the request and if successful, it creates, updates, or deletes the nf-instances records from the cnDBTier. The nf-instances is made available and used by the service operation in the remote set as well.

The *nfRegistration* service does not query the Cache Data Service (CDS) for these service operations. It directly updates or saves the data in the local cnDBTier.

NFProfileRetrieval

When the *nfRegistration* service receives the NFProfileRetrieval request, it queries the CDS to fetch the NFProfile.

The CDS provides the NFProfile of the local NRF set by querying the cnDBTier. The CDS queries the remote NRF sets periodically and caches the data in the in-memory cache. If the growth feature is enabled, the CDS provides the NFProfile registered at the remote NRF sets by querying the in-memory cache.

The response contains the matching NFProfile either from the local NRF set or from the remote NRF set. The NFProfileRetrieval response is created by the *nfRegistration* service with the nf-instances provided by the CDS.

The responses from the *nfRegistration* service vary based on the following conditions:

- If the NFProfile is not present in the response from the CDS, a response with status code 404 NOT FOUND is sent to the consumer NF.
- If the CDS is unreachable or a non-2xx status code is received, the registration service relies on the cnDBTier to obtain the nf-instances local data and fulfill the service operation. The *nfRegistration* service fetches data from the cnDBTier and only the matching NFProfile from the local NRF set data is sent.
- If the NRF Growth feature is enabled and the remote NRF set is not reachable, CDS provides state data of the local NRF set received from cnDBTier and the last known state data of the remote set from its in-memory cache.

Copyright © 2019, 2025, Oracle and/or its affiliates.

NFListRetrieval

When the nfRegistration service receives an NFListRetrieval request, it queries the CDS to get the list of nfInstanceIDs.

The CDS provides the nf-instances of the local NRF set by querying the cnDBTier. The CDS queries the remote set periodically and caches the data in the in-memory cache. If the growth feature is enabled, the CDS provides the nf-instances of the remote NRF sets by querying the in-memory cache.

The response contains the matching nf-instances from the local NRF set and the remote NRF set. The NFListRetrieval response is created by the *nfRegistration* service with the nf-instances provided by the CDS.

The responses from the *nfRegistration* service vary based on the following conditions:

- If there are no matching nf-instances in either of the sets, an empty response is sent.
- If the CDS is unreachable or a non-2xx status code is received, the registration service falls back to the cnDBTier to get the nf-instances local data and fulfill the service operation. The *nfRegistration* service fetches data from the cnDBTier and only the matching nf-instances from the local NRF set data is sent.
- If the NRF Growth feature is enabled and the remote NRF set is not reachable, CDS provides state data of the local NRF set received from cnDBTier and the last known state data of the remote set from its in-memory cache.

nfSubscription Service Operation

The NFs continue to subscribe for Producer NFs and update their subscriptions using the NRFs of the same georedundant set. The *nfSubscription* service is responsible for creating and maintaining the subscriptions in NRF. It also takes care of triggering NfStatusNotify to the consumer NFs for their subscriptions.

NFStatusSubscribe

When the *nfSubscription* service receives the NfStatusSubscribe request to create a subscription, the *nfSubscription* service checks the following constraints before creating the subscription:

- duplicate subscription based on the `allowDuplicateSubscriptions` parameter configuration. This check is performed across all the sets in the segment. For more information about the configuration of this parameter, see NF Management Options. For more information about the functionality of this parameter for different values configured, see Use Cases for Allow Duplicate Subscriptions.
- subscription limit conditions based on the `subscriptionLimit` parameter configuration. This check is applicable for all the subscriptions at the local NRF set. For more information about the Subscription Limit feature, see the Subscription Limit section.

If the constraints are passed, the *nfSubscription* service creates the subscription and saves it to the cnDBTier. The *nfSubscription* service queries the CDS to get the subscription already created. The CDS provides subscriptions at the local NRF set by querying the cnDBTier. The CDS queries the remote set periodically and caches the data in the in-memory cache. If the growth feature is enabled, the CDS provides the subscriptions of the remote NRF sets by querying the in-memory cache.

The responses from the *nfSubscription* service vary based on the following conditions:

Copyright © 2019, 2025, Oracle and/or its affiliates.

- If the CDS is unreachable or a non-2xx status code is received, the *nfSubscription* service relies on the cnDBTier to get the subscription in local NRF and fulfill the service operation. The *nfSubscription* service creates subscriptions only in the local NRF.
- If the NRF Growth feature is enabled and the remote NRF set is not reachable, CDS provides subscription details of the local NRF set, and the last known information of the remote NRF set is provided.

NfStatusSubscribe(Patch)

When the *nfSubscription* service receives the NfStatusSubscribe (Patch) request to update a subscription, the *nfSubscription* service checks the following constraints before updating the subscription:

- checks if the subscription exists. This check is performed across all the sets in the segment.
- subscription limit conditions based on the `subscriptionLimit` parameter configuration. This check is applicable for all the subscriptions at the local NRF set.

If the constraints are passed, the *nfSubscription* service updates the subscription and saves it to the cnDBTier.

The subscription microservice queries the CDS to retrieve the subscriptions that are created at the local NRF set. The CDS provides the corresponding subscriptions created at the local NRF set by querying the cnDBTier. If the NRF Growth feature is enabled, the CDS provides the matching subscriptions created at the remote NRF sets by querying the in-memory cache.

The responses from the *nfSubscription* service vary based on the following conditions:

- If the CDS is unreachable or a non-2xx status code is received, the *nfSubscription* service relies on the cnDBTier to get the subscription in local NRF and fulfill the service operation. The *nfSubscription* service updates subscriptions only in the local NRF.
- If the NRF Growth feature is enabled and the remote NRF set is not reachable, CDS provides subscription details of the local NRF set, and the last known information of the remote NRF set is provided.

NfStatusUnsubscribe

When the *nfSubscription* service receives the NfStatusUnsubscribe request to delete a subscription, the *nfSubscription* service checks if the subscription exists. This check is performed across all the sets in the segment.

The subscription microservice queries the CDS to retrieve the matching subscriptions that are created at the local NRF set. The CDS provides the corresponding subscriptions created at the local NRF set by querying the cnDBTier. If the NRF Growth feature is enabled, the CDS provides the matching subscriptions created at the remote NRF sets by querying the in-memory cache.

If the subscription is present, it deletes the subscription from the cnDBTier and responds with the 200 OK status code. If the subscriptions created at the one NRF set cannot be deleted at any other NRF set and such requests are rejected. If the subscription is not present, the response with a 404 status code is sent.

The responses from the *nfSubscription* service vary based on the following conditions:

- If the CDS is unreachable or a non-2xx status code is received, the *nfSubscription* service relies on the cnDBTier to get the subscription in local NRF and fulfill the service operation. The *nfSubscription* service unsubscribes the subscriptions only in the local NRF.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- If the NRF Growth feature is enabled and the remote NRF set is not reachable, CDS provides subscription details of the local NRF set, and the last known information of the remote NRF set is provided.

NfStatusNotify

When the *nfSubscription* service triggers notifications towards the consumer NFs that are subscribed for nf-instances status events when the conditions specified in the nfSubscription condition are met. This check is performed across all the sets in the segment.

The subscription service queries the CDS to retrieve the subscriptions matching the nf-instances event that is created at the local NRF set. The CDS provides the corresponding subscriptions created at the local NRF set by querying the cnDBTier. If the NRF Growth feature is enabled, the CDS retrieves the subscriptions matching the nf-instances event at the remote NRF sets by querying the in-memory cache.

The *nfSubscription* service triggers the notification event to the consumer NFs for all the subscriptions received from the CDS.

The responses from the *nfSubscription* service vary based on the following conditions:

- If the CDS is unreachable or a non-2xx status code is received, the nfSubscription service relies on the cnDBTier to get the subscription in local NRF and fulfill the service operation. The *nfSubscription* service triggers the notification for the subscriptions only in the local NRF.
- If the NRF Growth feature is enabled and the remote NRF set is not reachable, CDS provides subscription details of the local NRF set, and the last known information of the remote NRF set is provided.

NFDiscover

When the NFDiscover service receives a discovery request, it looks for Producer NFProfile maintained at the in-memory cache in the discovery service. The discovery service cache is periodically updated by querying the CDS.

CDS updates the local cache of the discovery microservice with the latest local NRF set data CDS sends the response with the latest local NRF set data and cached remote NRF data

Additionally, if the NRF Growth feature is enabled CDS updates the local cache of the discovery microservice with the latest local NRF set data and cached remote NRF data.

In case the CDS is not reachable or a non-2xx status code is received, the discovery service relies on the cnDBTier to get local NRF set data and update its in-memory cache for the local NRF set state data. If the NRF Growth feature is enabled, the last known data of the remote NRF set is used for the discovery service operation.

NfAccessToken

When the *nfAccessToken* service receives a nfAccessToken request with `targetNfInstanceId`, it queries the CDS to validate if the target NF and/or requester NF is registered.

The *nfAccessToken* service queries the CDS to retrieve the `targetNfInstanceId` and `requesterNfInstanceId` that are created at the local NRF set. The CDS provides the corresponding NF instances created at the local NRF set by querying the cnDBTier. If the NRF Growth feature is enabled, the CDS provides the matching NF instances created at the remote NRF sets by querying the in-memory cache.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> The `requesterNfInstanceId` is validated as part of the Access Token Request Authorization feature.

If the CDS is unreachable or a non-2xx status code is received, the *nfAccessToken* service falls back to the cnDBTier to get the NfInstanceId local NRF set and fulfill the service operation.

4.10.2 Interaction with Existing Features

With the deployment of multiple sets of NRFs, the functionality of the following features is impacted.

NRF Forwarding Feature
When the growth feature is enabled, the service requests can be forwarded to NRF in another segment. The service requests are forwarded to NRF in another segment if the requested Producer NfInstances are not registered in any of the NRF sets in the segment.

For more information about configuring the forwarding options, see the Enhanced NRF Set Based Deployment (NRF Growth) section.

Dynamic SLF Feature
When dynamic SLF is configured and growth feature is enabled, the dynamic selection of the SLF profiles is performed across local and remote NRF sets. In a given NRF set, the `preferredSLFLocality` attribute can be utilized to prefer SLFs of the local NRF set over SLFs of remote NRF set for sending SLF query. For more information about the dynamic SLF feature, see the Subscriber Location Function section.

Subscription Limit
When the growth feature is enabled, the subscription count is evaluated within each set.

Note Ensure that the total number of subscriptions in all the sets in the segment must not exceed the `globalMaxLimit` value. Configure the `globalMaxLimit` value accordingly.

For example, consider there are two sets deployed in the segment, the total subscription that is supported in the segment is 1000, and the `globalMaxLimit` of set1 is 300 and set2 is 700. In this case, NRF validates that the subscription count of set1 does not cross 300 and set2 does not cross 700.

For more information about the Subscription Limit feature, see the Subscription Limit section.

REST-Based NRF State Data Retrieval
REST-based NRF state data retrieval feature provides options for Non-Signaling APIs to access NRF state data. It allows the operator to access the NRF state data to understand and debug in the event of failures. NRF exposes the following APIs that are used to fetch the NfProfiles and NfSubscriptions at NRF across the local and remote NRF sets. When the growth feature is enabled, the API behavior is as follows:

1. `{apiRoot}/nrf-state-data/v1/nf-details`
   The API returns the NfProfiles registered at the NRF and its mated sites. The response includes the NfProfiles registered at the local site, and its mated sites if the replication channel between the given site and the mate site is UP.

   When the growth feature is disabled, the remote profiles can still be fetched by setting the query parameter `?testSegmentData=true`.

   Refer Enable section to know the usage of this query parameter.

Copyright © 2019, 2025, Oracle and/or its affiliates.

The NfProfiles are filtered based on the query parameters mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*. The filtered NFProfiles are included in the response.

2. `{apiRoot}/nrf-state-data/v1/subscription-details`
   The API returns the NfSubscriptions created at the NRF and its mated sites. The response includes the NfSubscriptions created at the local site, and its mated sites if the replication channel between the given site and the mate site is UP.

   When the growth feature is disabled, the remote subscriptions can still be fetched by setting the query parameter?`testSegmentData=true`.

   Refer Enable section to know the usage of this query parameter.

   The NfSubscriptions are filtered based on the query parameters mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*. The filtered NfSubscriptions are included in the response.

Kubernetes Probes
Startup Probe

The Cache Data Service startup probe proceeds when the following conditions are met:

1. Connectivity to the cnDBTier is successful.
2. The NRF State and Configuration tables are present, and configuration data is present in the cnDBTier.

Readiness Probe

The Cache Data Service readiness probe runs the container only if the following conditions are met:

1. Connectivity to the cnDBTier is successful.
2. The NRF State and Configuration tables are present, and configuration data is present in the cnDBTier.
3. One successful attempt to load the local NRF set state data into the local cache. The state data is the NfInstances and NFSubscriptions. If no NfInstances or NfSubscriptions are present in the cnDBTier, the cache will remain empty and the start-up probe is considered a success. This check is done only once when the pod comes up for the first time. Subsequent readiness probes will not check this condition.
4. If the growth feature is enabled, there will be one successful attempt to load the state data from all the remote NRF sets that are configured. If no NfInstances or NfSubscriptions are received from the remote NRF Set, that request is considered a success. If the remote set NRFs are not reachable, then after all the reattempts and reroutes, the request is considered a success.

Liveness Probe

The Liveness Probe monitors the health of the critical threads running the Cache Data Service. If any of the threads are detected to be in a deadlock state or non-running, the liveness Probe will fail.

For more information about Kubernetes probes, see the Kubernetes Probes section.

4.11 Support for Automated Certificate Lifecycle Management

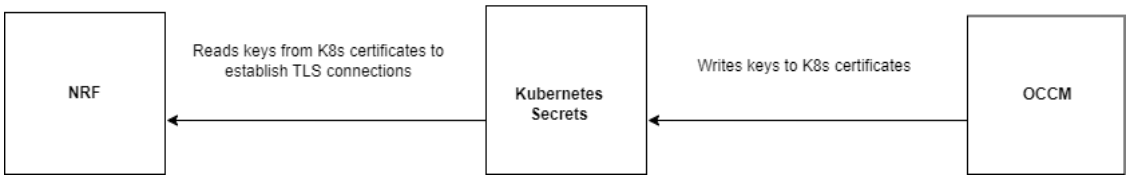
In NRF 23.3.x and earlier, X.509 and Transport Layer Security (TLS) certificates were managed manually. When multiple instances of NRF were deployed in a 5G network,

Copyright © 2019, 2025, Oracle and/or its affiliates.

certificate management, such as certificate creation, renewal, removal, and so on, became tedious and error-prone.

Starting with NRF 23.4.x, you can integrate NRF with Oracle Communications Cloud Native Core, Certificate Management (OCCM) to support automation of certificate lifecycle management. OCCM manages TLS certificates stored in Kubernetes secrets by integrating with Certificate Authority (CA) using the Certificate Management Protocol Version 2 (CMPv2) protocol in the Kubernetes secret. OCCM obtains and signs TLS certificates within the NRF namespace. For more information about OCCM, see *Oracle Communications Cloud Native Core, Certificate Management User Guide*.

Figure 4-13 Support for OCCM



The above diagram indicates that OCCM writes the keys to the certificates and NRF reads these keys to establish a TLS connection with other NFs.

OCCM can automatically manage the following TLS certificates:

- 5G Service Based Architecture (SBA) client TLS certificates
- 5G SBA server TLS certificates
- Message Feed TLS certificates

This feature enables NRF to monitor, create, recreate, and renew TLS certificates using OCCM, based on their validity. For information about enabling HTTPS, see "Configuring Secrets for Enabling HTTPS" in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Install Guide Considerations

- Upgrade: When NRF is deployed with OCCM, follow the specific upgrade procedure. For information about the upgrade strategy, see "Upgrade Strategy" in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
- Rollback: For more information on migrating the secrets from NRF to OCCM and removal of Kubernetes secrets from the yaml file, see "Postupgrade Task" in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Configure
There are no additional configuration changes required at NRF.

Observe
Metrics
This feature uses the existing metrics:

- `oc_egressgateway_connection_failure_total`
- `oc_ingressgateway_connection_failure_total`

For more information, see NRF Gateways Metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Maintain
If you encounter any OCCM-specific alerts, see the "*OCCM Alerts*" section in *Oracle Communications Cloud Native Core, Certificate Management User Guide*.

If you encounter alerts at system or application levels, see the NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

1. Collect the logs: For more information on collecting logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. Raise a service request: See My Oracle Support for more information on how to raise a service request.

4.12 Egress Gateway Route Configuration for Different Deployments

The Egress Gateway route configuration for mTLS and non-TLS based deployments is as follows:

Table 4-12 Egress Gateway Route Configuration for Static Peers

Egress Traffic	Egress Gateway Configuration			Outgoing Message Details		
	Peer under PeerSetConfiguration	RoutesConfiguration.httpsTargetOnly	RoutesConfiguration.httpRuriOnly	3gpp-Sbi-Target-apiRoot	Scheme	Authority Header
HTTPS Outgoing Traffic (TLS)	httpsConfiguration	true	false	Target FQDN Example: <SLF FQDN>/ <NotificationServer>	https	Peer FQDN Example: SCP/ SEPP FQDN
HTTP Outgoing Traffic (non-TLS)	httpsConfiguration	true	true	Target FQDN Example: <SLF FQDN>/ <NotificationServer>	http	Peer FQDN Example: SCP/ SEPP FQDN

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-13 Egress Gateway Route Configuration for virtualHost Peers

Egress Traffic	Egress Gateway Configuration		Outgoing Message Details		
	PeerUnderPeerSetConfiguration.httpsTargetOnly	RoutesConfiguration.httpRuriOnly	3gpp-Sbi-Target-apiRoot	Scheme	Authority Header
HTTPS Outgoing Traffic (TLS)	true http.tlsSCConfiguration	false	Target FQDN Example: <SLF FQDN>/ <NotificationServer>	https	Resolved Peer FQDN Example: Resolved SCP/SEPP FQDN

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-13 (Cont.) Egress Gateway Route Configuration for virtualHost Peers

Egress Traffic	Egress Gateway Configuration		Outgoing Message Details		
	PeerUnderPeerSetConfiguration	RoutesConfiguration.httpsTargetOnly	RoutesConfiguration.httpRuriOnly	3gpp-Sbi-Target-apiRoot	Scheme	Authority Header
HTTP Outgoing Traffic (non-TLS)	http Configuration	false	true	Target FQDN Example: <SLF FQDN>/<NotificationServer>	http	Resolved Peer FQDN Example: Resolved SCP/SEPP FQDN

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> - For service-mesh based deployment, NRF Egress Gateway application container sends HTTP outgoing traffic only. The sidecar container is responsible for sending out the traffic as HTTPS traffic. Hence, for service-mesh based deployment, perform the configuration as per HTTP Outgoing Traffic (non-TLS).

Configuration for HTTPS Outgoing Traffic
For HTTPS Outgoing Traffic, perform the following configuration at NRF Egress Gateway:

1. Update the **PeerConfiguration** as follows:

```
sample PeerConfiguration.json
[
  {
    "id": "peer1",
    "host": "scp-stub-service01",
    "port": "8080",
    "apiPrefix": "/",
    "healthApiPath":"/{scpApiRoot}/{apiVersion}/status"
  }
]
```

2. Update the **PeerSetConfiguration** as follows:

```
sample peerset.json
[
    {
        "id":"set0",
        "httpsConfiguration":[
        {
        "priority": 1,
        "peerIdentifier": "peer1"
        }]
    }
]
```

3. Update the **RoutesConfiguration** as follows:

```
sample RoutesConfiguration.json
{
      "id":"egress_scp_proxy2",
      "uri":"http://localhost:32069/",
      "order":3,
      "metadata":{
         "httpsTargetOnly":true,
         "httpRuriOnly":false,
         "sbiRoutingEnabled":true
      },
      "predicates":[
         {
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
            "args":{
               "pattern":"/nudr-group-id-map/v1/nf-group-ids"
            },
            "name":"Path"
         }
      ],
      "filters":[
         {
            "name":"SbiRouting",
            "args":{
               "peerSetIdentifier":"set0",
               "customPeerSelectorEnabled":false
            }
         }
      ]
   }
```

Configuration for HTTP Outgoing Traffic

For HTTP Outgoing Traffic, perform the following configuration at NRF Egress Gateway:

1. Update the **PeerConfiguration** as follows:

```
sample PeerConfiguration.json
[
  {
    "id": "peer1",
    "host": "scp-stub-service01",
    "port": "8080",
    "apiPrefix": "/",
    "healthApiPath":"/{scpApiRoot}/{apiVersion}/status"
  }
]
```

2. Update the **PeerSetConfiguration** as follows:

```
sample PeerSetConfiguration.json
[
    {
        "id":"set0",
        "httpsConfiguration":[
        {
        "priority": 1,
        "peerIdentifier": "peer1"
        }]
    }
]
```

3. Update the **RoutesConfiguration** as follows:

```
sample RoutesConfiguration.json
{
      "id":"egress_scp_proxy2",
      "uri":"http://localhost:32069/",
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
    "order":3,
    "metadata":{
       "httpsTargetOnly":true,
       "httpRuriOnly":true,
       "sbiRoutingEnabled":true
    },
    "predicates":[
       {
          "args":{
             "pattern":"/nudr-group-id-map/v1/nf-group-ids"
          },
          "name":"Path"
       }
    ],
    "filters":[
       {
          "name":"SbiRouting",
          "args":{
             "peerSetIdentifier":"set0",
             "customPeerSelectorEnabled":false
          }
       }
    ]
}
```

4.13 Routing Egress Messages through SCP

NRF allows routing of following Egress messages through SCP or directly with an option to configure each Egress request type independently:

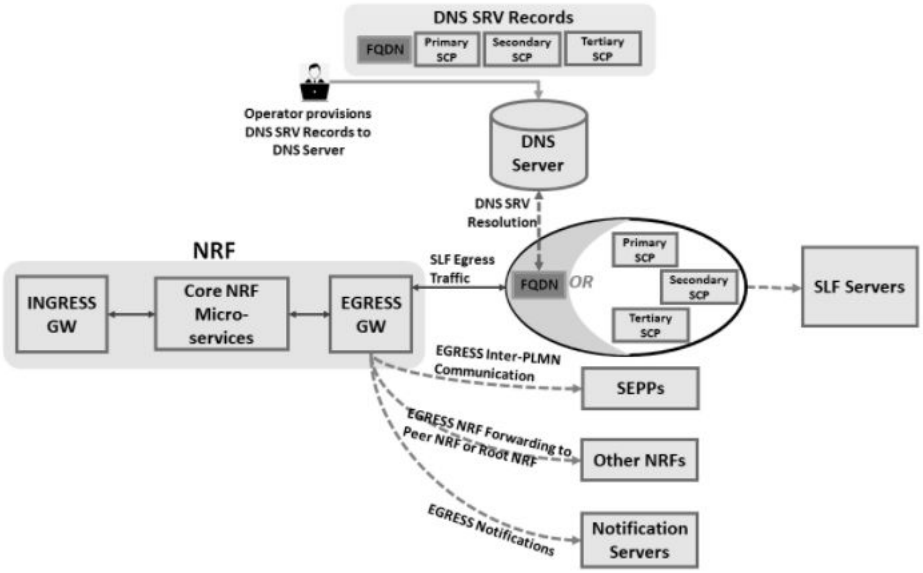
- SLF requests
- Notification requests
- NRF Forwarding requests
- Roaming requests

The above requests are routed through SCP based on the configuration in the Egress Gateway. For more information about the routes configuration, see the "Routes Configuration" section in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
NRF supports routing of the Egress requests through SCP configured with FQDN and virtual FQDN as peer instances.

Alternate route service is used for the resolution of virtual FQDN using DNS-SRV. Egress Gateway uses the virtual FQDN of peer instances to query the Alternate Route Service to get the list of alternate FQDNs, each of which has a priority assigned to them. The Egress Gateway selects the peer instances based on the priority value.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-14 Routing Egress Messages through SCP



Managing the feature

Enable

1. Open the `ocnrf_custom_values_25.1.201.yaml` file.
2. Set `enableNrfArtisanService` to **true** to enable Artisan microservice.

> **Note**
>
> This parameter must be enabled when dynamic SLF feature is configured.

3. Set `alternateRouteServiceEnable` to **true** to enable alternate route service. For more information on enabling alternate route service, see the "Global Parameters" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

> **Note**
>
> This parameter must be enabled when alternate route service through DNS-SRV is required.

4. Save the file.
5. Run helm install. For more information about installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
6. If you are enabling this parameter after NRF deployment, run helm upgrade. For more information about upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Configure

Copyright © 2019, 2025, Oracle and/or its affiliates.

Configure using REST API: Perform the following Egress Gateway configuration as described in the "Egress Gateway Configuration" section of the *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

- create or update the `peerconfiguration` with SCP FQDN details.
- create or update the `peersetconfiguration` to assign these peers.
- create or update the `sbiroutingerrorcriteriasets`.
- create or update the `sbiroutingerroractionsets`.
- create or update the `routesconfiguration` to provide the SCP details. For more information about the routes configuration specific to the above-mentioned requests, see the following sections:
  - SLF Requests
  - Notifications Requests
  - NRF Forwarding Requests
  - Roaming Requests
- If all the requests must be routed through SCP, then create or update the `routesconfiguration` as follows:

```
[
  {
    "id": "egress_scp_proxy1",
    "uri": "http://localhost:32068/",
    "order": 0,
    "metadata": {
      "httpsTargetOnly": true,
      "httpRuriOnly": true,
      "sbiRoutingEnabled": true
    },
    "predicates": [
      {
        "args": {
          "pattern": "/**"
        },
        "name": "Path"
      }
    ],
    "filters": [
      {
        "name": "SbiRouting",
        "args": {
          "peerSetIdentifier": "set0",
          "customPeerSelectorEnabled": false,
          "errorHandling": [
            {
              "errorCriteriaSet": "criteria_1",
              "actionSet": "action_1",
              "priority": 1
            },
            {
              "errorCriteriaSet": "criteria_0",
              "actionSet": "action_0",
              "priority": 2
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
            }
          ]
        }
      }
    ]
  },
  {
    "id": "default_route",
    "uri": "egress://request.uri",
    "order": 100,
    "filters": [
      {
        "name": "DefaultRouteRetry"
      }
    ],
    "predicates": [
      {
        "args": {
          "pattern": "/**"
        },
        "name": "Path"
      }
    ]
  }
]
```

For more information about the routes configuration on different deployments, see Egress Gateway Route Configuration for Different Deployments.

Maintain

If you encounter alerts at system or application levels, see the NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

- Collect the logs: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
- Raise a service request: See My Oracle Support for more information on how to raise a service request.

4.13.1 SLF Requests

NRF supports routing of SLF queries through SCP towards SLF or UDR.

Managing SLF Requests through SCP

Configure

You can configure the SLF requests through SCP using the Egress Gateway configurations either using REST API or Console:

- **Configure using REST API**: Perform the following feature configurations as described in the "Egress Gateway Configuration" section of *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide:*
  - Create or update `{apiRoot}/nrf/nf-common-component/v1/egw/peerconfiguration`.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- Create or update the {apiRoot}/nrf/nf-common-component/v1/egw/peersetconfiguration to assign these peers.
- Create or update {apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerrorcriteriasets.
- Create or update {apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerroractionsets.
- Update the {apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration as mentioned below:

```
curl -v -X PUT "http://10.75.226.126:32247/nrf/nf-common-
component/v1/egw/routesconfiguration" -H  "Content-Type: application/
json"  -d @header.json

sample header.json:-
[
    {
        "id":"egress_scp_proxy1",
        "uri":"http://localhost:32068/",
        "order":0,
        "metadata":{
            "httpsTargetOnly":true,
            "httpRuriOnly":false,
            "sbiRoutingEnabled":true
        },
        "predicates":[
            {
                "args":{
                    "pattern":"/nudr-group-id-map/v1/nf-group-ids"
                },
                "name":"Path"
            }
        ],
        "filters":[
            {
                "name":"SbiRouting",
                "args":{
                    "peerSetIdentifier":"set0",
                    "customPeerSelectorEnabled":true,
                    "errorHandling":[
                        {
                            "errorCriteriaSet":"criteria_1",
                            "actionSet":"action_1",
                            "priority":1
                        },
                        {
                            "errorCriteriaSet":"criteria_0",
                            "actionSet":"action_0",
                            "priority":2
                        }
                    ]
                }
            }
        ]
    },
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
 {
    "id": "default_route",
    "uri": "egress://request.uri",
    "order": 100,
    "predicates": [{
        "args": {
            "pattern": "/**"
        },
        "name": "Path"
    }]
 }
]
```

- **Configure using Console**: Perform the following feature configurations as described:
  - Create or update Peer Configuration.
  - Create or update the Peer Set Configuration to assign these peers.
  - Create or update SBI Error Criteria Sets.
  - Create or update SBI Error Action Sets.
  - Update the Routes Configuration as mentioned above.

> **Note**
>
> To disable the routing of SLF queries through SCP, remove the Egress Gateway configuration from the `{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration`, and retain the `default_route` configuration. This `default_route` configuration is available by default during fresh installation or upgrade.
>
> ```
> {
>     "id": "default_route",
>     "uri": "egress://request.uri",
>     "order": 100,
>     "predicates": [{
>         "args": {
>             "pattern": "/**"
>         },
>         "name": "Path"
>     }]
> }
> ```

4.13.2 Notifications Requests

NRF allows you to route the notification messages through SCP based on the configuration in the Egress Gateway.

Managing Notifications Requests through SCP

This section explains the procedure to configure the feature.
Configure

Copyright © 2019, 2025, Oracle and/or its affiliates.

You can configure the notifications through SCP using the Egress Gateway configurations either using REST API or Console:

- **Configure using REST API**: Perform the following feature configurations as described in the "Egress Gateway Configuration" section of *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide:*
  - Create or update `{apiRoot}/nrf/nf-common-component/v1/egw/peerconfiguration`.
  - Create or update the `{apiRoot}/nrf/nf-common-component/v1/egw/peersetconfiguration` to assign these peers.
  - Create or update `{apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerrorcriteriasets`.
  - Create or update `{apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerroractionsets`.
  - Update the `{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration` as mentioned below:

```
curl -v -X PUT "http://10.75.226.126:32247/nrf/nf-common-
component/v1/egw/routesconfiguration" -H  "Content-Type: application/
json"  -d @header.json

sample header.json:-
{
    "id": "egress_scp_proxy2",
    "uri": "http://localhost:32068/",
    "order": 20,
    "filters": [
      {
        "args": {
          "errorHandling": [
            {
              "priority": 1,
              "actionSet": "action_1",
              "errorCriteriaSet": "criteria_1"
            },
            {
              "priority": 2,
              "actionSet": "action_0",
              "errorCriteriaSet": "criteria_0"
            }
          ],
          "peerSetIdentifier": "set0",
          "customPeerSelectorEnabled": false
        },
        "name": "SbiRouting"
      }
    ],
    "metadata": {
      "httpRuriOnly": false,
      "httpsTargetOnly": true,
      "sbiRoutingEnabled": true
    },
    "predicates": [
      {
        "args": {
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
            "header": "3gpp-Sbi-Callback",
            "regexp": "Nnrf_NFManagement_NFStatusNotify"
          },
          "name": "Header"
        }
      ],
      {
      "id": "default_route",
      "uri": "egress://request.uri",
      "order": 100,
      "predicates": [{
          "args": {
              "pattern": "/**"
          },
          "name": "Path"
      }]
     }
    }
```

- **Configure using Console**: Perform the following feature configurations as described:
  - Create or update Peer Configuration.
  - Create or update the Peer Set Configuration to assign these peers.
  - Create or update SBI Error Criteria Sets.
  - Create or update SBI Error Action Sets.
  - Update the Routes Configuration as mentioned above.

> **Note**
>
> To disable the routing of notification through SCP, remove the `SbiRouting` filter for SCP from the `{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration`, and add the `default_route` configuration. Following the sample `default_route` configuration:
>
> ```
> {
>     "id": "default_route",
>     "uri": "egress://request.uri",
>     "order": 100,
>     "predicates": [{
>         "args": {
>             "pattern": "/**"
>         },
>         "name": "Path"
>     }]
> }
> ```

4.13.3 NRF Forwarding Requests

NRF allows you to route the NRF-NRF forwarding messages through SCP based on the configuration in the Egress Gateway.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Managing NRF Forwarding Requests through SCP

This section explains the procedure to configure the feature.

Configure

You can configure the forwarding through SCP using the Egress Gateway configurations either using REST API or Console:

- **Configure using REST API**: Perform the following feature configurations as described in the "Egress Gateway Configuration" section of *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide:*
  - Create or update `{apiRoot}/nrf/nf-common-component/v1/egw/peerconfiguration`.
  - Create or update the `{apiRoot}/nrf/nf-common-component/v1/egw/peersetconfiguration` to assign these peers.
  - Create or update `{apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerrorcriteriasets`.
  - Create or update `{apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerroractionsets`.
  - Update the `{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration` as mentioned below:

```
curl -v -X PUT "http://10.75.226.126:32247/nrf/nf-common-
component/v1/egw/routesconfiguration" -H  "Content-Type: application/
json"  -d @header.json

sample header.json:-
[
  {
    "id": "egress_scp_proxy1",
    "uri": "http://localhost:32068/",
    "order": 0,
    "filters": [
      {
        "args": {
          "errorHandling": [
            {
              "priority": 1,
              "actionSet": "action_1",
              "errorCriteriaSet": "criteria_1"
            },
            {
              "priority": 2,
              "actionSet": "action_0",
              "errorCriteriaSet": "criteria_0"
            }
          ],
          "peerSetIdentifier": "set0",
          "customPeerSelectorEnabled": false
        },
        "name": "SbiRouting"
      },
      {
        "args": {
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
            "name": "OC-NRF-Forwarding"
          },
          "name": "RemoveRequestHeader"
        }
      ],
      "metadata": {
        "httpRuriOnly": false,
        "httpsTargetOnly": true,
        "sbiRoutingEnabled": true
      },
      "predicates": [
        {
          "args": {
            "header": "OC-NRF-Forwarding",
            "regexp": ".*"
          },
          "name": "Header"
        }
      ]
    },
    {
      "id": "default_route",
      "uri": "egress://request.uri",
      "order": 100,
         "filters": [
           {
             "name": "DefaultRouteRetry"
           }
         ],
      "predicates": [{
          "args": {
              "pattern": "/**"
          },
          "name": "Path"
      }]
  }
  ]
```

- **Configure using Console**: Perform the following feature configurations as described:
  - Create or update Peer Configuration.
  - Create or update the Peer Set Configuration to assign these peers.
  - Create or update SBI Error Criteria Sets.
  - Create or update SBI Error Action Sets.
  - Update the Routes Configuration as mentioned above.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> To disable the NRF-NRF forwarding messages through SCP, remove the Egress Gateway configuration from the `{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration`, and retain the `default_route` configuration. This `default_route` configuration is available by default during fresh installation or upgrade. Following the sample `default_route` configuration:
>
> ```
> {
>     "id": "default_route",
>     "uri": "egress://request.uri",
>     "order": 100,
>        "filters": [
>          {
>            "name": "DefaultRouteRetry"
>           }
>        ],
>     "predicates": [{
>         "args": {
>             "pattern": "/**"
>         },
>         "name": "Path"
>     }]
> }
> ```

4.13.4 Roaming Requests

NRF allows you to route the roaming messages through SCP based on the configuration in the Egress Gateway.

Managing Roaming Requests through SCP

This section explains the procedure to configure the feature.
Configure

You can configure the roaming through SCP using the Egress Gateway configurations either using REST API or Console:

- **Configure using REST API**: Perform the following feature configurations as described in the "Egress Gateway Configuration" section of *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide:*
  - Create or update `{apiRoot}/nrf/nf-common-component/v1/egw/peerconfiguration`.
  - Create or update the `{apiRoot}/nrf/nf-common-component/v1/egw/peersetconfiguration` to assign these peers.
  - Create or update `{apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerrorcriteriasets`.
  - Create or update `{apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerroractionsets`.

Copyright © 2019, 2025, Oracle and/or its affiliates.

– Update the {apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration as mentioned below:

```
curl -v -X PUT "http://10.75.226.126:32247/nrf/nf-common-
component/v1/egw/routesconfiguration" -H  "Content-Type: application/
json"  -d @header.json

sample header.json:-
[
  {
    "id": "egress_scp_proxy1",
    "uri": "http://localhost:32068/",
    "order": 0,
    "filters": [
      {
        "args": {
          "errorHandling": [
            {
              "priority": 1,
              "actionSet": "action_1",
              "errorCriteriaSet": "criteria_1"
            },
            {
              "priority": 2,
              "actionSet": "action_0",
              "errorCriteriaSet": "criteria_0"
            }
          ],
          "peerSetIdentifier": "set0",
          "customPeerSelectorEnabled": false
        },
        "name": "SbiRouting"
      },
      {
        "args": {
          "name": "OC-MCCMNC"
        },
        "name": "RemoveRequestHeader"
      }],
    "metadata": {
      "httpRuriOnly": false,
      "httpsTargetOnly": true,
      "sbiRoutingEnabled": true
    },
    "predicates": [
      {
        "args": {
          "header": "OC-MCCMNC",
          "regexp": "310014"
        },
        "name": "Header"
      }
    ]
  },
  {
    "id": "default_route",
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
    "uri": "egress://request.uri",
    "order": 100,
       "filters": [
         {
           "name": "DefaultRouteRetry"
          }
       ],
    "predicates": [{
        "args": {
            "pattern": "/**"
        },
        "name": "Path"
    }]
   }
]
```

- **Configure using Console**: Perform the following feature configurations as described:
  - Create or update Peer Configuration.
  - Create or update the Peer Set Configuration to assign these peers.
  - Create or update SBI Error Criteria Sets.
  - Create or update SBI Error Action Sets.
  - Update the Routes Configuration as mentioned above.

> **Note**
>
> To disable the routing of roaming messages through SCP, remove the Egress Gateway configuration from the `{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration`, and retain the `default_route` configuration. This `default_route` configuration is available by default during fresh installation or upgrade. Following the sample `default_route` configuration:
>
> ```
> {
>     "id": "default_route",
>     "uri": "egress://request.uri",
>     "order": 100,
>        "filters": [
>          {
>            "name": "DefaultRouteRetry"
>           }
>        ],
>     "predicates": [{
>         "args": {
>             "pattern": "/**"
>         },
>         "name": "Path"
>     }]
> }
> ```


Copyright © 2019, 2025, Oracle and/or its affiliates.

4.14 Ingress Gateway Pod Protection

Ingress Gateway handles all the incoming traffic towards NRF. It may undergo overload conditions due to uneven distribution of traffic, network fluctuations leading to traffic bursts, or unexpected high traffic volume.

This feature protects the Ingress Gateway container from getting overloaded due to uneven traffic distribution, traffic bursts, and congestion. It ensures the protection and mitigation of container from entering an overload condition, while also facilitating necessary actions for recovery.

The pod protection is performed based on the CPU usage and the Pending Message Count of the container as explained in the Congestion State Parameters. These congestion parameters are measured at various states mentioned in the Ingress Gateway Load States to detect the overload condition.

In a service mesh based deployment, all incoming connections to the pod get terminated at the sidecar container, then the sidecar container creates a new connection toward the application container. These incoming connections from the peer are managed by the sidecar and outside the purview of the application container.

Hence when the Ingress Gateway container reaches DOC or Congested level, in a service mesh based deployment, the Ingress Gateway container will only be able to stop accepting new connections from the sidecar container. Also in this state, the Ingress Gateway container will reduce the concurrency of the existing connections between the sidecar container and the Ingress Gateway container. Any new request received over a new connection may get accepted or rejected based on the sidecar connection management.

In a non service mesh based deployment, all incoming connections to the pod get terminated at the Ingress Gateway container. Hence when the Ingress Gateway container reaches DOC or Congested level, the Ingress Gateway container will stop accepting new connections. Also in this state, the Ingress Gateway container will reduce the concurrency of the existing connections between the peer and the Ingress Gateway container. Any new request received over a new connection will result in to a request timeout at the peer.

Congestion State Parameter

As part of the Pod Protection feature, every Ingress Gateway microservice pod monitors its congestion state. Following are the congestion parameters to monitor the pod state:

- CPU
  The congestion state is monitored based on CPU usage to determine the congestion level. The CPU usage is monitored using the Kubernetes cgroup (`cpuacct.usage`) and it is measured in nanoseconds.

  It is monitored periodically and calculated using the following formula and then compared against the configured CPU thresholds to determine the congestion state. For more information about the parameters and the proposed threshold values, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-15 CPU Measurement

$$\frac{\frac{\text{CurrentCpuUsage - LastCpuUsage}}{\text{CurrentTime - LastSampleTime}} \times 100}{\text{CPUs}}$$

Where,

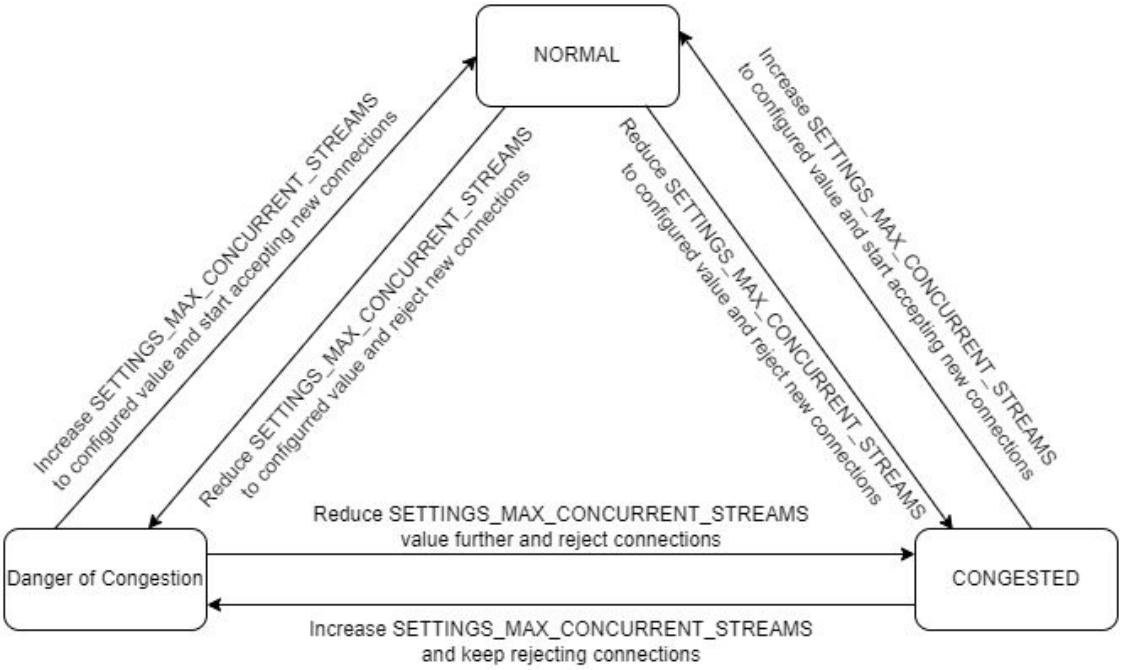
- `CurrentCpuUsage` is the counter reading at current periodic cycle.
- `LastCpuUsage` is the counter reading at previous periodic cycle.
- `CurrentTime` is the current time snapshot.
- `LastSampleTime` is the previous periodic cycle time snapshot.
- `CPUs` is the total number of CPUs for a given pod.

- Pending Message Count: The pending message count is the number of requests that are received by the Ingress Gateway pod from other NFs and yet to send the response. This includes all the requests triggered towards the Ingress Gateway pod. The pending message count is monitored periodically and compared against the configured thresholds to determine the congestion state.

Ingress Gateway Pod States

Following are the various states to detect overload conditions. This will protect and mitigate the pod entering into an unstable condition, and to take necessary actions to recover from this condition.

Figure 4-16 Pod Protection State Transition



Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> The transition can occur between any states. The threshold for these congestion parameters are preconfigured and must not be changed.

- Congested State: This is the upper bound state where the pod is congested. This means one or more congestion parameters are above the configured thresholds for the congested state. For more information about the configuration, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*. The pod can be transitioned to the Congested state either from the Normal State or the DoC state. When the pod reaches this state, the following actions are performed:
  - new incoming HTTP2 connection requests are not accepted.
  - the pod gradually decrements the number of concurrent streams by updating SETTINGS_MAX_CONCURRENT_STREAMS parameter in a SETTINGS frame to the configured `maxConcurrentStreamsPerCon` value at a regular interval. The number of concurrent streams are decremented based on the value configured in `decrementBy` parameter. And, the regular interval is configured in the `decrementSamplingPeriod` parameter.
- Danger of Congestion (DoC) State: This is the intermediate state where the pod is approaching a congested state. This means if one or more congestion parameters are above the configured thresholds for the DoC state. For more information about the configuration, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

  When the pod reaches this state, the following actions are performed:
  - any new incoming HTTP2 connection requests are not accepted.
  - if the pod is transitioning from the Normal state to the DoC state, the pod gradually decrements the number of concurrent streams by updating SETTINGS_MAX_CONCURRENT_STREAMS parameter in a SETTINGS frame to the configured `maxConcurrentStreamsPerCon` value at a regular interval. The number of concurrent streams are decremented based on the value configured in `decrementBy` parameter. And, the regular interval is configured in the `decrementSamplingPeriod` parameter.
  - if the pod is transitioning from the Congested state to the DoC state, the pod gradually increments the number of concurrent streams by updating SETTINGS_MAX_CONCURRENT_STREAMS parameter in a SETTINGS frame to the configured `maxConcurrentStreamsPerCon` value at a regular interval. The number of concurrent streams are incremented based on the value configured in `incrementBy` parameter. And, the regular interval is configured in the `incrementSamplingPeriod` parameter.
- Normal State: This is the lower bound state where the CPU usage is below the configured thresholds for DoC and Congested states. For more information about the configuration, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

  When the pod reaches this state, the following actions are performed:
  - the pod will continue accepting new incoming HTTP2 connection requests.
  - in case the pod is transitioning from the Congested or DoC state to Normal state, the pod gradually increments the number of concurrent streams by updating SETTINGS_MAX_CONCURRENT_STREAMS parameter in a SETTINGS frame to the configured `maxConcurrentStreamsPerCon` value at a regular interval. The number of

Copyright © 2019, 2025, Oracle and/or its affiliates.

concurrent streams are incremented based on the value configured in `incrementBy` parameter. And, the regular interval is configured in the `incrementSamplingPeriod` parameter.

To avoid toggling between these states due to traffic pattern, it is required for the pod to satisfy the condition of the target state for a given period before transitioning it to target state. For example, if the pod is transitioning from DoC to Congested state, then the pod must satisfy the threshold parameters of Congested state for a given period, before moving to Congested state.

The below configurations are used to define the period till which the pod has to be in a particular state:

- `stateChangeSampleCount`
- `monitoringInterval`

Formula for calculating the period is: `(stateChangeSampleCount * monitoringInterval)`

For more information about the configuration, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Managing Ingress Gateway Pod Protection

This section explains the procedure to enable and configure the feature.
Enable

You can enable the Pod Protection feature using the REST API or Console:

- **Enable using REST API**: Perform the following configurations
  1. Use the API path as `{apiRoot}/nf-common-component/v1/igw/podprotection`.
  2. Set `podprotection.enabled` to **true**.
  3. Set `podProtection.congestionControl.enabled` to **true**.
  4. Run the API using PUT method with the proposed values given in the Rest API. For more information about the configuration using REST API, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

  > **Note**
  >
  > The proposed values are engineering configured values and must not be changed.

- **Enable using Console**: Perform the following configurations in Pod Protection:
  1. Switch **Enabled** to enable the feature.
  2. Switch **Enabled** under **Congestion Control** section.
  3. Click **Save** to save the changes.

Configure

You can configure the feature using the Ingress Gateway configurations either using REST API or Console:

- **Configure using REST API**: Perform the following feature configurations as described in the "Ingress Gateway Configuration" section of *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- **Configure using Console**: Perform the feature configurations as described in Pod Protection.

Observe

Metrics

Following metrics are added in the NRF Gateways Metrics section:

- oc_ingressgateway_pod_congestion_state
- oc_ingressgateway_pod_resource_stress
- oc_ingressgateway_pod_resource_state
- oc_ingressgateway_incoming_pod_connections_rejected_total

Alerts

Following metrics are added in the NRF Alerts section:

- OcnrfIngressGatewayPodInDangerOfCongestionState
- OcnrfIngressGatewayPodInCongestedState
- OcnrfIngressGatewayPodCpuUsageInCongestedState
- OcnrfIngressGatewayPodCpuUsageInDangerOfCongestionState
- OcnrfIngressGatewayPodPendingMessageInCongestedState
- OcnrfIngressGatewayPodPendingMessageInDangerOfCongestionState

KPIs

Added the feature specific KPIs in the Ingress Gateway Pod Protection section.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information about how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.15 Network Slice Specific Metrics

The 5G Network slices are identified by Network Slice Instances (NSIs) and Single Network Slice Selection Assistance Information (SNSSAI).

A Network Function (NF) can have multiple NSIs and SNSSAIs listed under them to support multiple slices. NRF supports measuring the number of requests and responses for various service operation per network slice. This measurement is performed using the metrics mentioned in Observe.

Observe

Metrics

Copyright © 2019, 2025, Oracle and/or its affiliates.

Following are the metrics to measure the requests and responses:

- ocnrf_nfDiscover_rx_requests_perSnssai_total
- ocnrf_nfDiscover_tx_success_response_perSnssai_total
- ocnrf_nfDiscover_tx_empty_response_perSnssai_total
- ocnrf_nfDiscover_tx_failure_response_perSnssai_total
- ocnrf_nfDiscover_rx_requests_perNsi_total
- ocnrf_nfDiscover_tx_success_response_perNsi_total
- ocnrf_nfDiscover_tx_empty_response_perNsi_total
- ocnrf_nfDiscover_tx_failure_response_perNsi_total
- ocnrf_nfDiscover_tx_forwarded_requests_perSnssai_total
- ocnrf_nfDiscover_rx_success_forwarded_responses_perSnssai_total
- ocnrf_nfDiscover_rx_empty_forwarded_responses_perSnssai_total
- ocnrf_nfDiscover_rx_failure_forwarded_responses_perSnssai_total
- ocnrf_nfDiscover_tx_forwarded_requests_perNsi_total
- ocnrf_nfDiscover_rx_success_forwarded_responses_perNsi_total
- ocnrf_nfDiscover_rx_empty_forwarded_responses_perNsi_total
- ocnrf_nfDiscover_rx_failure_forwarded_responses_perNsi_total
- ocnrf_nfRegister_requests_perSnssai_total
- ocnrf_nfRegister_success_responses_perSnssai_total
- ocnrf_nfRegister_failure_responses_perSnssai_total
- ocnrf_nfRegister_requests_perNsi_total
- ocnrf_nfRegister_success_responses_perNsi_total
- ocnrf_nfRegister_failure_responses_perNsi_total
- ocnrf_nfUpdate_requests_perSnssai_total
- ocnrf_nfUpdate_success_responses_perSnssai_total
- ocnrf_nfUpdate_failure_responses_perSnssai_total
- ocnrf_nfUpdate_requests_perNsi_total
- ocnrf_nfUpdate_success_responses_perNsi_total
- ocnrf_nfUpdate_failure_responses_perNsi_total
- ocnrf_nfDeregister_requests_perSnssai_total
- ocnrf_nfDeregister_success_responses_perSnssai_total
- ocnrf_nfDeregister_failure_responses_perSnssai_total
- ocnrf_nfDeregister_requests_perNsi_total
- ocnrf_nfDeregister_success_responses_perNsi_total
- ocnrf_nfDeregister_failure_responses_perNsi_total
- ocnrf_nfHeartBeat_requests_perSnssai_total
- ocnrf_nfHeartBeat_success_responses_perSnssai_total
- ocnrf_nfHeartBeat_failure_responses_perSnssai_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

- ocnrf_nfHeartBeat_requests_perNsi_total
- ocnrf_nfHeartBeat_success_responses_perNsi_total
- ocnrf_nfHeartBeat_failure_responses_perNsi_total

For more information about the metrics, see Network Slice Specific Metrics section.

KPIs

The feature specific KPIs are added in the Network Slice Specific KPIs section.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

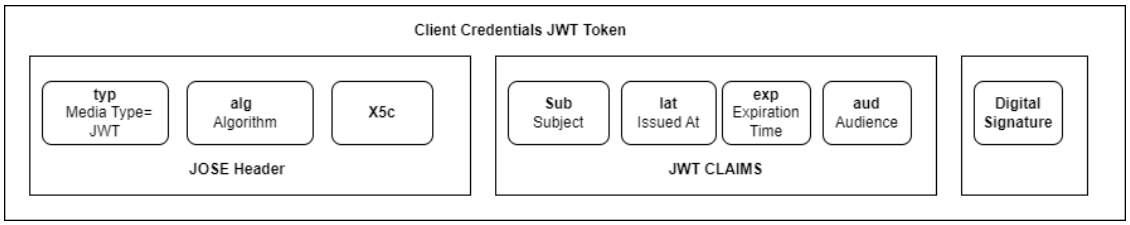
1. **Collect the logs**: For more information on how to collect logs, see "Collecting Logs" section in *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.16 CCA Header Validation in NRF for Access Token Service Operation

Client Credentials assertion (CCA) is a token signed by the Consumer NF. It enables NRF to authenticate the Consumer NF which includes the signed token in Access Token service request. CCA header contains the Consumer NF's NfInstanceId that gets checked against the certificate by the NRF. The CCA also includes a timestamp as the basis for the restriction of its lifetime.

Consumer NF sends the *3gpp-Sbi-Client-Credentials* header containing the CCA in the HTTP request and NRF performs the CCA validation. The CCA header validation is a JWT-based validation at NRF where the Consumer NF sends a JWT token as part of the header to NRF in Access Token request. The JWT token has an X5c certificate and other NF-specific information that is validated against the configuration values defined in NRF. The signature of the JWT token is validated against the CA root certificate configured at NRF.

Figure 4-17 Client Credentials JWT Token



Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-14 JOSE header

Attribute name	Data type	P	Cardinality	Description
typ	String	M	1	The "typ" (type) Header Parameter is used to declare the media type of the JWS. Default value: JWT
alg	String	M	1	The "alg" (algorithm) Header Parameter is used to secure the JWS. Supported Algorithm types: RSA/ECDSA.
X5c	Array	M	1	The "X5c" (X.509 certificate) Header Parameter contains the X.509 public key certificate corresponding to the key used to digitally sign the JWT.

Table 4-15 JWT claims

Attribute name	Data type	P	Cardinality	Description
sub	NfInstanceId	M	1	This IE contains the NF instance ID of the NF service consumer, corresponding to the standard "Subject" claim.
iat	integer	M	1	This IE indicates the time at which the JWT was issued, corresponding to the standard "Issued At" claim. This claim may be used to determine the age of the JWT.
exp	integer	M	1	This IE contains the expiration time after which the client credentials assertion is considered to be expired, corresponding to the standard "Expiration Time" claim.
aud	array(NFType)	M	1..N	This IE contains the NF type of NRF, for which the claim is applicable, corresponding to the standard "Audience" claim.

The digitally signed client credentials assertion is further converted to the JWS compact serialization encoding as a string.

If the validation is successful, the Access Token request is processed further.

If CCA header validation fails, the Access Token request is rejected by NRF with "`403 Forbidden`" with the cause attribute set to "`CCA_VERIFICATION_FAILURE`".

Managing CCA Header validation in NRF for Access Token Service Operation

1. Create Ingress Gateway Secret. For more information about configuring secrets, see section “Configuring Secret to Enable CCA Header” in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide.*
2. Configure the secret containing the CA root bundle and enable CCA header feature for AccessToken either using REST or Console:

Copyright © 2019, 2025, Oracle and/or its affiliates.

- **Using REST API**: Perform the feature configurations for CCA Header Validation in Ingress Gateway as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*
- **Using Console**: Perform the feature configurations for CCA Header Validation in Ingress Gateway as described in CCA Header.

3. Enable the feature using Helm, as follows:
   a. Customize the `ocnrf_custom_values_25.1.201.yaml` file.
   b. To enable this feature in Helm, set the `metadata.ccaHeaderValidation.enabled` to true for `accesstoken_mapping` id under routesConfig of Ingress Gateway microservice.

```
metadata:
  ccaHeaderValidation:
    enabled: true
```

> **Note**
>
> This feature can be enabled only by using Helm.

   c. Save the file.
   d. Run Helm upgrade, if you are enabling this feature after NRF deployment. For more information about upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide.*

Observe

Metrics

Following are the CCA Header Validation in NRF for Access Token Service Operation feature specific metrics in NRF Gateways Metrics section:

- `oc_ingressgateway_cca_header_request_total`
- `oc_ingressgateway_cca_header_response_total`
- `oc_ingressgateway_cca_certificate_info`

KPIs

There are no KPIs for this feature.

Alerts

Following are the CCA Header validation in NRF for Access Token Service Operation feature specific alerts in Feature Specific Alerts section:

- OcnrfCcaRootCertificateExpiringIn5Days
- OcnrfCcaRootCertificateExpiringIn1Day
- OcnrfCcaRootCertificateExpiringIn4Hours

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

Copyright © 2019, 2025, Oracle and/or its affiliates.

In case the alerts persist, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see "Collecting Logs" section in *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.17 Monitoring the Availability of SCP Using SCP Health APIs

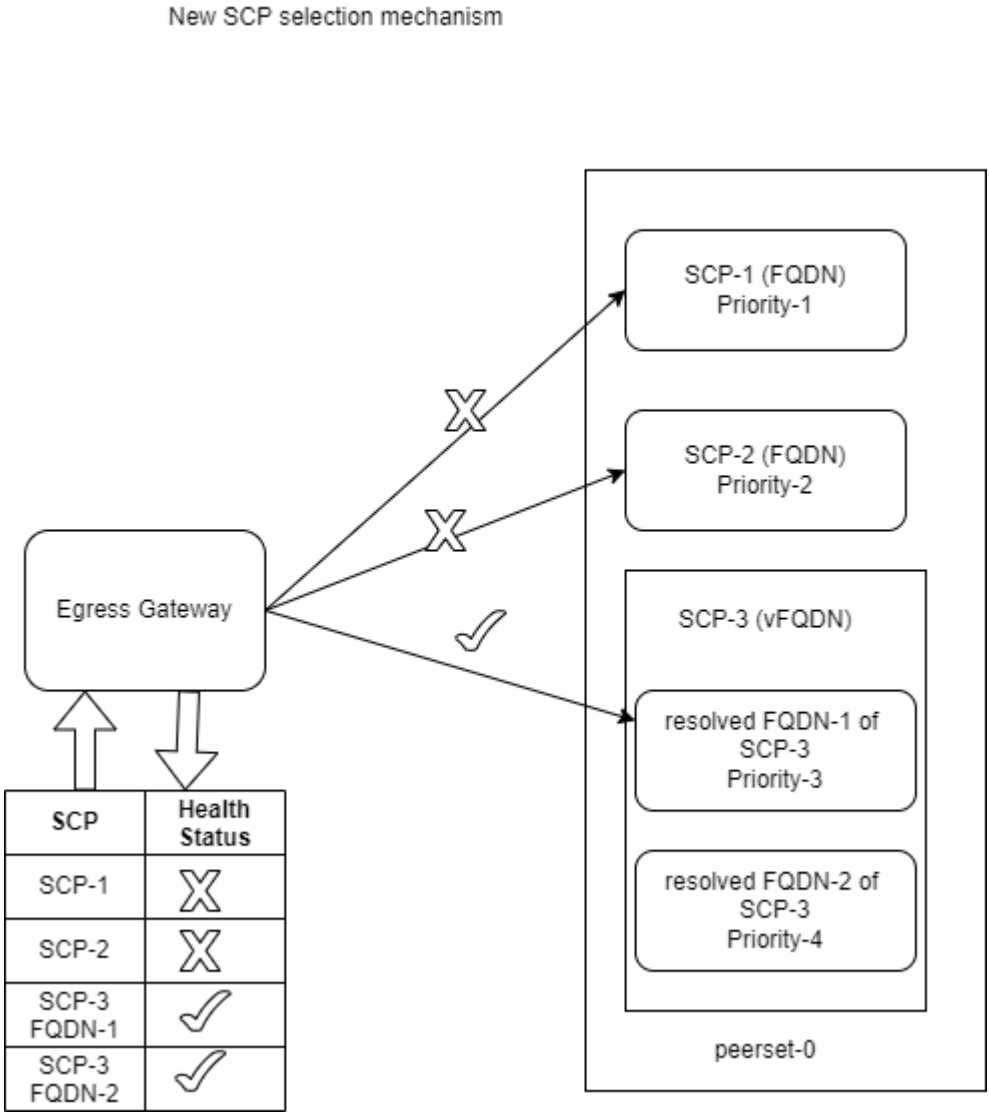
NRF determines the availability and reachability status of all SCPs irrespective of the configuration types.

This feature is an enhancement to the existing SBI routing functionality. Egress Gateway microservice interacts with SCP on their health API endpoints using HTTP2 OPTIONS method. It monitors the health of configured SCP peers to ensure that the traffic is routed directly to the healthy peers. This enhancement avoids routing or rerouting towards unhealthy peers, thus minimizing the latency time.

Egress Gateway microservice maintains the health status of all available and unavailable SCPs. It maintains the latest health of SCPs by periodically monitoring and uses this data to route egress traffic to the most preferred healthy SCP.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-18 SCP Selection Mechanism



Once *peerconfiguration*, *peersetconfiguration*, *routesconfiguration*, and *peermonitoringconfiguration* parameters are configured at Egress Gateway microservice, and all SCPs (after Alternate Route Service (ARS) resolution, if any vFQDN is configured) are marked initially as healthy. The peers attached to the associated peerset are scheduled to run health API checks and update the health status continuously.

During the installation, the value of the parameter `peermonitoringconfiguration` is set to false by default. Since, this feature is an add-on to the existing SBI Routing feature and will be activated if the `sbirouteconfig` feature is enabled. To enable this feature, perform the following:

- configure `peerconfiguration` with `healthApiPath` as `/{scpApiRoot}/{apiVersion}/status`
- configure `peersetconfiguration`
- configure `routesconfiguration`
- configure `sbiroutingerrorcriteriasets`

Copyright © 2019, 2025, Oracle and/or its affiliates.

- configure `sbiroutingerroractionsets`
- enable `peermonitoringconfiguration`

If SBI Routing feature is enabled before upgrading, the `healthApi` in `peerconfiguration` should be attached manually to existing configured peers. If the operator tries to enable `peermonitoringconfiguration` and the targeted peers do not have the `healthApiPath` then an appropriate error response is sent.

Managing Monitoring the Availability of SCP Using SCP Health APIs

This section explains the procedure to enable and configure the feature.
Configure

You can configure the Monitoring the Availability of SCP using the REST API or Console:

- **Configure using REST API**: Perform the following feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide:*
  - Create or update `{apiRoot}/nrf/nf-common-component/v1/egw/peerconfiguration`.
  - Create or update the `{apiRoot}/nrf/nf-common-component/v1/egw/peersetconfiguration` to assign these peers.
  - Create or update `{apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerrorcriteriasets`.
  - Create or update `{apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerroractionsets`.
  - create or update routesets `{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration` to use the above peerset.
  - enable the feature using the below peermonitoring configuration `{apiRoot}/nrf/nf-common-component/v1/egw/peermonitoringconfiguration`.

> **Note**
>
> Health Monitoring of the peer will start only after the feature is enabled and the corresponding peerset is used in `sbirouteconfig`.

- **Configure using Console**: Perform the following feature configurations as described:
  - create or update Peer Configuration with health status endpoint details.
  - create or update the Peer Set Configuration to assign these peers.
  - create or update SBI Error Criteria Sets.
  - create or update SBI Error Action Sets.
  - create or update Routes Configuration to use the above peerset.
  - enable the feature using the below Peer Monitoring Configuration.

> **Note**
>
> Health Monitoring of the peer will start only after the feature is enabled and the corresponding peerset is used in `sbirouteconfig`.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Observe

Metrics
Following metrics are added in the NRF Gateways Metrics section:

- oc_egressgateway_peer_health_status
- oc_egressgateway_peer_health_ping_request_total
- oc_egressgateway_peer_health_ping_response_total
- oc_egressgateway_peer_health_status_transitions_total
- oc_egressgateway_peer_count
- oc_egressgateway_peer_available_count

Alert
Following alerts are added in the NRF Alerts section:

- OcnrfSCPMarkedAsUnavailable
- OcnrfAllSCPsMarkedAsUnavailable

KPIs
Added the feature specific KPIs in the SCP Health Status section.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.18 Controlled Shutdown of NRF

NRF supports controlled shutdown feature to isolate NRF from the current network at a particular site. This isolation helps to perform any maintenance activities or recovery procedures as required without uninstalling the NRF at the particular site. During this time the operations of the Ingress Gateway, Egress Gateway, and NrfAuditor microservices are paused. These services read the operational state from the database periodically. The operational state of NRF can be changed by using REST API or CNC Console at any time.

The two operational states defined for NRF are as follows:

- NORMAL
- COMPLETE_SHUTDOWN

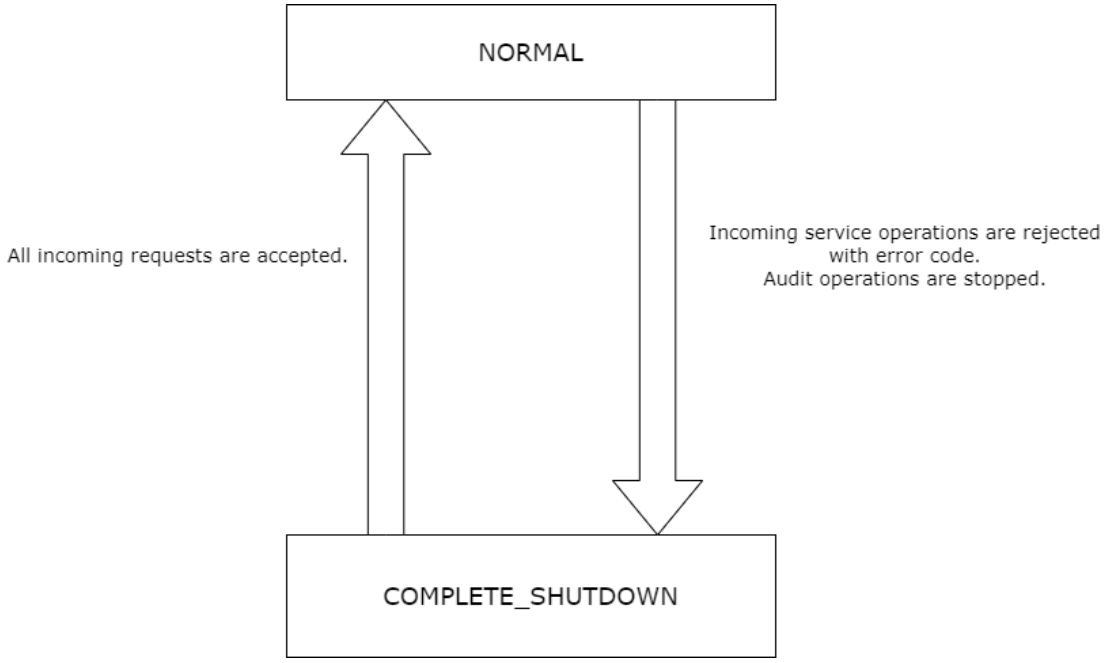
Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> - Ensure that the database is up and the NRF services must be able to connect and communicate with the database to change the operational state.
> - In either state, if the database goes down, the back-end pods go to the NOT_READY state, but the Ingress Gateway microservice will continue to be in the last known operational state. When the Controlled Shutdown feature is enabled and the database is not available, all the incoming messages will be rejected with the *Service Unavailable* messages if the operational state is "NORMAL", or with a configurable error code if the operational state is COMPLETE_SHUTDOWN". However, the operators will not be able to see the operational state in CNC Console due to database unavailability.

If the controlled shutdown operational state is NORMAL, then NRF processes the message as normal.

If the controlled shutdown operational state is COMPLETE_SHUTDOWN, then NRF rejects all incoming requests with a configurable error code.

Figure 4-19 Operational State changes



The following behavior changes occur when the operational state changes:

From NORMAL to COMPLETE_SHUTDOWN

- The Ingress Gateway microservice rejects all new requests towards the services that have been configured for controlled shutdown with a configurable error code. In this case, all the inflight transactions will be gracefully handled. The controlled shutdown applies to the following services:
  - nfregistration

Copyright © 2019, 2025, Oracle and/or its affiliates.

  - nfdiscovery
  - nfsubscription
  - nfaccesstoken

  The error codes are configured using `controlledshutdownerrormapping` and `errorcodeprofiles` APIs.

- The NrfAuditor microservice pauses all its audit procedures, hence no notifications are generated. `OcnrfAuditOperationsPaused` alert is raised to indicate that the audit is paused.
- The Egress Gateway microservice handles all the inflight requests gracefully. Since the Ingress Gateway and NrfAuditor microservices are in a COMPLETE_SHUTDOWN operational state, there would be no requests from the backend services.

> **Note**
>
> No specific configuration for controlled shutdown needs to be applied on the routes for the Egress Gateway microservice.

- The NrfConfiguration microservice continues to process any configuration requests in the COMPLETE_SHUTDOWN state.
- `OcnrfOperationalStateCompleteShutdown` alert is raised to indicate the operational state of NRF is COMPLETE_SHUTDOWN.
- The operational state change is recorded and can be viewed using the `controlledShutdownOptions` REST API. Additionally, the history of operational state changes can be viewed using the `operationalStateHistory` REST API.

From COMPLETE_SHUTDOWN to NORMAL

- The Ingress Gateway microservice resumes processing all incoming requests.
- The Egress Gateway microservice resumes processing all outgoing requests.
- The NrfAuditor pod waits for a preconfigured waiting period before resuming its audit procedures to allow the NFs to move back to this NRF after the NRF has changed to the NORMAL operational state. Once the waiting period has expired, the audit procedures resume, and the `OcnrfAuditOperationsPaused` alert is cleared. The waiting period is defined as

  ```
  (defaultHbTimer * nfHeartBeatMissAllowed) + maxReplicationLatency(s) +
  replicationLatencyThreshold * latencyMultiplier
  ```

  Where,

  - `defaultHbTimer` - defaultHbTimer configured in `nfManagementOptions.nfHeartBeatTimers` where the nfType is "ALL_NF_TYPE". For more information about the parameter, see "NF Management Options" in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
  - `nfHeartBeatMissAllowed` - nfHeartBeatMissAllowed configured in `nfManagementOptions.nfHeartBeatTimers` where the nfType is "ALL_NF_TYPE". For more information about the parameter, see "NF Management Options" in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

  - `maxReplicationLatency` - The maximum replication latency detected across all the sites. This parameter supports dynamic value.
  - `replicationLatencyThreshold` - The replicationLatencyThreshold configured in geoRedundancyOptions. For more information about the parameter, see "Georedundancy Options" in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
  - `latencyMultiplier` - A preconfigured fixed offset value set as 3.
- `OcnrfOperationalStateCompleteShutdown` alert is cleared to indicate that the operational state of NRF is NORMAL.
- The operational state change is recorded and can be viewed using the `controlledShutdownOptions` REST API. Additionally, the history of operational state changes can be viewed using the `operationalStateHistory` REST API.

NRF Behavior Post Fault Recovery

During the database restore procedure, along with the configuration data, the NFProfile data/ subscription data also gets restored which may not always be the latest state data at that moment. In this state, the NrfAuditor microservice may act upon the NFProfiles/ NFsubscriptions which are not up-to-date yet. Also, the NFs is in process of moving to the current NRF which is available now. In this state, if the audit procedure is performed, NRF suspends those NFs and send out notifications to the consumer NFs. The same is applicable for NfSubscriptions, where the subscriptions may get deleted due to an older lastUpdatedTimestamp in the backup data.

To avoid this problem, NrfAuditor microservice waits for a waiting period before resuming the auditing of NFProfiles and subscriptions as soon as it comes to Ready state from NotReady state. An alert ("`OcnrfAuditOperationsPaused`") is raised to indicate that the audit processes have paused, see NRF Alerts section. Once the waiting period is elapsed, the audit processes resume and the alert is cleared.

To know about the computation of the waiting period, see Controlled Shutdown of NRF .

> **Note**
>
> The NrfAuditor pod goes to NotReady state whenever it loses connectivity with the database. During temporary connectivity fluctuations, the NrfAuditor pod may transition between Ready and NotReady states cause the cool off period to kick in for every NotReady to Ready transition. To avoid such short and frequent transitions, NrfAuditor microservice applies the waiting period only when the pod is in NotReady for more than 5 seconds.

Managing Controlled Shutdown of NRF

Prerequisites

The following parameters are required in the `ocnrf_custom_values_25.1.201.yaml` file to configure the feature.

- `global.enableControlledShutdown` is the flag to enable the feature. The default value is true. To disable the feature, set the value of the flag to false.
  Please note that the below configurations are required for the feature to work.
- `controlled shutdown filter` under routesConfig.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> The route configuration must not be modified.

For more information about the `global.enableControlledShutdown` attribute, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide.*

Configure
You can configure the Controlled Shutdown of NRF feature using the REST API or Console:

- **REST API:** Perform the following feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
  - `{apiRoot}/nrf/nf-common-component/v1/igw/errorcodeprofiles`
  - `{apiRoot}/nrf/nf-common-component/v1/igw/controlledshutdownerrormapping`
  - `{apiRoot}/nrf-configuration/v1/controlledShutdownOptions`
  - `{apiRoot}/nrf-configuration/v1/operationalStateHistory`
- **CNC Console**:
  - You can configure error code profiles as described in the Error Code Profile section.
  - You can configure error mapping as described in the Controlled Shutdown Error Mapping section.
  - You can change only the Operational State and view the Operational State History as described in the Controlled Shutdown section.

Observe

Metrics
Following are the controlled shutdown feature specific metrics:

- ocnrf_operational_state
- ocnrf_audit_status

For more information about the metrics for the controlled shutdown of NRF see the NRF Metrics section.

Alerts
Following are the controlled shutdown feature specific alerts:

- OcnrfOperationalStateCompleteShutdown
- OcnrfAuditOperationsPaused

For more information about the alerts raised for the controlled shutdown of NRF see the Controlled Shutdown of NRF Feature section.

KPIs
Following are the controlled shutdown feature specific KPIs:

- Operational State {{ pod }}
- NRF Audit status

For more information about the KPIs in NRF see the Controlled Shutdown of NRF section.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see "Collecting Logs" section in *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.19 User-Agent Header for Outgoing Requests

NRF supports the addition of User-Agent header for outgoing messages for NFStatusNotify and SLF query requests. NRF adds the User-Agent header with the configured value to the mentioned outgoing messages.

In addition, NRF propagates the User-Agent header received in all forwarding and roaming requests.

Managing User-Agent Header for Outgoing Requests

Configure

You can configure User-Agent header value feature using the REST API or CNC Console.

- **Configure using REST API**: Provide the value for `ocnrfUserAgentHeader` in **generalOptions** configuration API. For more information about this API, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure using CNC Console**: Provide the value for **OCNRF User-Agent Header** on the General Options Page. For more information about the field, see General Options page.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts persist, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see "Collecting Logs" section in *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.20 Support for Kubernetes Resource

4.20.1 Node Selector

The Node Selector feature allows the Kubernetes scheduler to determine the type of nodes in which the NRF pods are scheduled, depending on the predefined node labels or constraints.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Node selector is a basic form of cluster node selection constraint. It allows you to define the node labels (constraints) in the form of key-value pairs. When the nodeSelector feature is used, Kubernetes assigns the pods to only the nodes that match with the node labels you specify.

To see all the default labels assigned to a Kubernetes node, run the following command:

```
kubectl describe node <node_name>
```

Where,

`<node_name>` is the name of Kubernetes node.

For example:

```
kubectl describe node pollux-k8s-node-1
```

Sample output:

```
Name:               pollux-k8s-node-1
Roles:              <none>
Labels:             beta.kubernetes.io/arch=amd64
                    kubernetes.io/hostname=pollux-k8s-node-1
                    kubernetes.io/os=linux
                    topology.kubernetes.io/region=RegionOne
                    topology.kubernetes.io/zone=nova
```

Managing NRF Node Selector Feature

Enable

You can enable Node Selector feature using the Helm:

1. Open the `ocnrf_custom_values_25.1.201.yaml` file.
2. Set `global.nodeSelection` to **ENABLED** to enable the node selection.
   For more information about the parameters, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
3. Save the file.
4. Install NRF. For more information about installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Configure

You can configure the node selector parameters using Helm.

- Configure the following parameters in the global, NRF microservices and Gateways sections:
  - `nodeSelection`
  - `helmBasedConfigurationNodeSelectorApiVersion`
  - `nodeSelector.nodeKey`
  - `nodeSelector.nodeValue`
- Configure the following parameters in the appinfo section:

Copyright © 2019, 2025, Oracle and/or its affiliates.

  - `nodeSelection`
  - `helmBasedConfigurationNodeSelectorApiVersion`
  - `nodeSelector`
- Configure the following parameters in the perfinfo section:
  - `nodeSelectorEnabled`
  - `helmBasedConfigurationNodeSelectorApiVersion`
  - `nodeSelectorKey`
  - `nodeSelectorValue`

  For more information about the parameters, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.20.2 Kubernetes Probes

One of the key feature that Kubernetes provides is high availability. This is achieved by the smallest deployment model called Pods. The health check of these Pods are performed by Kubernetes Probes.

There are three types of probes:

- **Liveness Probe**: Indicates if the container is operating. If the container is operating, no action is taken. If not, the kubelet kills and restarts the container.
- **Readiness Probe**: Indicates whether the application running in the container is ready to accept requests. If the application is ready, services matching the pod are allowed to send traffic to it. If not, the endpoints controller removes the pod from all matching Kubernetes Services.
- **Startup Probe**: Indicates whether the application running in the container has started. If the application is started, other probes start functioning. If not, the kubelet kills and restarts the container.

Managing Kubernetes Probes Feature

Configure

You can configure Kubernetes Probes feature using the Helm:

1. Open the `ocnrf_custom_values_25.1.201.yaml` file.
2. Configure the probes for each microservices. For more information about configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
3. Save the file.

Copyright © 2019, 2025, Oracle and/or its affiliates.

4. Install NRF. For more information about installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
5. If you are configuring these parameter after NRF deployment, upgrade NRF. For more information about upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.20.3 Pod Disruption Budget

PodDisruptionBudget (PDB) is a Kubernetes resource that allows you to achieve high availability of scalable application services when the cluster administrators perform voluntary disruptions to manage the cluster nodes.

PDB restricts the number of pods that are down simultaneously from voluntary disruptions. Defining PDB is helpful to keep the services running undisrupted when a pod is deleted accidentally or deliberately. PDB can be defined for high available and scalable NRF services.

It allows safe eviction of pods when a Kubernetes node is drained to perform maintenance on the node. It uses the `maxPdbUnavailable` parameter specified in the Helm chart to determine the maximum number of pods that can be unavailable during a voluntary disruption.

Managing Pod Disruption Budget

Enable

This feature is enabled automatically if you are deploying NRF with Release 16.

Configure

You can configure this feature using Helm. For information about configuring PDB, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Observe

There is no specific metrics and alerts required for the Support of PDB functionality.

4.20.4 Network Policies

NetworkPolicies are an application-centric construct that allows you to specify how a pod is allowed to communicate with various network entities. To control communication between the cluster's pods and services and to determine which pods and services can access one another inside the cluster, it creates pod-level rules.

Previously, NRF had the privilege to communicate with other namespaces, and pods of one namespace could communicate with others without any restriction. Now, namespace-level

Copyright © 2019, 2025, Oracle and/or its affiliates.

isolation is provided for the NRF pods, and some scope of communications is allowed between the NRF and pods outside the cluster. The network policies enforces access restrictions for all the applicable data flows except communication from Kubernetes node to pod for invoking container probe.

Managing Support for Network Policies

Enable

To use this feature, network policies need to be applied to the namespace in which NRF is deployed.

Configure

You can configure this feature using Helm. For information about configuring PDB, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Observe

There is no specific metrics and alerts required for the Network Policy feature.

4.20.5 Tolerations

Taints and tolerations are Kubernetes mechanism that allows you to ensure that pods are not placed on inappropriate nodes. Taints are added to nodes, while tolerations are defined in the pod specification. One or more taints are applied to a node, this marks that the node should not accept any pods that do not tolerate the taints.

When a taint is assigned by Kubernetes configurations, it repels all the pods except those that have a matching toleration for that taint. Tolerations are applied to pods to allow the pods to schedule onto nodes with matching taints. Tolerations allow scheduling but do not guarantee scheduling. For more information about the Kubernetes documentation.

Managing Kubernetes Tolerations

Enable

To enable the tolerations perform the following:

1. Open the `ocnrf_custom_values_25.1.201.yaml` file.
2. Set `global.tolerationsSetting` to **ENABLED** to enable toleration.
   For more information about the parameters, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
3. Save the file.
4. Install NRF. For more information about installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Configure

You can configure the tolerations parameters using Helm.

1. Configure `global.tolerations` as per the requirement based on the taints on the node.
   For more information about the parameters, see the Tolerations table.
2. By default `global.tolerations` is applied to all the microservices. In case you want to modify the tolerations for a specific microservice, configure `tolerationsSetting` under the specific microservice.

Copyright © 2019, 2025, Oracle and/or its affiliates.

For more information about the parameters, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Table 4-16 Tolerations

Parameter	Description
`key`	It is name of the key.
`value`	It is a value for the configured key.
`effect`	Indicates the taint effect applied for the node. The effect is defined by one of the following: • NoSchedule: – New pods that do not match the taint are not scheduled onto that node. – Existing pods on the node remain. • PreferNoSchedule: – New pods that do not match the taint might be scheduled onto that node, but the scheduler tries not to. – Existing pods on the node remain. • NoExecute: – New pods that do not match the taint cannot be scheduled onto that node. – Existing pods on the node that do not have a matching toleration are removed.
`operator`	Indicates the criteria to match the tolerations with the taint configuration. The value can be: • Equal: The key, value, and effect parameters must match with the taint configuration. • Exists: The key and effect parameters must match the taint configuration. The value parameter can be blank.

Observe

There are no specific metrics and alerts required for this feature.

4.20.6 Dual Stack

According to RFC 4213, Dual stack supports both versions of the Internet Protocol (IPv4 and IPv6).

Dual stack provides:

- coexistence strategy that allows hosts to reach IPv4 and IPv6 simultaneously.
- IPv4 and IPv6 allocation to the Kubernetes clusters during cluster creation. This allocation is applicable for all the Kubernetes resources unless explicitly specified during the cluster creation.

Using the dual stack mechanism, NRF communicates within services or deployments in a Kubernetes cluster using IPv4 or IPv6 or both simultaneously depending on the configured deployment mode. Previously, NRF was supporting only dual stack with IPv4 preferred deployment on a platform configured as dual stack with IPv4 preferred.

From 25.1.100, NRF also supports dual stack with IPv6 preferred deployment on a platform configured as dual stack with IPv4 preferred. This can be achieved by configuring the ipFamilyPolicy and ipFamilies based on the deployment mode for all the NRF microservices. For more information about the attributes, see Table 4-18.

Copyright © 2019, 2025, Oracle and/or its affiliates.

In a 5G network, for a dual stack based deployment, NRF microservices can be grouped as follows:

- Edge Microservices: The edge microservice is Ingress Gateway microservice which handles the incoming requests from other NFs. NRF supports dual stack for Ingress Gateway microservice. The Ingress Gateway microservice can accept any IP address types (IPv4 or IPv6) from other NFs.

> **Note**
>
> Ingress Gateway microservice establishes connection with NRF backend microservices on a single stack with IPv6 preferred.

- Backend Microservices: All other microservices except the Ingress Gateway microservice is considered as backend microservice. NRF supports single stack IPv4 or IPv6 for backend microservices. All the backend microservices interacts with each other using single stack.
- Egress Gateway microservice provide options for sending outgoing request using IPv4 or IPv6. The preferred routing mode can be configured using `global.egressRoutingMode` Helm parameter. For more information about this parameter, see the "Customizing NRF" section in the *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

> **Note**
>
> The IP type defined in Dynamic SLF feature (`preferredRoutingParameter.routingParameter`) should match with the IP type configured in Egress Gateway microservice dual stack preference.

IP Address Allocation to Pods

IP address allocation to pods depends on the IP address preference which is set in the Kubernetes cluster. Pods do not have the privilege to choose an IP address. For example, if the Kubernetes cluster has IPv4 preferred configuration, both IPv4 and IPv6 are allocated to the pod, but the primary address is IPv4.

Example of a pod with primary address as IPv4 deployed in an IPv4 preferred infrastructure:

```
Status: Running
IP: 10.xxx.xxx.xxx
IPs:
 IP: 10.xxx.xxx.xxx
 IP: fd00::1:1xxx:9xxx:bxxx:fxxx
```

IP address allocation to pods depends on the IP address preference set in the Kubernetes cluster. Pods do not have the privilege to choose an IP address. For example, if the Kubernetes cluster has IPv6 preferred configuration, both IPv4 and IPv6 are allocated to the pod, but the primary IP address is IPv6.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Example of a pod with primary address as IPv6 deployed in an IPv4 preferred infrastructure:

```
Status: Running
IP: fd00::1:1xxx:9xxx:bxxx:fxxx
IPs:
 IP: fd00::1:1xxx:9xxx:bxxx:fxxx
 IP: 10.xxx.xxx.xxx
```

IP Address Allocation to Services

IP address allocation to all the NRF services, depends on the NRF deployment mode as defined as per Supported Deployment Mode Table. When the Helm parameters are configured appropriately, NRF automatically configures IP Family Policy and IP Families attributes to the NRF services.

Table 4-17 Supported Deployment Mode

Supported Mode	Required Configurations in Helm		
	Edge Deployment Mode (global.edgeDeploymentMode)	**Backend Deployment Mode (global.backEndDeploymentMode)**	**Egress Routing Mode (global.egressRoutingMode)**
IPv4 Only	IPv4	IPv4	IPv4
IPv6 Only	IPv6	IPv6	IPv6
IPv4 over IPv6	IPv4_IPv6	IPv6	any
IPv6 over IPv4	IPv6_IPv4	IPv6	any
ClusterPreferred	ClusterPreferred	ClusterPreferred	None

For more information about the parameters, see the "Customizing NRF" section in the *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

You can customize the IP address allocation to services based on the above Helm parameters. Services route the traffic to the destination endpoints based on this configuration. If the Helm parameter is set to IPv4, then IPv4 is allocated to services, and services use IPv4 pod IPs to send the traffic to endpoints.

IP Family Policy Attribute, IP Families Attribute, Pod IP and Service IP indicates the Ingress Gateway Microservice whereas the Endpoint belongs to backend microservice.

IP address allocation for the edge microservice (Ingress Gateway Microservice) is configured using the `global.edgeDeploymentMode` Helm parameter.

IP address allocation for all the backend microservices are performed based on the value configured at the `global.backEndDeploymentMode` Helm parameter.

IP selection control and connections for the Egress Gateway microservice or the outgoing traffic is handled by the `global.egressRoutingMode` Helm parameter.

For more information about the above parameters, see the "Customizing NRF" section in the *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

The following table describes how IP address allocation, ipFamilyPolicy, and ipFamilies vary based on the above parameter configuration for services:

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-18 IP Address Allocation

Infrastructure Preference	Backend Deployment Mode Helm Parameter	ipFamilyPolicy Attribute	ipFamilies Attribute	Pod IP	Service IP	Endpoints
IPv4 Preferred	IPv4	SingleStack	IPv4	Ipv4,IPv6	IPv4	IPv4
IPv6 Preferred	IPv4	SingleStack	IPv4	Ipv6,IPv4	IPv4	IPv4
IPv4 Preferred	IPv6	SingleStack	IPv6	Ipv4,IPv6	IPv6	IPv6
IPv6 Preferred	IPv6	SingleStack	IPv6	Ipv6,IPv4	IPv6	IPv6
IPv4 Preferred	IPv4_IPv6(IPv4 Preferred)	RequiredDualStack	IPv4 Preferred	Ipv4,IPv6	Ipv4,IPv6	IPv4
IPv6 Preferred	IPv4_IPv6(IPv4 Preferred)	RequiredDualStack	IPv4 Preferred	Ipv6,IPv4	Ipv4,IPv6	IPv4
IPv4 Preferred	IPv6_IPv4(IPv6 Preferred)	RequiredDualStack	IPv6 Preferred	Ipv4,IPv6	Ipv6,IPv4	IPv6
IPv6 Preferred	IPv6_IPv4(IPv6 Preferred)	RequiredDualStack	IPv6 Preferred	Ipv6,IPv4	Ipv6,IPv4	IPv6

> **Note**
>
> NRF defines all the `global.edgeDeploymentMode` and `global.backEndDeploymentMode` as ClusterPreferred by default. This helps NRF or other microservices to fallback to the single stack support.

Upgrade Impact

In a dual stack environment, all the NRF backend services are configured as single stack. As per Kubernetes limitation, NRF upgrade is not supported when there is a mismatch in ipFamilies configured between NRF backend microservices and the deployment mode ipFamilies configuration. In this case, NRF recommends a fresh installation to change the preferred ipFamilies. However, this is not applicable for the edge microservice (Ingress Gateway microservice) as they can choose to increase the list of ipFamilies by setting it as dual stack according to the cluster preference.

Managing Dual Stack

Enable

The dual stack support is a core functionality of NRF. You do not need to enable or disable this feature.

Configure

This section lists the configuration details for this feature.

Helm

1. Customize the `ocnrf_custom_values_25.1.100.yaml` file.
2. Set `global.edgeDeploymentMode` to *clusterPreferred* to indicate the edge deployment mode.
3. Set `global.backEndDeploymentMode` to *clusterPreferred* to indicate the backend deployment mode.

Copyright © 2019, 2025, Oracle and/or its affiliates.

4. Set `global.egressRoutingMode` to *None* to control the IP address selection and connection for the Egress Gateway microservice. For more information about the parameters, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
5. Save the file.
6. Install NRF. For more information about installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Observe

Metrics

The following metrics are added in the Ingress Gateway Metrics section:

- `oc_ingressgateway_incoming_ip_type`
- `oc_ingressgateway_outgoing_ip_type`

The following metrics are added in the Egress Gateway Metrics section:

- `oc_egressgateway_incoming_ip_type`
- `oc_egressgateway_outgoing_ip_type`
- `oc_egressgateway_dualstack_ip_rejected_total`

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see "Collecting Logs" section in *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.21 Pod Protection Support for NRF Subscription Microservice

The NRF subscription microservice is responsible for the following service operations:

- NfStatusSubscribe
- NfStatusUnsubscribe
- NfStatusNotify

Of the above service operations, NfStatusSubscribe and NfStatusUnsubscribe requests are received by the subscription pod through the Ingress Gateway service. The registration (nfregistration) and auditor (nrfauditor) pods trigger notification events towards the subscription pod, which in turn triggers the NfStatusNotify requests to the consumer NFs.

The subscription pod currently undergo a risk of the congested condition. A congested condition is defined as a state where the pod's resource utilization (CPU and Pending Message count) is higher than the expected thresholds. This would result in higher latency, pod restarts, or traffic loss.

This situation may occur due to:

Copyright © 2019, 2025, Oracle and/or its affiliates.

- increase of notification events received from the registration and auditor pods.
- increase of NfStatusSubscribe or NfStatusUnsubscribe events.
- a large number of NfStatusNotify events being triggered.

The Overload Control feature through Ingress Gateway only ensures the traffic through the Ingress Gateway to the subscription pods is regulated. For more information about the Overload Control feature, see Overload Control.

However, in the NRF traffic call model, this traffic constitutes only 1% of the traffic to this service while 99% of the traffic is because of the NfStatusNotify events. Hence, the overload feature alone does not prevent the NRF subscription microservice pod from going into the congested state. The Pod Protection feature is introduced as a solution to address the overload situation by protecting the pod independently and continuing to provide service.

> **Note**
>
> Horizontal POD Autoscaling (HPA) at subscription microservice and pod protection mechanism at subscription microservice are independent features with different trigger conditions. While HPA considers microservice load, pod protection mechanism works only on the POD load. Therefore, their order of triggering cannot be predicted.

NRF monitors the congestion state for the subscription microservice pod. The congestion state is monitored by the following parameters:

- CPU: The CPU consumption of the Subscription microservice container is used to determine the congestion level. The CPU usage is monitored using the Kubernetes cgroup (cpuacct.usage) and it is measured in nanoseconds.
  It is monitored periodically and calculated using the following formula and then compared against the configured CPU thresholds to determine the congestion state.

  Figure 4-20 CPU Measurement

  $$\frac{\frac{\text{CurrentCpuUsage - LastCpuUsage}}{\text{CurrentTime - LastSampleTime}} \times 100}{\text{CPUs}}$$

  Where,

  – `CurrentCpuUsage` is the counter reading at current periodic cycle.
  – `LastCpuUsage` is the counter reading at previous periodic cycle.
  – `CurrentTime` is the current time snapshot.
  – `LastSampleTime` is the previous periodic cycle time snapshot.
  – `CPUs` is the total number of CPUs for a given pod.
- Pending Message Count: The pending message count is the number of requests that are received by the subscription pod and yet to send the response. This includes all the requests triggered towards the Subscription pods from the registration, auditor, and Ingress Gateway microservices. The processing of the requests may include DB operations, forwarding the request to another NRF, and triggering notifications. The pending message

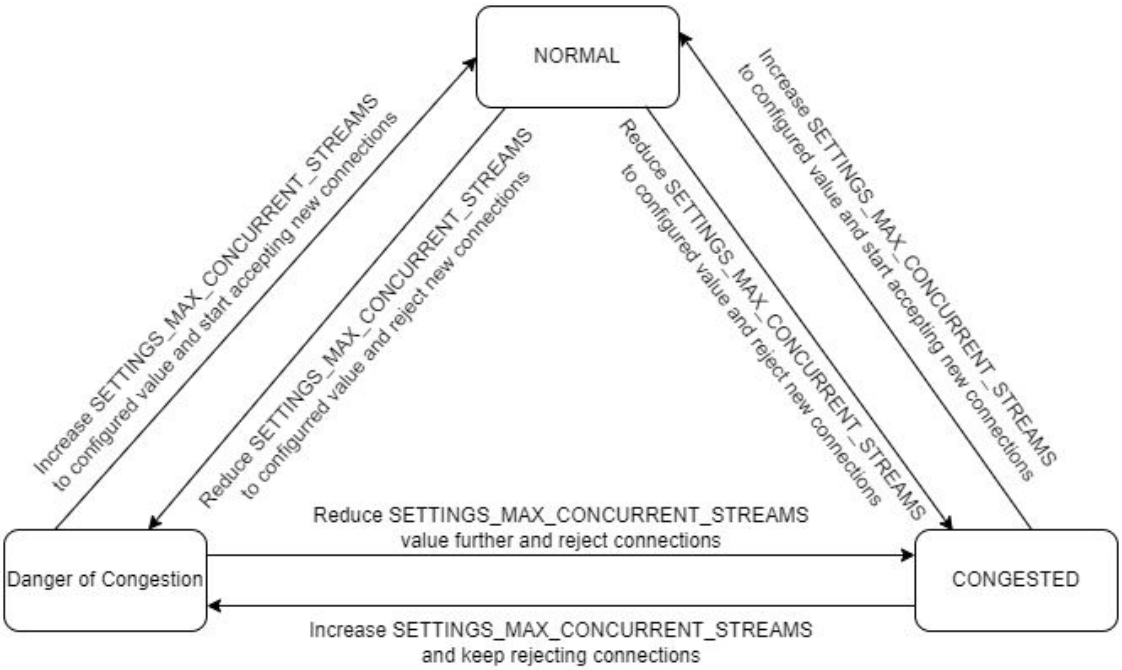
Copyright © 2019, 2025, Oracle and/or its affiliates.

count is monitored periodically and then compared against the configured thresholds to determine the congestion state.

Subscription Microservice Load States

Following are the different overload states to detect overload conditions. This will protect and mitigate the pod entering into an overload condition, and to take necessary actions to recover from overload.

Figure 4-21 Pod Protection State Transition



> **Note**
>
> The transition can occur between any states based on the congestion parameters. These congestion parameters are preconfigured and cannot be changed.

- Congested State: This is the upper bound state where the pod is congested. This means one or more congestion parameters is above the configured thresholds for the congested state. For more information about the configuration using CNC Console, see Pod Protection Options. The pod can be transitioned to the Congested State either from the Normal State or the DoC state.
  When pod reaches this state, the following actions are performed:
  - new incoming HTTP2 connection requests are not accepted.
  - the pod gradually decrements the number of concurrent streams by updating SETTINGS_MAX_CONCURRENT_STREAMS parameter in a SETTINGS frame to the configured `maxConcurrentStreamsPerCon` value at a regular interval. The concurrent streams are decremented based on the value configured in `decrementBy` parameter. And, the regular interval is configured in the `decrementSamplingPeriod` parameter.

  Alerts are raised when any of the subscription pods go into this state. For more information about alerts, see NRF Alerts.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- Danger of Congestion (DoC) State: This is the intermediate state where the pod is approaching a congested state. This means one or more congestion parameters, CPU or Pending Message Count, is above the configured thresholds for the DoC state. For more information about the configuration using CNC Console, see Pod Protection Options.
  - When pod reaches this state, the following actions are performed:
    * any new incoming HTTP2 connection requests are not accepted.
    * if the pod is transitioning from the Normal State to the DoC state, the pod gradually decrements the number of concurrent streams by updating SETTINGS_MAX_CONCURRENT_STREAMS parameter in a SETTINGS frame to the configured `maxConcurrentStreamsPerCon` value at a regular interval. The concurrent streams are decremented based on the value configured in `decrementBy` parameter. And, the regular interval is configured in the `decrementSamplingPeriod` parameter.
    * if the pod is transitioning from the Congested State to the DoC state, the pod gradually increments the number of concurrent streams by updating SETTINGS_MAX_CONCURRENT_STREAMS parameter in a SETTINGS frame to the configured `maxConcurrentStreamsPerCon` value at a regular interval. The concurrent streams are incremented based on the value configured in `incrementBy` parameter. And, the regular interval is configured in the `incrementSamplingPeriod` parameter.
- Normal State: This is the lower bound state where all the congestion parameters for the pod are below the configured thresholds for DoC and Congested states. For more information about the configuration using CNC Console, see Pod Protection Options. When pod reaches this state, the following actions are performed:
  - the pod will continue accepting new incoming HTTP2 connection requests.
  - in case the pod is transitioning from the Congested or DoC state to Normal state, the pod gradually increments the number of concurrent streams by updating SETTINGS_MAX_CONCURRENT_STREAMS parameter in a SETTINGS frame to the configured `maxConcurrentStreamsPerCon` value at a regular interval. The concurrent streams are incremented based on the value configured in `incrementBy` parameter. And, the regular interval is configured in the `incrementSamplingPeriod` parameter.

To avoid toggling between these states due to traffic pattern, it is required for the pod to be in a particular state for a given period before transitioning to another state. The below configurations are used to define the period till which the pod has to be in a particular state:

- `stateChangeSampleCount`
- `monitoringInterval`

Formula for calculating the period is: `(stateChangeSampleCount * monitoringInterval)`

When the Subscription pods are in DoC or Congested state, it forces the client pods, registration, auditor, and Ingress Gateway to send lesser traffic towards the subscription pod. Subscription pod receives traffic in following scenarios:

- The registration pod sends a notification trigger request to the subscription pods when:
  - an NF registers itself with NRF.
  - an NF updates its registered profiles.
  - an NF deregisters itself from NRF.
- The auditor pod sends a notification trigger request to the subscription pods when:
  - the auditor marks an NF as SUSPENDED.

Copyright © 2019, 2025, Oracle and/or its affiliates.

  - the auditor deregisters an NF when an NF is SUSPENDED state for a configurable amount of time.
- The subscription pod receives the below requests through the Ingress Gateway service when:
  - the consumer NF creates a new subscription in NRF using NFStatusSubscribe request.
  - the consumer NF updates an existing Subscription in NRF using NFStatusSubscribe request.
  - the consumer NF unsubscribes using NFStatusUnsubscribe request.

  If the registration and auditor pods are unable to trigger a notification request to the subscription pods since they are in DoC or Congested State, It is possible that the consumer NFs will not be notified about the particular change in the consumer NfProfile. However, this is mitigated as the consumer NFs are expected to perform rediscovery of the producer NFs whenever the discovery validity time expires. This will ensure that their producer NF data will be refreshed, despite the dropped NfStatusNotify message.

  The registration pod will not reject any of the incoming requests, like NfRegister, NfUpdate, NfDeregister, and NfHeartbeat even if it is unable to trigger a notification to the subscription pods for the same reason.

  It is expected that the requests through the Ingress Gateway to the congested subscription pods may timeout, or get rejected.

Managing Subscription Microservice Pod Protection

This section explains the procedure to enable and configure the feature.

Enable

You can enable the Pod Protection feature using the CNC Console or REST API.

- **Enable using REST API**: Set `podProtectionOptions.enabled` to **true** and `podProtectionOptions.congestionControl.enabled` to **true** in Pod Protection Options configuration API. For more information about API path, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Set `Enabled` to **true** for Pod Protection and `Enabled` to **true** for Congestion Control in Pod Protection Options page. For more information about enabling the feature using CNC Console, see the Pod Protection Options section.

Observe

Metrics

Following metrics are added in the Pod Protection Metrics section:

- ocnrf_pod_congestion_state
- ocnrf_pod_cpu_congestion_state
- ocnrf_pod_pending_message_count_congestion_state
- ocnrf_incoming_connections
- ocnrf_max_concurrent_streams
- ocnrf_pod_cpu_usage
- ocnrf_pod_pending_message_count
- ocnrf_pod_incoming_connection_rejected_total
- ocnrf_nfNotification_trigger_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

Alert
Following alerts are added in the NRF Alerts section:

- OcnrfPodInDangerOfCongestionState
- OcnrfPodCpuUsageInDangerOfCongestionState
- OcnrfPodPendingMessageCountInDangerOfCongestionState
- OcnrfPodInCongestedState
- OcnrfPodCpuUsageInCongestedState
- OcnrfPodPendingMessageCountInCongestedState

KPIs
Added the feature specific KPIs in the Subscription Pod Protection section.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.22 Pre and Post Install/Upgrade Validations

This feature applies validation checks on the infrastructure, application, databases, and its related tables before and after the installation, upgrade, or fault recovery.

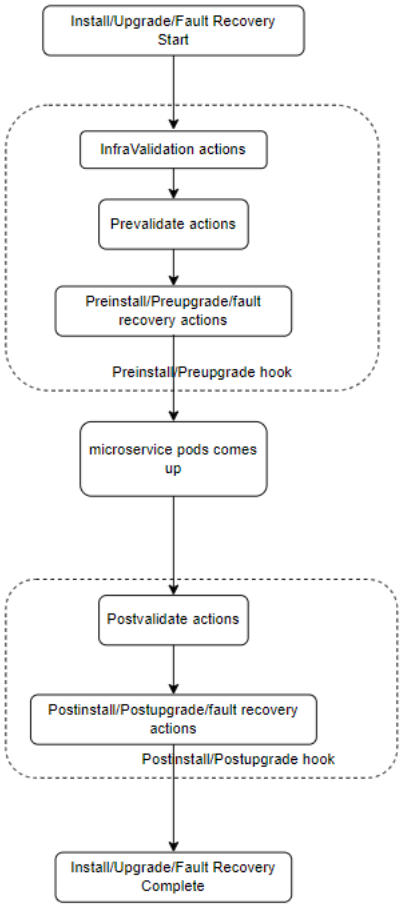
When NRF is deployed, there could be inconsistencies and unexpected results if the required NRF tables or the site-specific configurations are not available. This could happen due to network issues, system issues, human error, race conditions. The feature aims to detect any inconsistency early in the system state and report the same. The validations are done during the preinstallation, postinstallation, preupgrade, and postupgrade. The infrastructure validations are performed as part of the preinstallation and preupgrade validations.

> **Note**
>
> Validation of database and tables during rollback procedures is not supported.

The below diagram depicts the flow where the validations occur:

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-22 Pre and Post Install/Upgrade Validations



Database and its Related Tables Validation

Each microservice is responsible for validating the database and the tables that it creates and manages.

The following tables are validated by each microservice:

- NrfConfiguration Service validates the following tables in the NRF Application database:
  - NrfSystemOptions
  - NfScreening
  - SiteIdToNrfInstanceIdMapping
  - NrfEventTransactions
- NfRegistration Service validates the following tables in the NRF Application database:
  - NfInstances
  - NfStatusMonitor
- NfSubscription service validates the following table in the NRF Application database:
  - NfSubscriptions
- NrfAuditor service validates the following table in NRF Leader Election database:
  - NrfAuditorLeaderPod

Copyright © 2019, 2025, Oracle and/or its affiliates.

- NrfConfiguration Service validates the following tables in the NRF Network Database:
  - NfScreening_backup
  - NrfSystemOptions_backup
  - SiteIdToNrfInstanceIdMapping_backup

> **Note**
>
> - The common configuration database and its tables are currently not validated in the preinstallation, fault recovery, and postinstallation validation.
> - The schema of the ReleaseConfig table is not currently validated in the preinstallation and postinstallation validation.

NRF Pre and Post Validations

NRF performs the following validation checks:

- NRF Infrastructure Validation
- NRF Preinstallation Validation
  - For Fresh Installation
  - For Fault Recovery
- NRF Postinstallation Validations
  - For Fresh Installation
  - For Fault Recovery
- NRF Preupgrade Validation
- NRF Postupgrade Validation

Managing Pre and Post Install/Upgrade Validations Feature

Enable

You can enable the feature using the Helm:

1. Open the `ocnrf_custom_values_25.1.201.yaml` file.
2. Set `global.appValidate.preValidateEnabled` to **true** to validate the database tables during preinstallation and preupgrade.
3. Set `global.appValidate.postValidateEnabled` to **true** to validate the database tables during postinstallation and postupgrade.
4. Set `global.appValidate.infraValidateEnabled` to **true** to validate the database tables during infrastructure validation. Configure the following appInfo attributes:
   a. Configure `replicationUri` with the database monitoring service FQDN or port as per the deployment. The URI must be provided as "http://<db monitor service name>:<db monitor service port>/db-tier/status/replication/realtime".
   b. Configure `dbTierVersionUri` with the database monitoring service FQDN or port to retrieve the cnDBTier version. The URI must be provided as "`http://<db monitor service name>:<db monitor service port>/db-tier/version`".

Copyright © 2019, 2025, Oracle and/or its affiliates.

c. Configure `alertmanagerUrl` with the database monitoring service FQDN or port to retrieve the cnDBTier version. The URI must be provided as `"http://<alert manager service name>:<alert manager service port>/cluster/alertmanager"`.

5. Set `global.appValidate.faultRecoveryMode` to **true** to install NRF in fault recovery mode.
   For more information about the parameters, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
6. Save the file.
7. Install or upgrade NRF. For more information about the procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

4.22.1 NRF Infrastructure Validation

The infrastructure validations are performed as part of the preinstallation and preupgrade validations. The validation is done by the preinstallation and preupgrade hooks of the app-info service. The infrastructure validation is disabled by default.

> **Note**
>
> - It is highly recommended to enable it, and perform the required configuration to detect incompatibilities early. The infrastructure validation will be enabled by default from the next releases.
> - Infrastructure validation is not supported during rollback procedures.

This validation is enabled by setting `global.appValidate.infraValidateEnabled` parameter to **true** in the `ocnrf_custom_values_25.1.201.yaml` file. The following checks are performed as part of the infrastructure validation before installing or upgrading NRF:

- Validate the minimum viable path from the previous NRF version to the current NRF version for an upgrade. This is validated based on the value configured in the `global.minViablePath` parameter. If this minimum viable path is not supported, NRF upgrade will not proceed.
- Validate that the installed Kubernetes version is compatible with the target NRF version being installed or upgraded. This is validated based on the value configured in the `global.minKubernetesVersion` parameter. If the Kubernetes version is not compatible, NRF installation or upgrade will not proceed.
- Validate that the installed cnDBTier version is compatible with the target NRF version being installed or upgraded. This is validated based on the value configured in the `global.minDbTierVersion` parameter. If the cnDBTier version is not compatible, NRF installation or upgrade will not proceed.
- Verify the replication status with the connected peer. This is validated based on the value configured in the `appinfo.defaultReplicationStatusOnError` parameter. If the replication is disabled or failed for all peers, NRF installation or upgrade will not proceed.
- Verify that there are no critical alerts raised in the system. If critical alerts are found, installation or upgrade will not proceed.

Copyright © 2019, 2025, Oracle and/or its affiliates.

4.22.2 NRF Preinstallation Validation

The following sections describe the preinstallation validation performed when NRF is installed for fresh installation mode and fault recovery mode.

4.22.2.1 For Fresh Installation

The preinstallation validation is performed as part of the preinstallation hooks of each NRF microservice. It is the first set of actions performed. For more information about the list of microservices that perform validation, see NRF Microservice Table Validation.

This validation is configured by setting `global.appValidate.preValidateEnabled` parameter to **true** in the `ocnrf_custom_values_25.1.201.yaml`. The following checks are performed as part of the preinstallation validation:

- Validate the presence of the database. If not present, the installation will not proceed. The database is expected to be present, before proceeding with installation.
- Validate the ReleaseConfig table does not have release information about the site being installed. If present, the installation will not proceed, as it will not be a fresh installation.
- Validate the presence of the required tables. Each microservice validates the tables that it is responsible for. If all the required tables are present, then the table schema is validated against the schema expected as per the release version. If the tables are partially present or if the schema validation fails, installation will not proceed. The tables are expected to be present when multisite georedundant NRF is deployed.

4.22.2.2 For Fault Recovery

NRF can be installed in fault recovery mode by setting `global.appValidate.faultRecoveryMode` parameter to **true** in the `ocnrf_custom_values_25.1.201.yaml`. When NRF is deployed in the fault recovery mode, with a database backup in place, the following checks are performed as part of the preinstallation validation:

- Validate the presence of the database. If not present, the installation will not proceed. The database is expected to be present, before proceeding with installation.
- Validate if the ReleaseConfig table contains release information about the site. If not present, the installation will not proceed as the table is expected to contain information from the database backup.
- Validate the presence of the required tables. Each microservice validates the tables that it is responsible for. If the tables are present, then the table schema is validated against the schema expected as per the release version. If the tables are partially present or if the schema validation fails, installation will not proceed.
- Validate the configType in the NrfSystemOptions table against the configTypes expected as per the releaseVersion of the specific site. If validation fails, installation will not proceed.
- Validate the nfScreeningRulesListType in the NfScreening table against the nfScreeningRulesListType expected as per the releaseVersion of the specific site.

> **Note**
>
> It is recommended that after the installation is completed in fault recovery mode, this flag has to be set to `false`. This can be done while performing future upgrades.

Copyright © 2019, 2025, Oracle and/or its affiliates.

In case preinstallation fails, the error reason is logged in the preinstallation hook job of the particular microservice.

4.22.3 NRF Postinstallation Validations

The following sections describe the postinstallation validation performed when NRF is installed in fresh installation mode and fault recovery mode.

4.22.3.1 For Fresh Installation

The following sections describe the postinstallation validation performed when NRF is installed in fresh installation mode and fault recovery mode.

The postinstallation validation is done as part of the postinstallation hooks of each NRF microservice and is the last set of actions performed. For more information about the list of microservices that perform validation, see NRF Microservice Validation.

This validation is configured by setting `global.appValidate.postValidateEnabled` parameter to **true**. The following checks are performed as part of the postinstallation validation:

- Validate if the ReleaseConfig table contains release information about the site. If not present, the installation will not proceed.
- Validate the presence of the database. If not, the installation will not proceed.
- Validate the presence of the required tables. Each microservice validates the tables that it is responsible for. If all the required tables are present, then the table schema is validated against the schema expected as per the release version. If the tables are partially present or if the schema validation fails, installation will not proceed. The tables are expected to be present when multisite georedundant NRF is deployed.
- Validate the configType in the NrfSystemOptions table against the configTypes expected as per the releaseVersion of the specific site. If validation fails, installation will not proceed,
- Validate the nfScreeningRulesListType in the NfScreening table against the nfScreeningRulesListType expected as per the releaseVersion of the specific site.

The tables are expected to be present when multisite georedundant NRF is deployed.

In case postinstallation fails, the error reason will be logged in the postinstallation hook job of the particular microservice.

4.22.3.2 For Fault Recovery

When NRF is installed in the fault recovery mode, using previously backed up database, similar actions as mentioned in NRF in Fault Recovery Mode are performed.

> **Note**
>
> It is recommended that after the installation is complete in fault recovery mode, this flag has to be set to `false`. This can be done while performing future upgrades.

4.22.4 NRF Preupgrade Validation

The following sections describe the preupgrade validation done as part of the preupgrade hooks of each NRF microservice and is the first set of actions performed. The following checks are performed as part of the preupgrade validation.

Copyright © 2019, 2025, Oracle and/or its affiliates.

For more information about the list of microservices that perform validation, see NRF Microservice Validation.

This validation is configured by setting `global.appValidate.preValidateEnabled` parameter to **true** in the `ocnrf_custom_values_25.1.201.yaml` file. The following checks are performed as part of the preupgrade validation:

- Validate if the ReleaseConfig table contains release information about the site. If not present, the upgrade will not proceed.
- Validate the presence of the database required to proceed with the upgrade. If not present, the upgrade will not proceed.
- Validate the presence of the required tables required to proceed with the upgrade. Each microservice validates the tables that it is responsible for. If all the tables are not present or partially present, the upgrade will not proceed.
- Validate the table schema against the schema expected as per the release version. If validation fails, upgrade will not proceed.
- Validate the configType in the NrfSystemOptions table against the configTypes expected as per the releaseVersion of the specific site. If validation fails, upgrade will not proceed.
- Validate the nfScreeningRulesListType in the NfScreening table against the nfScreeningRulesListType expected as per the releaseVersion of the specific site.

The tables are expected to be present when multisite georedundant NRF is deployed.

4.22.5 NRF Postupgrade Validation

The following sections describe the postupgrade validation done as part of the postupgrade hooks of each NRF microservice and is the last set of actions performed. The following checks are performed as part of the postupgrade validation.

For more information about the list of services that perform validation, see NRF Microservice Validation.

This validation is configured by setting `global.appValidate.postValidateEnabled` parameter to **true** in the `ocnrf_custom_values_25.1.201.yaml` file. The following checks are performed as part of the postupgrade validation:

- Validate if the ReleaseConfig table contains release information about the site. If not present, the upgrade will not proceed.
- Validate the presence of the database required to proceed with the upgrade. If not present, the upgrade will not proceed.
- Validate the presence of the required tables required to proceed with the upgrade. Each microservice validates the tables that it is responsible for. If all the tables are not present or partially present, the upgrade will not proceed.
- Validate the table schema against the schema expected as per the release version. If validation fails, upgrade will not proceed.
- Validate the configType in the NrfSystemOptions table against the configTypes expected as per the releaseVersion of the specific site. If validation fails, upgrade will not proceed.
- Validate the nfScreeningRulesListType in the NfScreening table against the nfScreeningRulesListType expected as per the releaseVersion of the specific site.

The tables are expected to be present when multisite georedundant NRF is deployed.

Copyright © 2019, 2025, Oracle and/or its affiliates.

4.23 Ignore Unknown Attribute in NFDiscover Search Query

NRF by default rejects any nfDiscover request (400 Bad Request) with query attributes that are not supported. As part of this feature, instead of rejecting the request, NRF allows processing the nfDiscover request by ignoring the unsupported and unknown search query attributes. The list of query attributes that should be ignored while processing the nfDiscover request is configured using Helm.

Enable

You can enable this feature using the Helm:

1. Open the ocnrf_custom_values_25.1.201.yaml file.
2. List the search query in `searchQueryIgnoreList` under `nfdiscovery` setion. For more information about the parameter, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
3. Save the file.
4. Run *helm install.* For more information about the installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
5. If you are setting this parameter after NRF deployment, run the *Helm upgrade*. For more information about the upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

4.24 NF Authentication using TLS Certificate

This feature supports authentication of the Network Function before accessing the, NRF services. In case, authentication fails, NRF rejects the service operation requests. In this feature, NRF challenges some attributes from the TLS certificate against defined attributes.

4.24.1 XFCC Header Validation

HTTPS support is a minimum requirement for 5G NFs as defined in 3GPP TS 33.501. This feature enables extending identity validation from the Transport layer to the Application layer and provides a mechanism to validate the NF FQDN presence in Transport Layer Security (TLS) certificate as added by the Service Mesh against the NF Profile FQDN present in the request.

NRF provides configurations to dynamically enable or disable the feature. To enable the feature on Ingress Gateway in NRF deployment, see `xfccHeaderValidation` attribute in User Configurable Section of *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide* for more details.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> - This feature is disabled by default. The feature needs to be enabled at API Gateway and NRF. At NRF, the feature enabling or disabling can be performed using the following configuration.
> - Once this feature is enabled, all of NFs must re-register with FQDN in NF Profile or NFs can send NFUpdate with FQDN. For Subscription Service Operations, Network Functions must register with NRF. The NFs that are subscribed before enabling the feature, must register with NRF for further service operations.

Managing NF Authentication using TLS Certificate

Enable

To enable the feature on Ingress Gateway in NRF deployment:

1. Customize the `ocnrf_custom_values_25.1.201.yaml` file.
2. Set **enabled** parameter to `true` under XFCC header validation/extraction in Ingress Gateway Global Parameters section:

   ```
   xfccHeaderValidation:
      extract:
       enabled: false
   ```

3. Save the file.
4. Run helm upgrade, if you are enabling this feature after NRF deployment. For more information about upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Configure

Configure the NF Authentication using the TLS Certificate feature using REST API or CNC Console:

- **Configure NF Authentication using TLS Certificate using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
  Refer to attributes under `nfAuthenticationOptions` in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide* for more details.
- **Configure NF Authentication using TLS Certificate using CNC Console**: Perform the feature configurations as described in NF Authentication Options.

Observe

For more information on NF Authentication using TLS certificate feature metrics and KPIs, see NRF Metrics and NRF KPIs sections.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.24.2 TLS SNI Header Validation

Oracle Communications Cloud Native Core, Network Repository Function (NRF) supports Server Name Identification (SNI) when acting as a client and sending TLS handshake message. The FQDN in SNI is used to identify the server for which the TLS connection is needed. When the same server is supporting multiple services, the SNI identifies which service is to be used for the TLS connection. As a part of the TLS handshake initiation process, SNI is populated in the client handshake sent by Egress Gateway depending on the routing type. There are two routing options:

- Direct Routing – Egress Gateway adds the target server's FQDN in the SNI. If the target server has only IP address, then the SNI header will not be populated in the outgoing TLS requests.
- Indirect Routing – Egress Gateways uses the FQDN of the selected Peer (For example, host of the SCP or SEPP) for populating the SNI header. If the selected peer has an IP address, SNI header is not populated in the outgoing TLS requests.

> **Warning**
>
> This feature should be enabled only for non-servicemesh-based deployments.

> **Note**
>
> When the feature is enabled, SNI header is populated at Egress Gateway only when an FQDN is available.

Managing TLS SNI Header Validation

Enable

You can enable the TLS SNI header feature in NRF deployment:

1. Open the ocnrf_custom_values_25.1.201.yaml file.
2. Set `egress-gateway.sniHeader.enabled` to **true** to enable TLS SNI header validation.
3. Save the file.
4. Run *helm install.* For more information about the installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
5. If you are setting this parameter after NRF deployment, run the *Helm upgrade*. For more information about the upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Observe

For more information on TLS SNI header validation feature metrics `oc_egressgateway_sni_error_total` see NRF Metrics section.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Maintain

In case the SNI header is not sent in the TLS 1.2 handshake, perform the following:

1. **Collect the logs**: The logs are collected during the following scenarios:
   - To ensure that the SNI header is added and the feature is running as expected, look up the `SNI feature is enabled` log statement at the debug level.
   - To identify when the Peer rejected the client handshake due to invalid SNI sent by Egress Gateway, look up the `Unrecognized server name indication` log statement at the debug level.
   - To verify if both service mesh and SNI feature are enabled, look up the `Service mesh is enabled. As a result, SNI will not be populated even though it is enabled` log statement at the warn level.
     For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.25 Subscriber Location Function

Subscriber Location Function (SLF) feature of NRF allows you to select the network function based on the Subscription Permanent Identifier (SUPI) and Generic Public Subscription Identifier (GPSI) subscriber identities. The SLF feature supports Authentication Server Function (AUSF), Unified Data Repository (UDR), Unified Data Management (UDM), Policy Control Function (PCF), and Charging Function (CHF) nfTypes for discovery query.

The discovery of producer network functions is performed as follows:

- NRF checks if SLF feature is enabled or not. If the feature is enabled:
  - Check `slfLookupConfig` contains details for `target-nf-type` in NFDiscover query:
    * In case any of the NFDiscover search query attributes are present in configured `skipSLFLookupParameters` then the SLF lookup is not performed.
      For example, if the search query attribute is `group-id-list` and the same attribute is configured in `skipSLFLookupParameters`, then NRF will not perform SLF lookup rather NRF will use the `group-id-list` present in the search query while processing of NFDiscover service operation.
    * In case the configured `skipSLFLookupParameters` attribute does not match with any of the NFDiscover search query, then the mandatory parameter (either SUPI or GPSI) required for SLF lookup must be present in the NFDiscover search query.
    * In case none of the mandatory attributes (SUPI or GPSI) are present in NFDiscover search query and if NFDiscover search query attribute is present in configured `exceptionListForMissingMandatoryParameter` attribute, then NRF does not perform SLF lookup and processes the NFDiscover service operation without rejecting the NFDiscover search query.
    * In case both the mandatory attributes (SUPI or GPSI) are present in NFDiscover search query and if NFDiscover search query attribute is present in configured `exceptionListForMissingMandatoryParameter` attribute, NRF performs SLF lookup and drops the SUPI or GPSI and processes the NFDiscover service operation.

Copyright © 2019, 2025, Oracle and/or its affiliates.

    * In case none of the mandatory attributes (SUPI or GPSI) are present in NFDiscover search query and if NFDiscover search query attribute is present in configured `exceptionListForMissingMandatoryParameter` attribute, then NRF will process the NFDiscover service operation without rejecting the NFDiscover search query.
    * In case none of the mandatory attributes (SUPI or GPSI) are present in the NFDiscover search query and if NFDiscover search query attribute is not present in configured `exceptionListForMissingMandatoryParameter` attribute then NRF rejects the discovery query.
    * In case both of the mandatory attributes (SUPI or GPSI) are present in NFDiscover search query, then configured `preferredSubscriberIdType` is used to decide which of the mandatory attribute is used to perform the SLF query by ignoring the other attributes.

- Finds the NF Group Id by sending SLF query (that is, Nudr_GroupIDmap service operation) with the received Subscriber Identity (like SUPI and GPSI).
- Generates NFDiscover service response using the NFGroupId received in SLF response and other parameters. The received Subscriber Identifier is not used during NF Producer selection.
- `accessTokenCacheEnabled` flag enables caching of the oAuth2 token for SLF communication at NRF.

> **Note**
>
> – `accessTokenCacheEnabled` flag is functional only when the oAuth2 token is required for SLF communication that is controlled by the `accessTokenCacheEnabled` parameter.
>
> – Operators must enable the `accessTokenCacheEnabled` flag only after NRF deployment is successfully upgraded to 25.1.201.

NRF supports direct routing or dynamic routing of SLF queries:

- For direct routing, see Static Selection of SLF.
- For dynamic routing, see Dynamic Selection of SLF.

Managing Subscriber Location Function Feature

Enable

You can enable SLF feature using the REST API or CNC Console.

- **Enable using REST API**: Set `featureStatus` to `ENABLED` in **SLF Options** configuration API. For more information about API path, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Set **Feature Status** to **ENABLED** on the SLF Options page. For more information about enabling the feature using CNC Console, see SLF Options.

Configure

You can configure the SLF feature using the REST API or CNC Console:

- **Configure using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- **Configure using CNC Console**: Perform the feature configurations as described in SLF Options.

> **Note**
>
> - Atleast one of the attributes fqdn, ipv4Address, or ipv6Address must be included by the registered UDR.
> - For routing, the NfService level attributes are considered first. If none of the attributes are present at NfService level, then the NfProfile level attributes are considered, not in a mix mode where some of the attributes are from NfService and some from NfProfile.
>   However, with Support for Configurable Port and Routing Parameter Selection feature, routing selection parameter can be picked either from NfService or NfProfile in a mix mode as per the availability.
> - Of the three attributes, the endpoint is selected in the following order:
>   - `ipv4Address`
>   - `ipv6Address`
>   - `fqdn`
>
>   However with the Support for Configurable Port and Routing Parameter Selection feature, the order of the attributes can be configured.
> - For the cases where ports are not present, (example: ipv4Address and ipv6Address in NfProfile or FQDN from NfService/NfProfile or ipEndpoints with no port in NfService), scheme configured in the NfService is used to determine the port.
>   - If NfService.scheme is set to http, port 80 is used.
>   - If NfService.scheme is set to https, port 443 is used.
>
>   However with Support for Configurable Port and Routing Parameter Selection feature, the port selection can be configured.

Observe

Metrics

Following are the SLF feature specific metrics:

- ocnrf_nfDiscover_ForSLF_rx_requests_total
- ocnrf_nfDiscover_ForSLF_tx_responses_total
- ocnrf_SLF_tx_requests_total
- ocnrf_SLF_rx_responses_total
- ocnrf_slf_jetty_latency_seconds
- ocnrf_nfDiscover_SLFlookup_skipped_total
- ocnrf_nfDiscover_continue_mandatoryAttributes_missing_total
- ocnrf_max_slf_attempts_exhausted_total

For more information about SLF metrics, see NRF SLF Metrics section.

Alerts

Copyright © 2019, 2025, Oracle and/or its affiliates.

Following are the SLF feature specific alerts:

- OcnrfMaxSlfAttemptsExhausted

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

1. **Collect the logs**: For more information about how to collect logs, see "Collecting Logs" section in *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information about how to raise a service request.

4.25.1 Static Selection of SLF

NRF supports static selection of SLF to perform direct routing of discovery queries. These queries are sent to Unified Data Repository (UDR) or SLF (that is, Nudr_GroupIDmap service) to retrieve the corresponding NFGroupId. In the current configuration, the SLF or UDR is selected based on the `slfHostConfig` configuration and establishes direct communication with the selected SLF or UDR. For SLF query through Service Communication Proxy (SCP), see the SLF Requests section.

SLF host configuration attribute (`slfHostConfig`) allows the user to configure the details of SLF or UDR network functions.

The `slfHostConfig` configuration consists of attributes like apiVersion, scheme, FQDN, port, priority. NRF allows the configuration of more than two hosts. The host with the highest priority is considered as the Primary Host. The host with the second highest priority is considered as a Secondary Host.

> **Note**
>
> - Refer to 3GPP TS 29.510 (release 15.5) for definition and allowed range for `slfHostConfig` attribute (apiVersion, scheme, FQDN, port, priority, and so on).
> - Apart from priority attribute, no other attributes plays any role in primary or secondary host selection.
> - Apart from primary or secondary host, other configured hosts (if any) are not used during any message processing.
> - When more than one host is configured with highest priority, then two of them is picked as primary or secondary host randomly.

The SLF request is first sent to the primary SLF. In case of an error from primary SLF, the request is sent to secondary SLF based on the following configurations:

1. **rerouteOnResponseHttpStatusCodes**: This attribute is used to determine if SLF retry must be performed to an alternate SLF based on the response code received from the SLF. The alternate SLF is picked from the SLF Host Config, if `slfConfigMode` is set to STATIC_SLF_CONFIG_MODE or from the slfDiscoveredCandidateList if `slfConfigMode` is set to DISCOVERED_SLF_CONFIG_MODE. For primary and secondary SLF details, see `slfHostConfig` attribute.

Copyright © 2019, 2025, Oracle and/or its affiliates.

2. **maximumHopCount**: This configuration determines the maximum number of hops (SLF or NRF) that NRF can forward in a given service request. This configuration is useful during NRF Forwarding and SLF feature interaction.

Enable the feature in Static mode

The default configuration for `slfConfigMode` is **STATIC_SLF_CONFIG_MODE**. Perform the following steps to enabled the SLF feature in static mode:

1. Configure `slfHostConfig` and `slfLookupConfig` parameters. For more information on the parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
2. Set `featureStatus` to **ENABLED**.

4.25.2 Dynamic Selection of SLF

NRF supports dynamic selection of SLF based on the registered SLF or UDR profiles. This configuration is defined based on the SLF Configuration Mode attribute (`slfConfigMode`). This attribute allows the user to decide whether the SLF lookup can be performed based on preconfigured `slfHostConfig` configuration or use the SLFs registered with NRF.

The dynamic selection of producer network functions is as follows.

NRF checks if the SLF feature is enabled or not. When the feature is enabled:

- If `slfConfigMode` attribute is set to **STATIC_SLF_CONFIG_MODE**, the SLF lookup is performed based on preconfigured `slfHostConfig` as described in Static Selection of SLF.
- To perform SLF lookup based on the SLFs registered with NRF:
  - Set `populateSlfCandidateList` to **true**:

    ```
    ("populateSlfCandidateList": true)
    ```

  - Wait until atleast one entry of SLF candidate is listed under `slfDiscoveredCandidateList`. Perform GET operation to check if any SLF candidate is listed in the output.
  - Change `slfConfigMode` to **DISCOVERED_SLF_CONFIG_MODE**:

    ```
    {
      "featureStatus": "ENABLED",
       "slfLookupConfig": [{
            "nfType": "UDR"
    }],
      "slfConfigMode": "DISCOVERED_SLF_CONFIG_MODE"
    }
    ```

> **Note**
>
> Setting `populateSlfCandidateList` to **true** and `slfConfigMode` to **DISCOVERED_SLF_CONFIG_MODE** must be performed in two separate requests.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> Upgrade NRF to enable NrfArtisan service, if not enabled during installation. For more information on enabling Artisan Microservice, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide.*
>
> The `slfHostConfig` must be configured before or while setting the `slfConfigMode` as STATIC_SLF_CONFIG_MODE when `featureStatus` is ENABLED.
>
> Once the `featureStatus` is ENABLED, and the `slfConfigMode` is STATIC_SLF_CONFIG_MODE, the `slfHostConfig` cannot be empty.
>
> If `slfConfigMode` is set to DISCOVERED_SLF_CONFIG_MODE, `slfHostConfig` is not considered for discovery query.
>
> The `slfConfigMode` can be set to DISCOVERED_SLF_CONFIG_MODE only if there is atleast one slfCandidate present in the `slfDiscoveredCandidateList`. To trigger the population of `slfDiscoveredCandidateList`, set `populateSlfCandidateList` to **true**.

The nrfArtisan service populates the `slfDiscoveredCandidateList` by sending an nfDiscovery query to the nfDiscovery service to fetch the registered SLF or UDR NfProfiles. The discovery query used is:

```
/nnrf-disc/v1/nf-instances?target-nf-type=UDR&requester-nf-type=NRF&service-
names=nudr-group-id-map&preferred-
locality=<slfOptions.preferredSLFLocality>&limit=0
```

The nfDiscovery service fetches the registered SLF or UDR NfProfiles and apply the filtering criteria based on the discovery query parameters to get the relevant set of SLF or UDR NfProfiles. The UDR or SLF profiles are sorted and prioritized before adding to the discovery response. All the features that are available for discovery service operations, if enabled, are applied while processing the discovery query. For example: forwarding, emptylist and extended preferred locality features are included. The discovery response is sent to the nrfArtisan service which will then populate the `slfDiscoveredCandidateList`.

> **Note**
>
> If nrfArtisan service does not receive a success response from the discovery service, the `slfDiscoveredCandidateList` will contain the SLF producers that were received from the discovery service in the last successful response.

The order of the UDR or SLF profiles in the `slfDiscoveredCandidateList` list will be decided based on the sorting algorithms in nfDiscovery Service. The same is considered when NRF sends the SLF query to the SLF.

Refer Preferred Locality Feature Set for details on how the NfProfiles are sorted based on preferred-locality and extended-preferred-locality.

Prior to 22.2.x of NRF, Egress Gateway used to return 503 response for timeout exceptions (for example, No response received for SLF query or No Response received for Query sent to Forwarding NRF).

Copyright © 2019, 2025, Oracle and/or its affiliates.

From 22.3.x onwards, Egress Gateway returns 408 response codes for timeout exceptions (REQUEST_TIMEOUT and CONNECTION_TIMEOUT), to keep the behavior in line with the HTTP standard. Hence, after upgrading to 22.3.x or a fresh install of 22.3.x, if a retry needs to be performed for the timeout scenario, 408 error code explicitly needs to be configured under SLF reroute option. Following is the configuration for SLF options:

```
{
            "rerouteOnResponseHttpStatusCodes": {
              "pattern": "^[3,5][0-9]{2}$|408$"
            }
}
```

If the SLF request failed due to non-2xx error response (except 404response)/request timeouts all SLFs (and the error codes are configured under "`rerouteOnResponseHttpStatusCodes`"), then

- if forwarding is enabled and the message is a candidate to forward, NRF will forward the request to the other segment NRF.
- if forwarding is disabled, NRF will reject the discovery query with the configured error code under "SLF_Not_Reachable"

If the SLF request failed due to 404 error response and 404 is configured under "`rerouteOnResponseHttpStatusCodes`", then,

- NRF will retry to secondary or tertiary SLFs.
- forwarding is not performed in this case (there is an explicit code was added to have this functionality as per the SLF requirement).

Enable the feature in Dynamic mode

Prerequisite

Artisan Microservice (NrfArtisan) must be ENABLED. For more information on enabling alternate route service, see the "Global Parameters" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*

Configuring slfConfigMode to Dynamic

Perform the following steps to enable the SLF feature using registered SLF profiles:

1. Configure `slfLookupConfig` parameter. For more information on the parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
2. Set `populateSlfDiscoveredCandidateList` to true. This will trigger the population of `slfDiscoveredCandidateList`.
3. Perform GET on the slfOptions to see if `slfDiscoveredCandidateList` has atleast one candidate. If present, go to step 4. If not, then wait till the `slfDiscoveredCandidateList` is populated with SLF profiles.
4. Set `slfConfigMode` to **DISCOVER_SLF_CONFIG_MODE** and `featureStatus` to **ENABLED**.

Moving from Dynamic to Static

Perform the following to switch from dynamic to static configuration:

Copyright © 2019, 2025, Oracle and/or its affiliates.

1. Configure `slfHostConfig` parameter. For more information on the parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
2. Set `slfConfigMode` to **STATIC_SLF_CONFIG_MODE**, provided `slfHostConfig` and `slfLookupConfig` are already present.
3. Set `populateSlfDiscoveredCandidateList` to **false**.

Moving from Static to Dynamic

Perform the following to switch from static to dynamic configuration:

1. Upgrade NRF to enable NrfArtisan service, if not enabled previously. Configure `enableNrfArtisanService` under **global attributes** to **true** in the `ocnrf_custom_values_25.1.201.yaml` file.
2. Set `populateSlfDiscoveredCandidateList` to **true**. This triggers the population of `slfDiscoveredCandidateList`.
3. Perform GET on the slfOptions to see if `slfDiscoveredCandidateList` has atleast one candidate. If present, go to step 4. If not, then wait till the `slfDiscoveredCandidateList` is populated with SLF profiles.
4. Set `slfConfigMode` to **DISCOVER_SLF_CONFIG_MODE** and `featureStatus` to **ENABLED**.

> **Note**
>
> If `slfConfigMode` is set to **DISCOVERED_SLF_CONFIG_MODE**, `slfHostConfig` is not considered for discovery query.

4.25.3 Support for Configurable Port and Routing Parameter Selection

NRF provides a configurable option to select a port, if not configured explicitly, either from IPEndpoint or using the scheme attribute of NfService. The scenario where this configuration is used for port selection is when the routing attribute is IPv4Address/Ipv6Addresses of NfProfile or FQDN of NfService/NfProfile.

Additionally, NRF allows to configure the preferred routing attribute in case more than one of the below attribute is present in the NfProfile or NfService:

- IPv4 (ipv4Addresses from IpEndpoint of NfService if present, or ipv4Addresses of NfProfile)
- IPv6 (ipv6Addresses from IpEndpoint of NfService if present, or ipv6Addresses of NfProfile)
- FQDN (fqdn of NfService if present, or fqdn of NfProfile)

Managing Configurable Port and Routing Parameter Selection Feature

Configure

You can configure the following parameters using the REST API or CNC Console:

- **Configure using REST API**: Perform the `preferredPortFromIPEndpoint` and `preferredRoutingParameter` feature configurations as described in *Oracle*

Copyright © 2019, 2025, Oracle and/or its affiliates.

Communications Cloud Native Core, Network Repository Function REST Specification Guide.

- **Configure using CNC Console**: Perform the feature configurations as described in SLF Options.

4.25.4 Rerouting SLF Requests Using Alternate SCP and Alternate SLF

NRF supports routing SLF requests through SCP as per the configuration in the Egress Gateway SBI routing configuration. SLFs are selected based on Static or Dynamic SLF feature configuration.

In case an error response is received from SLF, NRF reroutes the SLF request to an alternate SLF. However, in this case, the same SCP that was used in the previous route is always used. It is possible that SCP had rerouted the SLF request to the alternate SLF in the first attempt and it failed. In the subsequent attempt if the same SCP is used, there is a possibility that this reroute might also fail. Also, NRF doesn't consider the number of SLFs that are already been attempted by SCP.

With the implementation of this feature, NRF allows you to configure maximum number of SLF attempts or an alternate SCP route to enhance the routing strategy and minimize the number of reroutes. When an error response is received, the subsequent reroutes to SLF are through alternate SCP and SLF path, improving resilience and service continuity.

NRF discovery microservice sends the `Nudr_GroupIDmap` request to SLF through SCP based on the Egress Gateway routes configuration.

This feature allows you to configure the maximum number of SLF attempts or an alternate SCP route to process the subsequent attempts.

This can be achieved by:

- **Limiting number of SLF attempts**: This is performed by configuring the number of SLF attempts in the `maxSLFattempts` parameter. For more information, see Maximum SLF Attempts.
- **Choosing an alternate SCP**: This is performed by enabling the `useAlternateScp` parameter. For more information, see Use Alternate SCP.

> **Note**
>
> If the Egress Gateway is down, then no reroute is performed and an error response is sent to the producer NF.

Maximum SLF Attempts

In a routing using SCP deployment, when the discovery service receives a non-2xx response, NRF extracts the server header to deduce the number of SLFs that have been attempted.

For example:

- If the server header is `Server: UDR-UDR1-Instance-Id UDR-UDR2-Instance-Id`, the number of attempts is considered as **2**.
- If the server header is `Server: UDR-UDR1-Instance-Id UDR-UDR2-Instance-Id SCP-SCP1-Instance-Id`, the number of attempts is considered as **2**.
- If the server header is `Server: envoy`, the number of attempts is considered as **1**.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- If the server header is `Server: UDR-UDR1-Instance-Id`, the number of attempts is considered as **1**.
- If the server header is `Server: SCP-SCP1-Instance-Id`, the number of attempts is considered as **1**.
- If there is no server header received or server header is empty, the number of attempts is considered as **1**.
- If the request has timed out, the number of attempts is considered as **1**.

The number of attempts to SLF is validated against the value configured for `maxSLFattempts` parameter, as follows:

- If the `maxSLFattempts` is not exhausted, discovery selects the next SLF to reroute the request.
- If the `maxSLFattempts` is exhausted and there is no successful response, discovery sends the error response. If there is only one SLF, NRF will not reroute to the already attempted SLF and send the error response.
- If the `maxSLFattempts` is exhausted and forwarding is enabled, NRF routes the discovery request to another NRF based on `maximumHopCount` configuration.
  The `maximumHopCount` provides the total number of SLF attempts and NRFs forwarding attempts performed. NRF tries to attempt to all available SLFs and then if `maximumHopCount` is left, NRF will forward the request to another NRF. When the `maxSLFattempts` parameter is configured, the operator can choose how many maximum SLF attempts can be done before forwarding the request.

For example:

In 3 SLF and 4 NRF deployments,

- If `maximumHopCount` is set to 5 and `maxSLFattempts` is set as 0, then the number of attempts to SLF is 3 and number of attempts to NRF is 2.
- If `maximumHopCount` is set to 5 and `maxSLFattempts` is set as 2, the following combinations are possible:
  - the number of attempts to SLF is 1 and number of attempts to NRF is 4.
  - the number of attempts to SLF is 2 and number of attempts to NRF is 3.

> **Note**
>
> The value of `maxSLFattempts` parameter must be less than or equal to the value configured for `maximumHopCount` parameter in `{apiRoot}/nrf-configuration/v1/generalOptions` API. For more information about the `maximumHopCount` parameter, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

For more information about server header propagation in service-mesh deployment, see Adding Filters for Server Header Propagation.

Use Alternate SCP

In a routing using SCP deployment, when the discovery service receives a non-2xx response, NRF checks the `useAlternateScp` parameter value.

If `useAlternateScp` parameter is set to **true**, then NRF chooses an alternate SCP, if available, to reroute the in a round-robin format to reroute the request. If there is only one SCP instance,

Copyright © 2019, 2025, Oracle and/or its affiliates.

NRF reroutes the request using the same SCP irrespective of the `useAlternateScp` configuration.

If all the available SCPs are tired and attempts are left, then NRF revolves back to the first SCP and repeat the SCP selection based on the round-robin.

> **Note**
>
> - When `useAlternateScp` is set to **true**, alternate SCP selection will not guaranteed if SCP Health APIs feature is enabled. Hence, it is recommended to disable SCP Health APIs feature when `useAlternateScp` is set to **true**. For more information about the SCP Health APIs feature, see the Monitoring the Availability of SCP Using SCP Health APIs section.
> - When `useAlternateScp` is set to **true**, `customPeerSelectorEnabled` parameter in `{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration` must also be set to **true**.

Managing the feature

This section explains the procedure to enable and configure the feature.

Configure

You can configure the feature either using REST API or Console:

- **Configure using REST API**: Perform the following feature configurations as described in the "Egress Gateway Configuration" section of *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*:
  - Configure the value of `maxSLFattempts` parameter greater than **0** to limit the number of SLF attempts. The number of attempts is calculated based on the server header received in the response.
  - (Optional) Enable `useAlternateScp` in the `{apiRoot}/nrf-configuration/v1/slfOptions` to allow alternate routing using SCP.
  - Create or update the alternate SCP peer details in the `{apiRoot}/nrf/nf-common-component/v1/egw/peerconfiguration`.
  - Create or update the `{apiRoot}/nrf/nf-common-component/v1/egw/peersetconfiguration` to assign these peers.
  - Update the `{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration` as mentioned in the below sample:

    > **Note**
    >
    > When `maxSLFattempts` is greater than **0** or `useAlternateScp` is set to **true**, don't configure `errorHandling` under `routesconfiguration`, as NRF discovery microservice will handle the reroute.

    Sample body:

```
[
  {
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
    "id": "egress_scp_proxy1",
    "uri": "http://localhost:32068/",
    "order": 3,
    "metadata": {
      "httpsTargetOnly": true,
      "httpRuriOnly": true,
      "sbiRoutingEnabled": true
    },
    "predicates": [
      {
        "args": {
          "pattern": "/nudr-group-id-map/v1/nf-group-ids"
        },
        "name": "Path"
      }
    ],
    "filters": [
      {
        "name": "SbiRouting",
        "args": {
          "peerSetIdentifier": "set0",
          "customPeerSelectorEnabled": true
        }
      }
    ]
  },
  {
    "id": "default_route",
    "uri": "egress://request.uri",
    "order": 100,
    "filters": [
      {
        "name": "DefaultRouteRetry"
      }
    ],
    "predicates": [
      {
        "args": {
          "pattern": "/**"
        },
        "name": "Path"
      }
    ]
  }
]
```

- **Configure using Console**: Perform the following feature configurations as described:
  - Configure the value of **Maximum SLF Attempts** parameter greater than **0** in the SLF Options to limit the number of SLF attempts.
  - (Optional) Enable **Use Alternate SCP** in the SLF Options to allow alternate routing using SCP.
  - Create or update Peer Configuration.
  - Create or update the Peer Set Configuration to assign these peers.
  - Update the Routes Configuration as mentioned above.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> When **Maximum SLF Attempts** is greater than **0** or **Use Alternate SCP** is set to **true**, don't configure **Error Handling** under **Routes Configuration**, as NRF discovery microservice will handle the reroute.

Observe

Metrics

- Updated the dimension list for the following metrics with `slfFqdn` dimension:
  - `ocnrf_SLF_tx_requests_total` in Table 6-154.
  - `ocnrf_SLF_rx_responses_total` in Table 6-155.
- Updated the description of the `ocnrf_max_slf_attempts_exhausted_total` metric in the Table 6-158.

Alerts

Updated the description of the OcnrfMaxSlfAttemptsExhausted alert.

KPIs

There are no KPIs for this feature.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information about how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information about how to raise a service request.

4.25.5 Discovery Parameter Value Based Skip SLF Lookup

Currently, NRF does not have an option to skip SLF lookup based on specific values of the discovery query parameter.

With the implementation of this feature, when NRF receives a discovery request, it provides an option to skip the Subscriber Location Function (SLF) lookup if a certain discovery query parameter, configured under `skipSLFLookupParameters` is present in the discovery request. In this case, the SLF lookup is skipped regardless of the parameter's value.

This feature provides an option to configure a set of allowed values for the discovery query parameter. If the parameter is present in the discovery request and its value matches the configured values, the SLF lookup will be skipped.

NRF processes the discovery request and performs skip SLF lookup as follows:

- If the value of `enableValueBasedSkipSLFLookup` parameter is set as *true*, NRF validates the discovery query parameters and its values received in the discovery request against the values configured in `valueBasedSkipSLFLookupParams` parameter to determine if the

Copyright © 2019, 2025, Oracle and/or its affiliates.

SLF lookup is to be skipped. The `valueBasedSkipSLFLookupParams` parameter comprises of `parameterName` and `parameterValue` attributes.
`valueBasedSkipSLFLookupParams": [ {"parameterName": "dnn","parameterValue": "abc*"},{"parameterName": "group-id-list" ,"parameterValue": ".*"}]`

Where,

– `parameterName` - indicates a valid discovery query parameter supported by NRF for which the SLF lookup is to be skipped.

> **Note**
>
> In 25.1.100, this feature supports only dnn based skipSlfLookup.

– `parameterValue` - indicates the value for the corresponding discovery parameter name for which the SLF lookup is skipped. This value can be configured as a regular expression.

  * If the `parameterValue` is configured as .* , it indicates all values. NRF skips the SLF lookup for any value received for the discovery parameter.
  * If the `parameterValue` is a regular expression, NRF matches this value with the discovery request.
    * If the values match, NRF skips the SLF lookup for a value that matches with the expression and proceeds with discovery processing.
    * If the expression does not match the value, NRF performs SLF lookup. NRF will not fallback to the skip SLF lookup feature, even if (`skipSLFLookupParameters`) is configured.

> **Note**
>
> Pattern matching is performed using Java Platform Standard Ed. 8 Documentation. For more information, about regular expressions, see https://docs.oracle.com/javase/8/docs/api/java/util/regex/Pattern.html.

- If the value of `enableValueBasedSkipSLFLookup` is set as *false*, NRF falls back to the existing skip SLF lookup feature based on the list of discovery parameters configured in the `skipSLFLookupParameters` parameter.

The following table describes the functionality of the feature based on the sample values configured in `valueBasedSkipSLFLookupParams` and the incoming discovery query parameters.

Table 4-19 Examples for Discovery Parameter Value Based Skip SLF Lookup

Value of enableValueBasedSkipSLFLookup parameter	parameterName and parameterValue configured in valueBasedSkipSLFLookupParams	Query Parameter name and value received in the incoming discovery requests	Result
true	{"parameterName":"dnn","parameterValue":".*"}	dnn=dnn-c4, dnn=dnn-1, dnn=DnNvAl123	SLF lookup is skipped for all the values (dnn-c4, dnn-1 , and DnNvAl123).

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-19 (Cont.) Examples for Discovery Parameter Value Based Skip SLF Lookup

Value of enableValueBasedSkipSLFLookup parameter	parameterName and parameterValue configured in valueBasedSkipSLFLookupParams	Query Parameter name and value received in the incoming discovery requests	Result
true	{"parameterName":"dnn","parameterValue":"dnn*"}	dnn=dnn-c4, dnn=dnn-1, dnn=DnvAl123, dnn=DnaaAl123	SLF lookup is skipped for the following incoming requests with the values dnn-c4 and dnn-1 and but SLF lookup will be performed for the incoming requests with the values DnvAl123 and DnaaAl123.
true	{"parameterName":"dnn","parameterValue":"^dnn-c4$"}	dnn=dnn-c4, dnn=adnn-c4, dnn=dnn-1, dnn=dnnAl123, dnn=DnaaAl123	SLF lookup is skipped for the incoming request with value dnn-c4 and SLF lookup is performed for the incoming requests with the values adnn-c4, dnn-1, dnnAl123, and DnaaAl123.
true	{"parameterName":"dnn","parameterValue":"^prov[a-g\d]{1,3}\w{3}$"}	dnn=prova5aaa, dnn=provaa15aaa, dnn=prova5a	SLF lookup is skipped for the incoming request with value prova5aaa and SLF lookup is performed for the incoming requests with the values provaa15aaa and prova5a.
true	{"parameterName":"dnn","parameterValue":"^province1\.mnc\d{3}\.mcc\d{3}\.gprs$"}	dnn=province1.mnc014.mcc310.gprs, dnn=province3.mnc012.mcc3150.gprs, dnn=province1.mnc0141.mcc001.gprs, dnn=province1.mnc001.mcc003.gprs	SLF lookup is skipped for the incoming request with value province1.mnc014.mcc310.gprs, province1.mnc001.mcc003.gprs and SLF lookup is performed for the incoming requests with the values province3.mnc012.mcc3150.gprs, andprovince1.mnc0141.mcc001.gprs
true	{"parameterName":"dnn","parameterValue":"parameterValue":".+\.mnc012\.mcc345\.gprs"}	prov.mnc012.mcc345.gprs, prov2.mnc011.mcc234.gprs,prov2.mnc001.mcc233.gprs	SLF lookup is skipped for the incoming request with value prov.mnc012.mcc345.gprs and SLF lookup is performed for the incoming requests with the values prov2.mnc011.mcc234.gprs and prov2.mnc001.mcc233.gprs.
false	{"parameterName":"dnn","parameterValue":"dnn*"}	dnn=dnn-c4, dnn=dnn-1, dnn=DnvAl123, dnn=DnaaAl123	Value based SLF lookup is not performed as the feature is disabled. falls back to the existing skip SLF lookup feature (skipSLFLookupParameters), if it is configured.

Managing Discovery Parameter Value Based Skip SLF Lookup

Configure

You can configure the following parameters using the REST API or CNC Console:

Copyright © 2019, 2025, Oracle and/or its affiliates.

- **Configure using REST API**: Configure the values for `valueBasedSkipSLFLookupParams` as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure using CNC Console**: Configure the values for **Value Based Skip SLF Lookup Parameter List** as described in SLF Options.

> **Note**
>
> After NRF upgrade to 25.1.100, the `valueBasedSkipSLFLookupParams` is empty by default. Populate the values for `valueBasedSkipSLFLookupParams` attribute and then enable the `enableValueBasedSkipSLFLookup` parameter. The feature cannot be enabled when the `valueBasedSkipSLFLookupParams` is empty.

Enable

You can enable Discovery Parameter Value Based Skip SLF Lookup feature using the REST API or CNC Console.

- Enable using REST API: Set `enableValueBasedSkipSLFLookup` to true in SLF Options configuration API. For more information about API path, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide.*
- Enable using CNC Console: Set **Enable Value Based Skip SLF Lookup** to **true** on the SLF Options page. For more information about enabling the feature using CNC Console, see SLF Options.

Observability

Metrics

The existing `ocnrf_nfDiscover_SLFlookup_skipped_total` metric is enhanced with a new dimension `SkipSLFLookupValue` for this feature.

For more information about SLF metrics, see NRF SLF Metrics section.

Alerts

There are no alerts added or updated for this feature.

KPIs

There are no KPIs related to this feature.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts persist, perform the following:

1. **Collect the logs:** For more information on how to collect logs, see "Collecting Logs" section in *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.

2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

Copyright © 2019, 2025, Oracle and/or its affiliates.

4.26 NRF Forwarding

NRF Forwarding feature forwards the service operation messages to another NRF, if NRF is not able to fulfill the required service operation.

> **Note**
>
> Service operations explained below with specific cases or scenarios are eligible for forwarding.

A consumer NF instance can perform the following:

- Subscribe to changes of NF instances registered in an NRF to which it is not directly interacting. The NF subscription message is forwarded by an intermediate NRF to another NRF.
- Retrieve the NF profile of the NF instances registered in an NRF to which it is not directly interacting. The NF profile retrieval message is forwarded by an intermediate NRF to another NRF.
- Discover the NF profile of the NF instances registered in an NRF to which it is not directly interacting. The NF discover message is forwarded by an intermediate NRF to another NRF.
- Request OAuth 2.0 access token for the NF instances registered in an NRF to which it is not directly interacting. The OAuth 2.0 access token service request is forwarded by an intermediate NRF to NRF (which may issue the token).

NRF also enables the users to define the forwarding criteria for NFDiscover and AccessToken Service Requests from NRF to an intermediate NRF. The user can configure the preferred NF Type, the NF Services, or both for which forwarding is applicable. This provides the flexibility to regulate the traffic between the NRFs. For enabling Forwarding Rules Feature configuration, see the Enable Forwarding Rules section.

Managing NRF Forwarding Feature

Enable
The Forwarding feature is a core functionality of NRF. You need not enable or disable this feature.

If you are enabling NRF Growth feature, then perform the configuration as described in Configure Forwarding Options for NRF Growth.

Enable Forwarding Rules

You can enable the Forwarding Rules feature using REST API or CNC Console.

- **Enable using REST API**: Set `featureStatus` to `ENABLED` in **Forwarding Options** configuration API. For more information about API path, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Set **Feature Status** to **ENABLED** on the Forwarding Options page. For more information about enabling the feature using CNC Console, see Forwarding Options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> - Before enabling `forwardingRulesFeatureConfig`, ensure that corresponding configurations are completed.
> - The `featureStatus` flag under `forwardingRulesFeatureConfig` can be **ENABLED**, only if `discoveryStatus` or `accessTokenStatus` attributes are already **ENABLED**.
> - Once `featureStatus` flag under `forwardingRulesFeatureConfig` is **ENABLED**, both `discoveryStatus` and `accessTokenStatus` attributes cannot be **DISABLED**.

Configure

Configure the NRF Forwarding feature using REST API or CNC Console:

- **Configure NRF Forwarding feature using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure NRF Forwarding feature using CNC Console**: Perform the feature configurations as described in Forwarding Options.

> **Note**
>
> The forwarding rules for nfDiscover service requests are based on following 3GPP discovery request parameters:
>
> - target-nf-type ***(Mandatory Parameter)***
> - service-names ***(Optional Parameter)***
>
> The forwarding rules for access token service requests are based on the following 3GPP access token request parameter:
>
> - scope ***(Mandatory Parameter)***

Configure Forwarding Options for NRF Growth

Configure the NRF Forwarding Options for NRF Growth using REST API or CNC Console:

- **Configure NRF Forwarding Options for NRF Growth feature using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure NRF Forwarding Options for NRF Growth feature using CNC Console**: Perform the feature configurations as described in Forwarding Options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> The forwarding rules for nfDiscover service requests are based on following 3GPP discovery request parameters:
>
> - target-nf-type ***(Mandatory Parameter)***
> - service-names ***(Optional Parameter)***
>
> The forwarding rules for access token service requests are based on the following 3GPP access token request parameter:
>
> - scope ***(Mandatory Parameter)***

NRF Host configuration attribute (`nrfHostConfig`) allows the user to configure the details of another NRF to which the service operation messages are forwarded.

> **Note**
>
> For NRF-NRF Forwarding feature to work, `nrfHostConfig` attribute must be configured in both NRFs with each other values.
> For example: NRF1 is forwarding requests towards NRF2, then `nrfHostConfig` configuration attribute of NRF1 shall have NRF2 details and similarly `nrfHostConfig` configuration attribute of NRF2 shall have NRF1 details. These configurations are used while handling different service operations of NRF.

The `nrfHostConfig` configuration consists of attributes like apiVersion, scheme, FQDN, port, priority, and so on. NRF allows to configure maximum of four host details. However, the host with the highest priority is considered as the Primary host.

> **Note**
>
> - Refer 3GPP TS 29.510 (release 15.5) for definition and allowed range for NfHostConfig attributes (apiVersion, scheme, FQDN, port, priority, and so on).
> - Apart from priority attribute, no other attributes plays any role in the host selection.
> - Apart from Primary, three more hosts (if configured) can used as alternate NRF.
> - When more than one host is configured with same priority, the order of NRF host selection between the same priority is random.

In the NRF Forwarding feature, a request is first forwarded to Primary NRF. In case of error, the request is forwarded to alternate NRFs based on the following configurations:

- `nrfRerouteOnResponseHttpStatusCodes`: This configuration is used to determine if the service operation message can be forwarded to alternate NRF or not. After receiving a response from the primary NRF, if the response status code from the primary NRF matches with this configuration, then NRF reroutes the request to the alternate NRF. Refer `nrfHostConfig` attribute for host NRF details.
  Prior to 22.2.x of NRF, Egress Gateway used to return 503 response for timeout exceptions (for example, No response received for SLF query or No Response received for Query sent to Forwarding NRF).

Copyright © 2019, 2025, Oracle and/or its affiliates.

From 22.3.x onwards, Egress Gateway returns 408 response codes for timeout exceptions (REQUEST_TIMEOUT and CONNECTION_TIMEOUT), to keep the behavior in line with the HTTP standard. Hence, after upgrading to 22.3.x or a fresh install of 22.3.x, if a retry needs to be performed for the timeout scenario, 408 error code explicitly needs to be configured under forwarding reroute option. Following is the configuration for forwarding options:

```
{
          "nrfRerouteOnResponseHttpStatusCodes": {
            "pattern": "^[3,5][0-9]{2}$|408$"
          }
}
```

- `maximumHopCount`: This configuration determines maximum number of hops (SLF or NRF) that NRF can forward in a given service request. This configuration useful during NRF Forwarding and SLF feature interaction.

Observe

Following are the NRF Forwarding feature specific metrics filters:

- ocnrf_forward_accessToken_tx_requests_total
- ocnrf_forward_accessToken_rx_responses_total
- ocnrf_forward_nfProfileRetrieval_tx_requests_total
- ocnrf_forward_nfProfileRetrieval_rx_responses_total
- ocnrf_forward_nfStatusSubscribe_tx_requests_total
- ocnrf_forward_nfStatusSubscribe_rx_responses_total
- ocnrf_forward_nfDiscover_tx_requests_total
- ocnrf_forward_nfDiscover_rx_responses_total
- ocnrf_forwarding_jetty_latency_seconds
- ocnrf_forward_nfDiscover_barred_total
- ocnrf_forward_accessToken_barred_total
- ocnrf_forward_nfStatusSubscribe_barred_total
- ocnrf_forward_profileRetrieval_barred_total

For more information on NRF Forwarding feature metrics and KPIs, see NRF Forwarding Metrics, and NRF KPIs sections.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persist, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see "Collecting Logs" section in *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

Copyright © 2019, 2025, Oracle and/or its affiliates.

4.27 NRF Georedundancy

NRF supports georedundancy to ensure high availability and redundancy. It offers a two, three, or four-sites georedundancy to ensure service availability when one of the NRF site is down. When NRF is deployed as georedundant site, all the sites work in an active state and the same data is available at all the sites.

The NFs send service requests to its primary NRF. When the primary NRF site is unavailable, the NFs redirect the service requests to the alternate site NRF. In this case, NFs get the same information from alternate NRF, and each site maintains and uses its set of configurations.

The NRF's state data gets replicated between the georedundant sites using DBTier's replication service.

Following are the prerequisites for georedundancy:

- Each site configures the remote NRF sites that it is georedundant with.
- Once the georedundancy feature is enabled on a site, it cannot be disabled.
- Georedundant sites must be time synchronized.
- Georedundant sites must be reachable from NFs or Peers on all the sites.
- NFs are required to configure primary and alternate NRFs so that when one site is down, the alternate NRF can provide the required service operations.
- At any given instance, NFs communicate with only one NRF, that is, NFs register services and maintain heartbeats with only one NRF. The data is replicated across the georedundant NRFs, thereby allowing the NFs to seamlessly move between the NRFs in case of failure.
- Ensure that site is isolated before performing georeplication.

Managing NRF Georedundancy Feature

Prerequisites

1. cnDBTier is installed and configured for each site. The DB replication channels between the sites are up. For more information about cnDBTier installation, see
2. Configure MySQL Database, Users, and Secrets. For the configuration procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Preconfigurations for Georedundancy Feature

The following are the preconfigurations that must be performed before enabling georedundancy feature.

- **Preconfigurations using Helm**:
  1. Before performing upgrade, update the Database Monitor service host and port in the below attributes as per the deployment in `appInfo` parameter section.

```
The attribute indicates to monitor the database status
    watchMySQL: true
  #The URI used by the appinfo service to retrieve the replication
channel status from the CN DB Tier. The service name used is the db-
monitor service.The URI should be provided as "http://<db monitor
service name>:<db monitor service port>/db-tier/status/replication/
realtime"
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
  replicationUri: "http://mysql-cluster-db-monitor-svc.occne-
infra:8080/db-tier/status/replication/realtime"
  #The URI used by the appinfo service to retrieve the database status
from the CN DB Tier. This is for future usage. The service name used is
the db-monitor service. It is recommended to configure these URI
correctly to avoid continuous error log in App-info.The URI should be
provided as "http://<db monitor service name>:<db monitor service
port>/db-tier/status/local"
  dbStatusUri: "http://mysql-cluster-db-monitor-svc.occne-infra:8080/db-
tier/status/local"
  #The URI used by the appinfo service to retrieve the realtime
database status from the CN DB Tier. This is for future usage.It is
recommended to configure these URI correctly to avoid continuous error
log in App-info.The service name used is the db-monitor service.The URI
should be provided as "http://<db monitor service name>:<db monitor
service port>/db-tier/status/local"
  realtimeDbStatusUri: "http://mysql-cluster-db-monitor-svc.occne-
infra:8080/db-tier/status/cluster/local/realtime"
  #The URI used by the appinfo service to retrieve the dbTier version
from the CN DB Tier. [This url is supported from CN DBTier 22.4.0].The
service name used is the db-monitor service.The URI should be provided
as "http://<db monitor service name>:<db monitor service port>/db-tier/
version"
  dbTierVersionUri: "http://mysql-cluster-db-monitor-svc.occne-
infra:8080/db-tier/version"
  #The URI used by the appinfo service to retrieve alerts from alert
manager. The service name used is the alert manager service name of the
CNE.The URI should be provided as "http://<alert manager service
name>:<alert manager service port>/<cluster name>/alertmanager".
  alertmanagerUrl: "http://occne-prom-alertmanager.occne-infra:80/
cluster/alertmanager"
```

2. Run `helm upgrade` to apply the above configuration.
3. Verify if the appinfo service is able to fetch the database replication status by querying the following API:

   ```
   Resource Method : GET
           Resource URI: <appinfo-svc>:<appinfo-port>/status/category/
   replicationstatus
   ```

   Where,

   `<appinfo-svc>` is the appinfo service name.

   `<appinfo-port>` is the port of appinfo service.

   Sample response:

   ```
   [
     {
       "localSiteName": "nrf1",
       "remoteSiteName": "nrf2",
       "replicationStatus": "UP",
       "secondsBehindRemote": 0,
       "replicationGroupDelay": [
         {
   ```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
        "replchannel_group_id": "1",
        "secondsBehindRemote": 0
      }
    ]
  }
]
```

Recheck the configuration if the sample response is not received. If the sample response is not received, appinfo is not querying the Database Monitor service correctly.

4. This step must be done only after verifying that appinfo is able to fetch the status correctly. If the above configuration is not correct, NRF services will receive incorrect information about the database replication channel status and hence will assume replication channel status as DOWN.
   Configure the NRF microservice to query appinfo for the replication status using the **Georedundancy Options** API or CNC Console:

   ```
   Resource Method : PUT
   Resource URI: <configuration-svc>:<configuration-port>/nrf-
   configuration/v1/geoRedundancyOptions
   {
      “replicationStatusUri”: "http://<appinfo-svc>:<appinfo-port>/status/
   category/replicationstatus"
   }
   ```

   Where,

   `<configuration-svc>` is the nrfconfiguration service name.

   `<configuration-port>` is the port of nrfconfiguration service.

   `<appinfo-svc>` is the appinfo service name.

   `<appinfo-port>` is the port of appinfo service.

   Perform the configurations in CNC Console as described in Georedundancy Options.

5. Verify that the `OcnrfDbReplicationStatusInactive` and `OcnrfReplicationStatusMonitoringInactive` alerts are not raised on the alert dashboard.

- **Preconfigurations using REST API**:

  1. Deploy NRF as per the installation procedure provided in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
  2. Enable the Georedundancy feature using the CNC Console or REST API:
     Following is the sample configuration at Site Chicago which is georedundant with Sites Atlantic (siteName: Atlantic, NrfInstanceId: 723da493-528f-4bed-871a-2376295c0020) and Pacific (siteName: Pacific, NrfInstanceId: cfa780dc-c8ed-11eb-b8bc-0242ac130003)

     ```
     # Resource URI:  {apiRoot}/nrf-configuration/v1/geoRedundancyOptions
     # Method: GET
     # Content Type: application/json
     {
     "useRemoteDataWhenReplDown":false,
     "featureStatus":"ENABLED",
     ```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
"monitorNrfServiceStatusInterval":"5s",
"monitorDBReplicationStatusInterval":"5s",
"replicationDownTimeTolerance":"10s",
"replicationUpTimeTolerance":"10s",
"replicationLatencyThreshold":"20s",
"replicationStatusUri": "http://<appinfo-svc>:<appinfo-port>/status/
category/replicationstatus",
"siteNameToNrfInstanceIdMappingList":
    [{"siteName":"atlantic",
      "nrfInstanceId":"723da493-528f-4bed-871a-2376295c0020"},
     {"siteName":"pacific",
      "nrfInstanceId":"cfa780dc-c8ed-11eb-b8bc-0242ac130003"}
    ]}
}
```

NRF considers NfProfiles, registered across georedundant sites, for all its service operations. However, if NRF detects that the replication status with its mated sites is down, it will consider NfProfiles registered at local site only. If NfProfiles that are registered across georedundant sites must be considered for certain service operations (for example, service operations that result into DB Update), set the `useRemoteDataWhenReplDown` attribute to **true**.

This attribute is applicable only for the NfHeartBeat, NfUpdate (Patch), NfDeregister, NfStatusSubscribe (Patch), and NfStatusUnSubscribe service operations. It ensures that during the NRF site down scenario, if the NF is moving to its mated NRFs, NF need not reregister or resubscribe again. However, if this attribute is set to false and NF is switching to mated site, then the NF receives a 404 response and:

- for NfUpdate (Patch), NfDeregister, and NfHeartBeat opearations, the NF is expected to register again with the NRF using NfRegister service operations.
- for NfStatusSubscribe (Patch) and NfStatusUnSubscribe, the NF is expected to subscribe again with the NRF using NfStatusSubscribe service operation.

> **Note**
>
> This attribute is not applicable for NfDiscovery, NfAccessToken, NfProfileRetrieval, and NfListRetrieval service operations.

- **Preconfigurations using CNC Console**:
  1. Configure **Replication Status Uri** to `"http://<appinfo-svc>:<appinfo-port>/status/category/replicationstatus"`
  2. Configure **Site Name To NRF InstanceId Mapping List** group with the NRF Instance Id and the corresponding database Site Name of the remote site(s) with which the given NRF is georedundant.

> **Note**
>
> Configure these mandatory attributes before enabling the Georedundancy feature. If these attributes are not configured during deployment or using the CNC Console postdeployment, georedundancy cannot be enabled, and NRF at the site acts as a standalone NRF.

Enable

Copyright © 2019, 2025, Oracle and/or its affiliates.

After performing the above mentioned preconfigurations, enable the NRF georedundancy feature using the REST API or CNC Console.

- **Enable using REST API**: Set `featureStatus` to `ENABLED` in **Georedundancy Options** configuration API. For more information about API path, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Set **Feature Status** to **ENABLED** on the Georedundancy Options page. For more information about enabling the feature using CNC Console, see Georedundancy Options.

Configure

You can configure the georedundancy feature using REST API or CNC Console:

- **Configure Georedundancy using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure Georedundancy using CNC Console**: Perform the feature configurations as described in Georedundancy Options.

Fetching the Database Replication Channel Status Using appinfo Microservice

NRF microservices require the database replication channel status for various NRF service operations. The Database Monitor service exposes APIs which provides the database replication channel status. The following NRF microservice query this API:

- Nfregistration
- Nfsubscription
- NfDiscover
- nrfAuditor
- NfAccessToken
- NrfConfiguration
- NrfArtisan

Currently, all pods of the above services perform the REST API query towards the Database Monitor service periodically, and maintains the status in memory. When NRF is scaled to handle high traffic rate, the number of pods querying the Database Monitor service is very high. But, the Database Monitor service is not designed to handle high traffic rate.

Observe

Following are the georedundancy feature specific metrics:

- ocnrf_dbreplication_status
- ocnrf_dbreplication_down_time_seconds
- ocnrf_nf_switch_over_total
- ocnrf_nfSubscriptions_switch_over_total
- ocnrf_stale_nf_deleted_total
- ocnrf_stale_nfSubscriptions_deleted_total
- ocnrf_reported_dbreplication_status

For more information on georedundancy metrics and KPIs, see Georedundancy Metrics and NRF KPIs sections.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.27.1 NRF Last Known Response

When an NRF detects that the replication channel with its georedundant NRF(s) is down, it stops considering the NFs registered in the georedundant NRF(s) for any service operations. Due to unavailability of the replication channel, the latest status of the NFs registered in the georedundant NRF(s) will not get replicated. In this state the last replicated NfProfiles may be stale, hence NRF does not consider these profiles for any service operations.

During platform maintenance, network issues, or for any other reason, if the replication channels goes down, given that usually, the NFs do not register, deregister or change their profiles very often, it is acceptable for NRF to use the last known NfProfiles of remote NRFs for any service operations.

As part of this feature, `overrideReplicationCheck` Helm configurable parameter is provided to control the behavior when replication channels are down. The parameter is applicable for the following NRF microservices only:

- nfregistration
- nfaccesstoken
- nfdiscovery
- nrfartisan
- nrfcachedata

> **Note**
>
> `overrideReplicationCheck` is a global parameter. This is a reference variable. The value of this parameter is applicable for the above-mentioned microservices.

Enable

You can enable Last Known Response using the Helm.

> **Note**
>
> Perform the following steps only after Geo Redundancy Feature is enabled as described in Georedundancy Options.

1. Open the `ocnrf_custom_values_25.1.201.yaml` file.

Copyright © 2019, 2025, Oracle and/or its affiliates.

2. Set `global.overrideReplicationCheck` to **true** to enable the Last Known Response feature.

   > **Note**
   >
   > `global.overrideReplicationCheck` is a reference variable.

   For more information about the parameters, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
3. Save the file.
4. Run `helm upgrade`. For more information about upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

NRF behavior When the Feature is Disabled

Following are the different scenarios that explain the behavior of each NRF microservices when this feature is disabled:

- **Replication is Up**
  - **NF Register Microservice**:
    - **NfRegister, NfUpdate, and NfDeregister Processing**: The service operations will be successful, irrespective of which site the NF was originally registered at the local site.
    - **NfListRetrieval and NfProfileRetrieval Processing**:

      The NRF will consider:

      - The NFs registered and heartbeating at the local site.
      - The NFs registered and heartbeating at the remote sites, which replicated from its Georedundant mate NRFs.
  - **NF Discover and NF AccessToken Microservice**:

    The NRF will consider:

    - The NFs registered and heartbeating at the local site.
    - The NFs registered and heartbeating at the remote sites, which replicated from its Georedundant mate NRFs.
  - **NrfArtisan Microservice**: The SLFs registered at both local site and remote sites are used for discovering UDRs for performing SLF query, when `slfConfigMode` is set to DISCOVERED_SLF_CONFIG_MODE.
  - **NrfAuditor Microservice**:
    - NRF will audit the NfProfiles and subscriptions of the local site.

      When NRF discovers any of its registered profiles has missed heartbeats, it will be marked as SUSPENDED. The NF will thereafter not be included in the nfDiscovery or nfAccessToken response. The Consumer NFs that have subscribed for the change are notified about the change.
    - When NRF discovers any of its subscriptions have crossed the validity period, it deletes the subscription.

Copyright © 2019, 2025, Oracle and/or its affiliates.

* NRF will also audit the remote site NRFs NfProfiles and Subscriptions. If NRF discovers any of the registered Profiles at the remote site has missed heartbeats, it will be marked as PROBABLE_SUSPENDED (internal state). The NF will thereafter not be included in the nfDiscovery or nfAccessToken response. The NF will eventually be marked as SUSPENDED, if the NF continues to miss its heartbeat for a configurable period. The Consumer NFs that have subscribed for the change are not notified. When NRF receives a profile Retrieval request for a NfProfile which is in PROBABLE_SUSPENDED state, it shall return the profile in the response with the status set to UNDISCOVERABLE.
* When NRF discovers any of the subscriptions at the remote site have crossed the validity period, it marks the subscription as SUSPENDED.

> **Note**
>
> `global.overrideReplicationCheck` attribute is not configurable for NrfAuditor microservice. Refer the behavior for nrfAuditor when replication channel status is UP or DOWN.

- **Replication is Down**
  - **NF Register Microservice**:
    * **NfRegister, NfUpdate, and NfDeregister Processing**:
      * The NfProfiles will be processed if the NF is registered at the local site
      * If the NF is registered at the remote site, then the NfProfile will be processed only if `geoRedundancyOptions.useRemoteDataWhenReplDown` is **true**. Else, the request will be rejected with 404 Not Found.
    * **NfListRetrieval and NfProfileRetrieval Processing**:
      * The NRF will consider the NFs registered and heartbeating with the local site.
      * The NRF will not consider the NFs that are registered with remote NRF site with which the replication is down.
  - **NF Discover and NF AccessToken Microservice**:

    The NRF will consider:
    * The NFs registered and heartbeating at the local site.
    * NRF will not consider the NFs registered and heartbeating at the remote sites with which the replication is down.
  - **NrfArtisan Microservice**:
    * When slfConfigMode is set to DISCOVERED_SLF_CONFIG_MODE, NRF will consider
      * The SLFs registered and heartbeating at the local site.
      * The SLFs registered with remote site NRFs with which the replication is down will not be considered.
  - **NrfAuditor Microservice**:
    * NRF will audit the NfProfiles and subscriptions of the local site.
      When NRF discovers any of its registered profiles has missed heartbeats, it will be marked as PROBABLE_SUSPENDED (internal state). The NF will thereafter not be included in the nfDiscovery or nfAccessToken response. The Consumer NFs that have subscribed for the change shall not be notified of the change.

Copyright © 2019, 2025, Oracle and/or its affiliates.

When NRF discovers any of its subscriptions have crossed the validity period, it shall mark the Subscription as SUSPENDED (internal state).

* NRF will not audit the remote site NRFs NfProfiles and Subscriptions.

NRF behaviour When the Feature is Enabled, replication is UP or Down

- **NF Register Microservice**:
  - **NfRegister, NfUpdate, and NfDeregister Processing**:
    * The profiles will be processed for the NFs registered at the local site or at the remote sites, irrespective of the value of the `geoRedundancyOptions.useRemoteDataWhenReplDown` parameter.
  - **NfListRetrieval and NfProfileRetrieval Processing**:

    The NRF will consider:
    * The NFs registered and heartbeating at the local site.
    * The NFs registered and heartbeating at the remote sites (last known status) which replicated from its Georedundant mate NRFs.
- **NF Discover and NF AccessToken Microservice**:

  The NRF will consider:
  - The NFs registered and heartbeating at the local site.
  - The NFs registered and heartbeating at the remote sites (last known status) which replicated from its Georedundant mate NRFs.
- **NrfArtisan Microservice:**
  The SLFs registered at both local site and remote sites (last known status) are used for discovering UDRs for performing SLF query, when `slfConfigMode` is set to DISCOVERED_SLF_CONFIG_MODE.
- **NrfAuditor Microservice:**
  - If the replication channel status is up,
    * NRF will audit the NfProfiles and subscriptions of the local site.
      When NRF discovers any of its registered profiles has missed heartbeats, it will be marked as SUSPENDED. The NF will thereafter not be included in the nfDiscovery or nfAccessToken response. The Consumer NFs that have subscribed for the change are notified about the change.

      When NRF discovers any of its subscriptions have crossed the validity period, it deletes the subscription.
    * NRF will also audit the remote site NRFs NfProfiles and Subscriptions. If NRF discovers any of the registered Profiles at the remote site has missed heartbeats, it will be marked as PROBABLE_SUSPENDED (internal state). The NF will thereafter not be included in the nfDiscovery or nfAccessToken response. The Consumer NFs that have subscribed for the change are not notified.
    * When NRF discovers any of the subscriptions at the remote site have crossed the validity period, it marks the subscription as SUSPENDED.
  - If the replication channel status is down,
    * NRF will audit the NfProfiles and subscriptions of the local site.
      When NRF discovers any of its registered profiles has missed heartbeats, it will be marked as PROBABLE_SUSPENDED (internal state). The NF will thereafter not

Copyright © 2019, 2025, Oracle and/or its affiliates.

be included in the nfDiscovery or nfAccessToken response. The Consumer NFs that have subscribed for the change shall not be notified of the change.

* When NRF discovers any of its subscriptions have crossed the validity period, it shall mark the Subscription as SUSPENDED (internal state).
* NRF will not audit the remote site NRFs NfProfiles and Subscriptions.

> **Note**
>
> `global.overrideReplicationCheck` attribute is not configurable for NrfAuditor microservice. Refer the behavior for nrfAuditor when replication channel status is UP or DOWN.

4.28 NF Heartbeat Enhancement

This feature allows the operator to configure minimum, maximum, default heartbeat timers and the maximum number of consecutive heartbeats that the NF is allowed to skip. Further, these values can be customized per NF type.

According to 3GPP 29.510, every NF registered with NRF keeps its operative status alive by sending NF heartbeat requests periodically. The NF can optionally send the heartbeatTimer value when it registers its NFProfile or when it wants to update its registered NFProfile.

NRF may modify the value of the heartbeatTimer based on its configuration and return the new value to the NF on successful registration. The NF will thereafter use the heartbeatTimer as received in the registration response as its heartbeat interval.

In case the configuration changes at the NRF for the heartbeatTimer, the changed value must be communicated to the NF in the response of the next periodic NF heartbeat request or when it next sends an NFUpdate request to the NRF.

NRF monitors the operative status of all the NFs registered with it, and when it detects that an NF has missed updating its NFProfile or sending a heartbeat for the heartbeat interval, NRF must mark the NFProfile as SUSPENDED. The NFProfile and its services may no longer be discoverable by the other NFs through the NfDiscovery service. The NRF notifies the subscribed NFs of the change in status of the NFProfile.

Managing NF Heartbeat Enhancement

Enable
The NF heartbeat is a core functionality of NRF. You do not need to enable or disable this feature.

Configure

Configure the NF Heartbeat Enhancement using REST API or CNC Console:

- **Configure NF Heartbeat using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure NF Heartbeat using CNC Console**: Perform the feature configurations as described in NF Management Options.

Observe

For more information on heartbeat metrics and KPIs, see NRF Metrics and NRF KPIs sections.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.29 Service mesh for intra-NF communication

Oracle NRF leverages the Istio or Envoy service mesh (Aspen Service Mesh) for all internal and external communication. The service mesh integration provides inter-NF communication and allows API gateway co-working with service mesh. The service mesh integration supports the services by deploying a special sidecar proxy in the environment to intercept all network communication between microservices. For more information on configuring ASM, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

4.30 NF Screening

The incoming service requests from 5G Network Functions (NFs) must be screening before allowing access to Nnrf_NfManagement service operations to ensure security.

NF Screening feature screens the incoming service requests based on certain attributes in the NfProfile against a set of screening rules configured at NRF. NRF processes the incoming service requests, only if the screening is successful, and allows to invoke Nnrf_NfManagement service operations.

NRF supports the following NF screening rules list type. For more information about the screening rules applicable to different attributes in NfProfile, see Table 4-22.

Table 4-20 Screening Rules List Type

Screening Rule List	Description
"NF_FQDN"	Screening List type for NF FQDN. This screening rule type is applicable for `fqdn` of a NfProfile in NF_Register and NF_Update service operations.
"NF_IP_ENDPOINT"	Screening list type for IP Endpoint. This screening rule type is applicable for `ipv4address`, `ipv6address` attributes at NfProfile level and `ipEndPoint` attribute at nfServices level for NF_Register and NF_Update service operations.
"CALLBACK_URI"	Screening list type for callback URIs in NF Service and `nfStatusNotificationUri` in SubscriptionData. This is also applicable for `nfStatusNotificationUri` attribute of SubscriptionData for NFStatusSubscribe service operation. This screening rule type is applicable for `defaultNotificationSubscription` attribute at nfServices level for NF_Register and NF_Update service operations.
"PLMN_ID"	Screening list type for PLMN ID. This screening rule type is applicable for `plmnList` attribute at NfProfile level for NF_Register and NF_Update service operations.
"NF_TYPE_REGISTER"	Screening list type for allowed NF Types to register. NRF supports 3GPP TS 29510 Release 15 and specific Release 16 NF Types. For more information on the supported NF Types list, see "Supported NF Types" section. This screening rule type is applicable for `nfTypeList` attribute at NfProfile level for NF_Register and NF_Update service operations.

Copyright © 2019, 2025, Oracle and/or its affiliates.

When a service request is received, NRF performs screening of the service request for each of the above mentioned screening rules list type, as follows:

- Checks if the global screening option is Enabled or Disabled.
  You can configure the **nfScreeningOptions.featureStatus** parameter as "ENABLED" using REST or Console.

  > **Note**
  >
  > By default, the NF Screening feature is disabled globally.

- If it is Enabled, NRF checks if the attribute in the NfProfile is configured under BLOCKLIST or ALLOWLIST. For more information about these attributes, see Table 4-21.
  You can configure ***nfScreeningType*** parameter as BLOCKLIST or ALLOWLIST for the specific screening rule configuration. For more information about the screening rules applicable to different attributes in NfProfile, see Table 4-22.

Table 4-21 NF Screening Type

NfScreeningType	Description
BLOCKLIST	If the attribute is configured as Blocklist and the attribute in the request matches with the configured value, the service request is not processed further. For example, `nfFqdn` is configured as Blocklist, then the service request with `fqdn` present in the NfProfile is not processed further.
ALLOWLIST	If the attribute is configured as Allowlist and the attribute in the request matches with the configured value, then the service request is processed further. For example, `nfIpEndPointList` is configured as Allowlist, then the service request with ipv4Addresses present in the NfProfile is processed further.

- Based on the ***nfScreeningType*** parameter configuration, NRF checks screening rules, per NfType and global screening rules data level. Depending on the configuration, the service request is processed. For more information about the configuration, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

The following table describes the screening rules applicable to different attributes in NfProfile.

Table 4-22 Screening rules applicable to NfProfile attributes

Management Service Operation	Screening Rules List	Attribute in NfProfile	Attribute in REST
NF_Subscribe	CALLBACK_URI	SubscriptionData.nfStatusNotificationUri	nfCallBackUriList
NF_Register, NF_Update	CALLBACK_URI	NfService.defaultNotificationSubscriptions	nfCallBackUriList
NF_Register, NF_Update	NF_FQDN	NfProfile.fqdn	nfFqdn
NF_Register, NF_Update	NF_IP_ENDPOINT	• NfProfile.ipv4Addresses • NfProfile.ipv6Addresses • NfService.ipEndPoints	nfIpEndPointList
NF_Register, NF_Update	PLMN_ID	NfProfile.plmnList	plmnList
NF_Register, NF_Update	NF_TYPE_REGISTER	NfProfile.nfType	nfTypeList

Copyright © 2019, 2025, Oracle and/or its affiliates.

Managing NF Screening Feature

Enable

You can enable the NF Screening feature using the CNC Console or REST API.

- **Enable using REST API**: Set `featureStatus` to `ENABLED` in **NF Screening Options** configuration API. For more information about API path, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Set **Feature Status** to **ENABLED** on the NF Screening Options page. For more information about enabling the feature using CNC Console, see NF Screening Options.

Configure

You can configure the NF Screening feature using REST API or CNC Console:

- **Configure using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure using CNC Console**: Perform the feature configurations as described in NF Screening Options.

Observe

Following are the NF Screening feature specific metrics:

- ocnrf_nfScreening_nfFqdn_requestFailed_total
- ocnrf_nfScreening_nfFqdn_requestRejected_total
- ocnrf_nfScreening_nfIpEndPoint_requestFailed_total
- ocnrf_nfScreening_nfIpEndPoint_requestRejected_total
- ocnrf_nfScreening_callbackUri_requestFailed_total
- ocnrf_nfScreening_callbackUri_requestRejected_total
- ocnrf_nfScreening_plmnId_requestFailed_total
- ocnrf_nfScreening_plmnId_requestRejected_total
- ocnrf_nfScreening_nfTypeRegister_requestFailed_total
- ocnrf_nfScreening_nfTypeRegister_requestRejected_total
- ocnrf_nfScreening_notApplied_InternalError_total

For more information about NF Screening metrics and KPIs, see NF Screening Metrics and NRF KPIs sections.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts persists, perform the following:

1. **Collect the logs**: For more information about how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information about how to raise a service request.

Copyright © 2019, 2025, Oracle and/or its affiliates.

4.31 REST Based NRF State Data Retrieval

REST based NRF state data retrieval feature provides options for Non-Signaling APIs to access NRF state data. It helps the operator to access the NRF state data to understand and debug in the event of failures.

This feature provides various queries which can help to get data as per the requirement.

Managing REST Based NRF State Data Retrieval Feature

Configure

You can configure the REST Based NRF State Data Retrieval feature using REST API. For more information on state data retrieval, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.32 Key-ID for Access Token

Key-ID (kid) feature is about to add "kid" header in Access Token Response generated by NRF. As per RFC 7515 section 4.1.4, the use of Key-ID (kid) is to indicate which key was used to secure JSON Web Signature (JWS).

> **Note**
>
> You must perform OCNRF Access Token Service Operation configuration before configuring Key-ID for Access Token.

Each NRF and NF producer(s) will have multiple keys with algorithm indexed with "kid" configured. NRF REST based configuration provides options to configure different key-ids and corresponding NRF private keys and public certificates along with corresponding oauth signing algorithms. One of the configured key-ids, can be set as current Key-ID. While generating the oauth access token, NRF uses the keys, algorithm and certificates corresponding to current Key-ID.

NRF configuration provides "`addkeyIDInAccessToken`" attribute which tells to add Key-ID as header in Access Token Response or not. If value is true, then currentKeyID value will be added in "kid" header in AccessToken Response. If value is false, then "kid" header will not be added in AccessToken Response.

See *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide* for more information on how to check AccessToken Signing Key Status.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Managing Key-ID for Access Token Feature

Enable

You can enable the Key-ID for Access Token feature using the CNC Console or REST API.

Enable the Key-ID for Access Token feature using REST API as follows:

1. Use the API path as `{apiRoot}/nrf-configuration/v1/nfAccessTokenOptions`.
2. Content type must be application/json.
3. Run the PUT REST method using the following JSON:

```
{
   "addkeyIDInAccessToken": true

}
```

Enable the Key-ID for Access Token feature using CNC Console as follows:

1. From the left navigation menu, navigate to **NRF** and then select **NF Access Token Options**. The **NF Access Token Options** page is displayed.
2. Click **Edit** from the top right side to edit or update **NF Access Token Options** parameter. The page is enabled for modification.
3. Select the **Add KeyID in AccessToken** under **Token Signing Details** section as **True** from the drop-down menu.
4. Click **Save** to save the **NF Access Token Options**.

Configure

You can configure the Key-ID for Access Token Feature using REST API or CNC Console:

- **Configure Key-ID for Access Token using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure Key-ID for Access Token using CNC Console**: Perform the feature configurations as described in NF Access Token Options.

Observe

Metrics
For more information on Key-ID for Access Token metrics, see the NF Access token Metrics section.

KPIs
For more information on Key-ID for Access Token KPIs, see the NRF KPIs section.

Alerts
Following are the KeyID feature specific alerts:

- OcnrfAccessTokenCurrentKeyIdNotConfigured
- OcnrfAccessTokenCurrentKeyIdInvalidDetails
- OcnrfAccessTokenCurrentKeyIdNotConfigured
- OcnrfOauthCurrentKeyDataHealthStatus
- OcnrfOauthNonCurrentKeyDataHealthStatus

Copyright © 2019, 2025, Oracle and/or its affiliates.

- OcnrfOauthCurrentCertificateExpiringIn1Week
- OcnrfOauthNonCurrentCertificateExpiringIn1Week
- OcnrfOauthCurrentCertificateExpiringIn30days
- OcnrfOauthNonCurrentCertificateExpiringIn30days

For more information about the alerts raised for the feature, see the KeyID for AccessToken Feature section.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.33 NRF Access Token Service Usage Details

NRF implements Nnrf_AccessToken service (used for OAuth2 authorization), along with the "Client Credentials" authorization grant. It exposes a "Token Endpoint" where the Access Token Request service can be requested by NF service consumers.

The Nnrf_AccessToken service operation is defined as follows:

- Access Token Request (Nnrf_AccessToken_Get)

> **Note**
>
> This procedure is specific to the NRF Access Token service operation. NRF general configurations, database, and database-specific secret creation are not part of this procedure.

Procedure to use NRF Access Token Service Operation

This procedure provides step-by-step details to use 3GPP defined access token service operation supported by NRF.

1. **Create NRF private key and public certificate**
   This step explains the need to create the NRF private keys and public certificates.

   Private keys are used by NRF to sign the Access Token generated. It is available only with NRF.

   Public certificates are used by producer NFs to validate the access token generated by NRF. So, public certificates are available with producer network functions. NRF does not need the public certificate while signing the access token.

   The expiry time of the certificate is required to set appropriate validity time in the AccessTokenClaim.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> For more details about the validity time of AccessTokenClaim, see "`oauthTokenExpiryTime`".

Two types of signing algorithms are supported by NRF. For both types different keys and certificates required to be generated:

- ES256: ECDSA digital signature with SHA-256 hash algorithm
- RS256: RSA digital signature with SHA-256 hash algorithm

Anyone or both of the algorithm files can be generated depending upon the usage of hash algorithms. Depending upon NRF rest based configuration-specific keys and certificates are used to sign the Access Token.

> **Note**
>
> The creation process for private keys, certificates, and passwords is at the discretion of the user or operator.

Sample keys and certificates:
After running this step, the private keys and public certificates of NRF (generated files depends upon algorithms chosen by operator or user) are created. There can be multiple such pairs of private keys and public certificates which will eventually be configured in NRF with different KeyIds.

For example:

ES256 based keys and certificates:

- ecdsa_private_key.pem
- ecdsa_certificate.crt

RS256 based keys and certificates:

- rsa_private_key.pem
- rsa_certificate.crt

> **Note**
>
> - Unencrypted key and certificates are only supported.
> - For RSA, the supported versions are PKCS1 and PKCS8.
> - For ECDSA, the supported version is PKCS8.

2. **Namespace creation for Secrets**
   This step explains the need to create Kubernetes namespace in which Kubernetes secrets are created for NRF private keys and public certificates. For creating namespace, see the "Verifying and Creating NRF Namespace" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> - Different namespaces or the same namespace can be used for NRF private keys and public certificates.
> - It can be the same namespace as for NRF deployment.
> - Appropriate RBAC permission needs to be associated with the ServiceAccount, if the namespace is other than NRF's namespace.

3. **Secret creation for NRF private keys and public certificates**
   This step explains commands to create the Kubernetes secret(s) in which NRF private keys and public certificates can be kept safely. For configuring secrets, see the "Configuring Secret for Enabling Access Token Service" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

   > **Note**
   >
   > The same secret can be used which is already there for NRF private keys.
   >
   > A sample command is provided in the "Configuring Kubernetes Secret for Accessing NRF Database" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide* to create or update the secret. In case, there is a need to create or update multiple secrets for each entity, then the same section can be followed.

4. **Perform NRF REST based configuration with outcome details of Steps 1 to 3**
   This step explains the NRF REST based configuration to use the NRF private keys, public certificates, secret(s), and secret namespace(s).

   NRF REST based configuration provides options to configure different key-ids and corresponding NRF private keys and public certificates along with corresponding oauth signing algorithms. One of the configured key-ids, can be set as the current key-id.

   While generating the oauth access token, NRF uses the keys, algorithm, and certificates corresponding to the current key-id.

   For more information on NF Access Token options configuration using REST APIs, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

4.34 Access Token Request Authorization

NRF follows and supports the 3GPP TS 29.510 based verification for Access Token Authorization requests for specific NF producer based on the allowed NFType, PLMN present in the NFProfiles. Extension to this requirement is to include screening for Access Token requests based on NFType.

NRF plays a major role as an OAuth2.0 Authorization server in 5G Service based architecture. When an NF service Consumer needs to access the services of an NF producer of a particular NFType and NFInstanceId, it obtains an OAuth2 access token from the NRF. NRF performs the required authorization, and if the authorization is successful, a token is issued with the requested claims. NRF provides an option to the user to specify the authorization of the Producer-Consumer NF Types along with the producer NF's services.

Copyright © 2019, 2025, Oracle and/or its affiliates.

The operator can configure the mapping of the Requester NFType, Target NFType, and the allowed services of the Target NF. Access Token request received based on the configuration and is furthered processes the request only if the authorization is successful. Allowed Services can be configured as a single wild card '*' which denotes all the Target NFs services are allowed for the consumer NF. The operator can also configure the HTTP status code and error description that can be used in the Error Response sent by the NRF when the Access Token request is rejected.

> **Note**
>
> When Access Token Authorization feature is enabled, it is expected that the requester and the target NFs are registered in NRF for the validation. So if the targetNfType is not specifically mentioned in the request, the targetNfType is extracted from the registered profile in the database using the targetNfInstanceId. Similarly if the requesterNFInstanceId is present in the request, the requesterNfType is extracted from the registered profile in the database.

Access Token configurable attribute "`logicalOperatorForScope`" is used while authorizing the services in the Access Token Request's scope against the allowed services in the configuration. If the `logicalOperatorForScope` is set to "OR", at least one of the services in the scope will be present in the allowed services. If it is set to "AND", all the services in the scope will be present in the allowed services.

The `authFeatureConfig` attribute under `nfAccessTokenOptions` provides the support required to use NRF Access Token Request Authorization Feature. For more details, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

When `authFeatureConfig` attribute is ENABLED, the nfType validation is performed as follows:

- targetNfType and requesterNfType is matched with the nfTypes used for accessToken generation. This configuration overrides the nfType validation against the allowedNfTypes in nfProfile or nfServices.
- If the above mentioned validation is not met, then:
  - requesterNfType from the accessToken request is validated against the allowedNfTypes present in the producer NF's nfServices, if present.
  - requesterNfType from the accessToken request is validated against the allowedNfTypes from the producer nfProfile, if allowedNfTypes is not present in the nfServices.

Managing Access Token Request Authorization Feature

Enable

You can enable the Access Token Request Authorization Feature using the REST API or CNC Console.

- **Enable using REST API**: Set `featureStatus` to `ENABLED` in **NF AccessToken Options** configuration API. For more information about API path, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Set **Feature Status** to **ENABLED** on the NF Access Token Options page. For more information about enabling the feature using CNC Console, see NF Access Token Options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Configure

You can configure the Access Token Request Authorization feature using REST API or CNC Console:

- **Configure using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
  With `nfAccessTokenOptions` API, `authFeatureConfig` attribute provides the support required to use NRF Access Token Request Authorization Feature. For more details, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure using CNC Console**: Perform the feature configurations as described in NF Access Token Options.

Observe

Following are the Access Token ReSquest Authorization feature specific metrics:

- ocnrf_accessToken_rx_requests_total
- ocnrf_accessToken_tx_responses_total

For more information on Access Token Request Authorization metrics and KPIs, see NF Access token Metrics and NRF KPIs sections.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.35 Preferred Locality Feature Set

Preferred Locality Feature Set comprises of the following features:

- Preferred Locality
- Extended Preferred Locality with Location or Location Sets
- Limiting the Number of Producers Based on NF Set Ids

For more information about these features and their processing, see the following sections.

- Preferred Locality
- Extended Preferred Locality
- Discovery Processing for Extended Preferred Locality Feature
- Configuring Extended Preferred Locality

Copyright © 2019, 2025, Oracle and/or its affiliates.

4.35.1 Preferred Locality

When the consumer NF sends the discovery query along with an additional attribute “**preferred-locality**”, the **Preferred Locality** feature is applied. By default, such discovery queries are processed as follows:

1. NRF searches and collects all the NFs profiles matching the search criteria sent in the discovery query except the "**preferred-locality**" attribute.
2. NF profiles collected in the above step are arranged as per "**preferred-locality**". Here, the NFs matching the "**preferred-locality**" are arranged as per the increasing order of their priority and the NF profiles that do not have "**preferred-locality**" are then placed in the increasing order of post-processed priority value. Here, post-processed priority value means adding the highest NF priority of the NFs that are matching with the "**preferred-locality**" to the priority value of non-preferred NF profiles with an additional increment offset value of 1.
   For example: If the highest NF priority value of "**preferred-locality**" is 10 and the priority value of non-preferred NF profile is 5, then after post processing the priority value of non-preferred NF profile will be 16 (where, 10 is the highest priority value of NF profile with "**preferred-locality**", 5 is the priority of non-preferred NF and 1 is the offset value).

   For service-name based discovery query, if there is single service in the NF profile after discovery processing, service level priority is updated.

   For service-name based discovery query, if there are multiple services in the NF profile with same service name after discovery processing, along with priority value update at NF profile, service level priority is also updated.

Managing Preferred Locality feature

Enable
Enabling the **Preferred Locality** feature: This feature is enabled by default as per 3GPP TS29.510. There is no option to enable or disable it explicitly.

Observe

There are no specific metrics and alerts for this feature.

For the entire list of NRF metrics and alerts, see NRF Metrics and NRF Alerts section.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.35.2 Extended Preferred Locality

The limitation with the default "**Preferred Locality**" feature is that, it does not provide a mechanism to select matching producers from more than one "**preferred-locality**" because "**preferred-locality**" attribute allows to provide only one location in the discovery query.

Copyright © 2019, 2025, Oracle and/or its affiliates.

To overcome this limitation, the "**Extended Preferred Locality**" feature is implemented where a preferred locality triplet, that is, collection of primary, secondary and tertiary preferred locality is associated with the given "**preferred-locality**" and "**target-nf- type**". The "Primary Preferred Locality", "Secondary Preferred Locality", and "Tertiary Preferred Locality" in "Preferred Locality Triplet" can be configured with single location or a location set. That is, the "**Location**" attribute under "**Target Preferred Locations**" can be an individual "**Location**" or a "**Location Set Name**" configured under the "**Location Sets**". You can configure a maximum of 255 location sets, and each location set can have upto 10 locations.

Managing Extended Preferred Locality and Limiting the Number of Producers Based on NF Set Ids

Enable
For enabling the features, see below:

- **Enable using REST API**: Set `featureStatus` to `ENABLED` in **NF Discovery Options** configuration API. For more information about the API path, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Set **Feature Status** to **ENABLED** on the NF Discovery Options page. For more information about enabling the feature using CNC Console, see NF Discovery Options.

Configure

You can configure the Extended Preferred Locality feature using REST API or CNC Console:

- **Configure using REST API**: Perform the feature configurations in the *{apiRoot}/nrf-configuration/v1/nfDiscoveryOptions* as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure using CNC Console**: Perform the feature configurations as described in NF Discovery Options.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.35.2.1 Discovery Processing for Extended Preferred Locality Feature

Discovery query is processed with respect to Extended Preferred Locality Feature is as follows:

1. NRF filters out all of the NFs profiles matching the search criteria sent in the discovery request except the "preferred-locality" query parameter.
2. NFProfiles filtered in the above step are then arranged matching the locality attribute as per the configuration for "**Primary Preferred Locality**", "**Secondary Preferred Locality**", "**Tertiary Preferred Locality**" in "Preferred Locality Triplet". The NFs profiles that are not matching with the "**preferred-locality**" query parameter are kept in a different group.

Copyright © 2019, 2025, Oracle and/or its affiliates.

3. NFProfiles in each "Preferred Locality Triplet" will be arranged based on their priority. For NFProfiles having the same priority, load will be used for for arranging the NFProfiles. If load values are also same, then the NFProfiles will be arranged in a random order.
4. Additionally, priorities of NFProfiles falling in the groups of secondary, tertiary, and remaining non-matching locations are updated based upon a similar way of post-processed priority as defined in the "Preferred Locality" feature.

> **Note**
>
> When the "**Extended Preferred Locality**" feature is enabled and in case, if any of the required configurations are not matched with the respective discovery query parameters, NRF will apply the "**Preferred Locality**" feature.

5. Locations or a set of locations can be defined in "Preferred Locality Triplet" for the discovery search query.

4.35.2.2 Configuring Extended Preferred Locality

According to 3GPP TS 29.510 V15.5.0, consumer NFs can send discovery query with **preferred-locality**, along with **requester-nf-type**, **target-nf-type**, and **nf-services**.

> **Note**
>
> Only "**preferred-locality**", "**target-nf-type**", and "**nf-services"** attributes are considered for this feature, other attributes in the search query are not applicable.

1. Configure the location types as per the network deployment. Location types can be configured using `locationTypes` attribute.
   For Example: Category-S, Category-N, and so on.

   Sample location types attribute value:

   ```
   "locationTypes": ["Category-S", "Category-N", "Category-x"]
   ```

   > **Note**
   >
   > - Maximum 25 location types can be configured.
   > - `locationTypes` can have minimum 3 characters and maximum 36 characters. It can have only alphanumeric and special characters '-' and '_'. It cannot start and end with special characters.
   > - Duplicate location types are not allowed to be configured in this attribute.
   > - `locationTypes` are case sensitive it means Category-x and Category-X are different.

2. Configure NF types (along with different services) under `locationTypeMapping` attribute. This attribute allows the operator to create mapping between different `nfType` (along with `nfServices`) and the `locationTypes`. (Location types is already configured by following Step 1 above).
   For Example: PCF is the NF type, am_policy, bdt_policy are NF services and Category-x is the location type.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Sample output:

```
"locationTypeMapping": [{
      "nfType": "PCF",
      "nfServices": ["am_policy", "bdt_policy"],
      "locationType": "Category-x"
   }],
```

Note

- Configure **nfType** attribute to map it with the "**target-nf-type**" attribute in incoming discovery query.
- `nfServices` attribute of this configuration is used when "**service-names**" query parameter is present in discovery query. Different `nfServices` along with `nfType` can be mapped with different location types.
- In case, if `nfServices` is not required to be mapped with any particular location type, then configure the value of `nfServices` as '*'. This indicates that `locationType` mapped all NF services of that "**target-nf-type**".
- '*' value record for "**target-nf-type**" is used when no configuration is found corresponding to `service-names` attribute of discovery query.
- If "**service-names**" attribute is unavailable in the discovery query, then '*' value is used for `locationType` selection.
- "**service-names**" of discovery search query can be subset of configured `nfServices` attribute.
  For Example:
  - If sm_policy is the only `service-names` present in discovery query and configured value for `nfServices` are sm_policy, bdt_policy. Then this configuration is used for `locationType` selection.
  - If am_policy, nudr_policy are the `service-names` present in discovery query and configured value for `nfServices` are am_policy, nudr_policy, xyz_policy. Then this configuration is used for `locationType` selection.
- Same `nfServices` cannot be mapped to different location type.
  For example:

  One `locationType` say Category-N is already mapped to `nfServices` with value 'sm_policy' with `nfType` as 'PCF', then sm_policy (individual or with group of NF services) cannot be mapped to another location type for 'PCF'.
- In case, the `service-names` attribute of search query has multiple `nfServices` but in configured `nfServices` attribute (individual or with group of NF services) are mapped to different `locationType` then '*' value record for `target-nf-type` is used for location type selection.
- Maximum 100 `locationTypeMapping` values can be configured.

Sample locationTypeMapping attribute value

```
"locationTypeMapping": [{
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
    "nfType": "PCF",
    "nfServices": ["am_policy", "bdt_policy"],
    "locationType": "Category-x"
  },
  {
    "nfType": "PCF",
    "nfServices": ["sm_policy"],
    "locationType": "Category-N"
  },
  {
     "nfType": "PCF",
     "nfServices": ["*"],
     "locationType": "Category-S"
  },
  {
     "nfType": "AMF",
     "nfServices": ["*"],
     "locationType": "Category-S"
   }
]
```

3. Configure the `preferredLocationDetails` corresponding to `locationType` selected and `preferred-locality`. The `preferredLocation` attribute defines the `preferred-locality` (derived from discovery search query) and `locationType` selected from the `locationTypeMapping` attribute maps to a `preferredLocationDetails`.

> **Note**
>
> - Maximum 650 `preferredLocationDetails` values can be configured.
> - Different priorities can be configured for preferred locations.
> - `preferredLocation` attribute for this configuration is mapped to `preferred-locality` (derived from discovery search query).
> - `targetLocationType` attribute for this configuration is mapped to `locationType` selected from `locationTypeMapping` attribute.
> - Corresponding to `preferredLocation` and `targetLocationType` attributes, `targetPreferredLocations` can be configured. `targetPreferredLocations` are defined by operator.
> - `targetPreferredLocations` attribute can have upto 3 target preferred locations.
> - `priority` is assigned to `targetPreferredLocations`. There cannot be same priority to different `targetPreferredLocations`.
> - `targetPreferredLocations` can be same as `preferredLocation` attribute value.
> - In case the location attribute in `targetPreferredLocations` is a set of location, then configure the `locationSets` attribute:
>   - Maximum of 255 locationSets can be configured.
>   - Length of the location name can be in the range of 5 to 100 characters.
>   - Maximum of 10 locations can be configured.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Sample preferredLocationDetails attribute value

```
"preferredLocationDetails": [{
                "preferredLocation": "Azusa",
                "targetLocationType": "Category-x",
                "maxNFProfilesFromFirstMatchLoc":0,
                "targetPreferredLocations": [{
                    "priority": 1,
                    "location": "Azusa"
                }, {
                    "priority": 2,
                    "location": "Vista"
                }, {
                    "priority": 3,
                    "location": "Ohio"
                }]
            },
            {
                "preferredLocation": "Azusa",
                "targetLocationType": "Category-y",
                "maxNFProfilesFromFirstMatchLoc":0,
                "targetPreferredLocations": [{
                    "priority": 1,
                    "location": "RKL"
                }, {
                    "priority": 2,
                    "location": "CSP"
                }, {
                    "priority": 3,
                    "location": "West-Region-Edge-Set01"
                }]
            }
        ],
        "locationSets": [
            {
                "locationSetName" : "West-Region-Edge-Set01",
                "locations": ["LA", "SFO"]
            }
        ]]
```

4.35.2.3 Limiting the Number of Producers Based on NF Set Ids

"**Limiting the Number of Producers Based on NF Set Ids**" feature is a supplement to the "**Extended Preferred Locality**" feature.

In the **Extended Preferred Locality** feature, if there are more number of producers that are matching in each location (primary, secondary, tertiary), then NRF potentially returns all the matching producer NFs. For Example, If each location is having 10 matching producer, NRF ends up in sending 30 producers in discovery response. This results in returning all the 30 producers in a message being too long which ends up in not using the network resources efficiently.

To enhance the producer selection, NRF supports to limit the number of producers by using `nfSetIdList` as follows:

Copyright © 2019, 2025, Oracle and/or its affiliates.

- After NF profile selection is done based on the extended preferred locality logic, from the first matching location, only a limited number of NF profile will get selected based on the configuration attribute "**Maximum number of NF Profiles from First Matching Location**." The first matching location is the location where the first set of matching producers are found from the Preferred Locality Triplet.
- From the remaining locations, only those producers will be shortlisted whose `nfSetIdList` matches with the NF Set Ids of the first matching location's producers (after limiting feature is applied). As these producers can be used as alternate producers during failover scenarios.

In case of the following scenarios, NRF fallbacks to the "**Extended Preferred Locality**" feature from the "**Limiting the Number of Producers Based on NF Set Ids**" feature.

- if the value of the **Maximum number of NF Profiles from First Matching Location** attribute is configured as 0.
- if any one of the NF profiles from the first matching location after limiting the profiles, does not have the `nfSetIdList` attribute (that is the profiles are registered without NF Set Ids).

> **Note**
>
> - It is recommended to enable this feature only if NFs are registered with the `nfSetIdList` attribute.
> - In upgrade scenarios, if an extended preferred locality is configured, then for each preferred locality attribute the value of **Maximum number of NF Profiles from First Matching Location** attribute will become 0, which disables the feature by default.

Managing Limiting the Number of Producers Based on NF Set Ids

Enable

For enabling the features, see below:

- **Enable using REST API**: Enabling **Limiting the Number of Producers Based on NF Set Ids** feature: Set the value of the `maxNFProfilesFromFirstMatchLoc` field to a value greater than 0 in *{apiRoot}/nrf-configuration/v1/nfDiscoveryOptions* configuration API. For more information about the API path, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Enabling **Limiting the Number of Producers Based on NF Set Ids** feature: Set the value of **Maximum number of NF Profiles from First Matching Location** attribute to a number greater than 0 under **Preferred Location Details** on the NF Discovery Options page. For more information about enabling the feature using CNC Console, see NF Discovery Options.

> **Note**
>
> Extended Preferred Locality feature should be enabled prior enabling the Limiting the Number of Producers Based on NF Set Ids feature.

Configure

You can configure the Limiting the Number of Producers Based on NF Set Ids features using REST API or CNC Console:

Copyright © 2019, 2025, Oracle and/or its affiliates.

- **Configure using REST API**: Perform the feature configurations in the *{apiRoot}/nrf-configuration/v1/nfDiscoveryOptions* URI as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Configure using CNC Console**: Perform the feature configurations as described in NF Discovery Options.

Observe

The following metrics are added for the **Limiting the Number of Producers Based on NF Set Ids** feature:

- `ocnrf_nfDiscover_limiting_profile_count_for_nfSet_total`
- `ocnrf_nfDiscover_limiting_profiles_not_applied_for_nfSet_total`

For more information on metrics, see NRF Metrics section.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.35.2.3.1 Discovery Processing for Limiting Number of Producers Based on NF Set Ids

Discovery query is processed with respect to Limiting the Number of Producers Based on NF Set Ids feature is as follows:

1. Perform the same steps with the discovery processing as mentioned in Discovery Processing for Extended Preferred Locality Feature.
2. When the attribute **Maximum number of NF Profiles from First Matching Location** is configured to a value greater than 0, NRF allows the consumer NF to select the top "n" number of producers from the first preferred locality. And from the remaining preferred localities, only those producers are selected that have matching `nfsetIdList` of the top "n" producers selected from the first preferred locality.

4.35.2.3.2 Configuring Limiting Number of Producers Based on NF Set Ids

This section describes about Configuring Limiting Number of Producers Based on NF Set Ids feature.

1. Perform the same steps with the configuration as mentioned in Configuring Extended Preferred Locality.
2. Configure the `maxNFProfilesFromFirstMatchLoc` attribute in the `preferredLocationDetails` to limit the number of producers based on NF Set Ids from the first matching location in a preferred locality triplet. If the value of `maxNFProfilesFromFirstMatchLoc` attribute is greater than 0, then the feature is enabled. Sample output:`"maxNFProfilesFromFirstMatchLoc":0`, indicates feature is disabled.

```
"preferredLocationDetails": [{
                "preferredLocation": "Azusa",
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
            "targetLocationType": "Category-x",
            "maxNFProfilesFromFirstMatchLoc":0,
            "targetPreferredLocations": [{
                "priority": 1,
                "location": "Azusa"
            }, {
                "priority": 2,
                "location": "Vista"
            }, {
                "priority": 3,
                "location": "Ohio"
            }]
        },
        {
            "preferredLocation": "Azusa",
            "targetLocationType": "Category-y",
            "maxNFProfilesFromFirstMatchLoc":0,
            "targetPreferredLocations": [{
                "priority": 1,
                "location": "RKL"
            }, {
                "priority": 2,
                "location": "CSP"
            }, {
                "priority": 3,
                "location": "West-Region-Edge-Set01"
            }]
        }
    ],
    "locationSets": [
        {
            "locationSetName" : "West-Region-Edge-Set01",
            "locations": ["LA", "SFO"]
        }
    ]]
```

4.35.3 NFService Priority Update

This feature is an extension of the existing functionality of the Preferred Locality Feature Set. For more details, see Preferred Locality and Extended Preferred Locality.

With this feature, NRF now updates the NFService level priority along with the NFProfile level priority while processing the discovery query. NRF updates the following:

- NFProfile level priority considering the lowest NFProfile level priority of NFProfiles
- NFService level priority considering the lowest NFService level priority of all NFServices

Impact on Preferred Locality feature

For the **Preferred Locality** feature, there can be two groups of NFProfiles. One group contains NFProfiles with locality attribute matching preferred-locality attribute query parameter and another group with the rest of the NFProfiles with non-matching preferred-locality query parameter.

With this NF Service priority feature, NFProfile level priority is updated for the second group using the Lowest NFProfile Level Priority value of the first group. Similarly, NFService level

Copyright © 2019, 2025, Oracle and/or its affiliates.

priority is updated for the second group using the Lowest NFService Level Priority value among all the NFServices present in the first group.

Priority Value Calculation:

- NFProfile Level Priority = Actual NFProfile Level Priority Value + Lowest NFProfile Level Priority of the first group + Offset Value (that is, 1)
- NFService Level Priority = Actual NFService Level Priority Value + Lowest NFService Level Priority among all the NFServices present in the first group + Offset Value (that is, 1)

Impact on Extended Preferred Locality feature

For the **Extended Preferred Locality** feature, there can be multiple groups of NFProfiles depending on the Extended Preferred locality feature configuration. The first group contains NFProfiles with locality attribute configured with the first matching location/location set. And subsequent group(s) can be present depending upon the configured locations and the matching NFProfiles.

With NF Service priority feature, NFProfile level priority is updated for the second group onwards using the Lowest NFProfile Level Priority of the previous group. Similarly, NFService level priority is updated for the second group onwards using the Lowest NFService Level Priority among all the NFServices of its previous group.

Priority Value Calculation:

- NFProfile Level Priority = Actual NFProfile Level Priority Value + Lowest NFProfile Level Priority of the previous group + Offset Value (that is, 1)
- NFService Level Priority = Actual NFService Level Priority Value + Lowest NFService Level Priority among all the NFServices present in the previous group + Offset Value (that is, 1)

In case, priority or load is not present in a NFService/NFProfile, then the values are considered as below:

- For NFProfile level, if priority or load values are not present, then the configured `defaultPriority` and `defaultLoad` values are considered respectively. Similarly, for NFService level, if priority or load values are not present, then the NFProfile level priority or load values are considered respectively. However, if priority or load values are not present in NFProfile as well, then the configured `defaultPriority` and `defaultLoad` values are considered respectively.
  For more information about these parameters, see General Options section.
- In case, priority or load attributes are not present in the registered NFProfile or its NFServices, then the final calculated value is set based on the configured `defaultPriorityAssignment` and `defaultLoadAssignment` flags respectively.
  For more information about these parameters, see General Options section.

This feature can be enabled or disabled using the `servicePriorityUpdateFeatureStatus` flag in the *{apiRoot}/nrf-configuration/v1/nfDiscoveryOptions* API or **Service Priority Update Feature Status** flag in the NF Discovery Options page in the CNC Console.

Managing NFService Priority Update

Enable
To enable the features, see below:

- **Enable using REST API**: Set the value of the `servicePriorityUpdateFeatureStatus` to `ENABLED` in the *{apiRoot}/nrf-configuration/v1/nfDiscoveryOptions* configuration API. For more information about the API, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- **Enable using CNC Console**: Set the value of the **Service Priority Update Feature Status** to **ENABLED** on the **NF Discovery Options** page. For more information about enabling the feature using CNC Console, see NF Discovery Options.

Configure

You can configure the NFService Priority Update feature using REST API or CNC Console:

- **Configure using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Sample configuration:

```
"profilesCountInDiscoveryResponse": 3,
"discoveryResultLoadThreshold": 0,
"servicePriorityUpdateFeatureStatus": "ENABLED"
 "discoveryValidityPeriodCfg":
```

- **Configure using CNC Console**: Perform the feature configurations as described in NF Discovery Options.

Observability

Metrics

There are no new metrics added to this feature.

Alerts

There are no alerts added or updated for this feature.

KPIs

There are no new KPIs related to this feature.

Maintain

If you encounter alerts at system or application levels, see the NRF Alerts section for the resolution steps.

In case the alerts still persist, perform the following:

1. **Collect the logs**: For more information on collecting logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.

2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.35.3.1 Discovery Processing in NFService Priority Update with preferred-locality

NRF receives the discovery request with the preferred-locality query parameter. It checks the status of **servicePriorityUpdateFeatureStatus** feature flag. If this feature flag is enabled, perform the following steps:

1. Filter out the NF Profiles matching the preferred-locality attribute of NFDiscover service operation query.
    - With preferred locality feature, the NF Profiles can be filtered into two groups:

Copyright © 2019, 2025, Oracle and/or its affiliates.

  - the matching location group
  - the non-matching location group
- With the Extended Preferred Locality feature enabled, the NF Profiles can be filtered into four groups:
  - the first matching location group
  - the second matching location group
  - the third matching location group
  - the non-matching location group

2. Order the NF Profiles based on both priority and load values of NFService level and/or NF Profile level depending on the scenarios where Extended Preferred Locality feature is enabled or disabled.
3. Modify and Set the priority for both NFService and NFProfile level, based on the offset calculation logic. For more information, regarding the priority value calculation and offset calculation logic see the priority value calculation in NFService Priority Update.
4. NRF continues with the rest of the discovery flow.

If the feature is disabled, NRF continues with the existing flow skipping the above steps.

4.35.3.2 Configuring NFService Priority Update

If the value of the `servicePriorityUpdateFeatureStatus` is `ENABLED`, NRF updates the NFService level priority along with the NFProfile level priority. NRF updates the NFProfile level priority considering the lowest NFProfile level priority of NFProfiles based upon specific preferred locality features and NFService level priority considering the lowest NFService level priority of NFServices based upon specific preferred locality features.

By default, the value of this `servicePriorityUpdateFeatureStatus` is DISABLED.

Enable this feature in the *{apiRoot}/nrf-configuration/v1/nfDiscoveryOptions* API.

```
"servicePriorityUpdateFeatureStatus": "ENABLED"
```

4.36 Roaming Support

NRF supports the 3GPP defined inter-PLMN routing for NRF specific service operations like NFDiscover, AccessToken, NFStatusSubscribe, NFStatusUnSubscribe, and NFStatusNotify. To serve 5G subscribers roaming in a non-home network also known as visited or serving networks, consumer network functions in visited or serving networks need to access the NF Profiles located in the home network of 5G subscribers.

In this process, the consumer NFs sends the NRF specific service operations towards NRF which is in visited or serving network. Then, the visited or serving NRF routes these service operations towards the home NRF through SEPPs in visited or serving and home networks. NFDiscover, AccessToken, NFStatus Subscribe, and NFStatus UnSubscribe service operations are routed from visited or serving NRF to home NRF. NFStatusNotify service operation is initiated by the home network NRF towards consumer NFs residing in visited or serving network for inter-PLMN specific subscriptions.

> **Note**
>
> XFCC specific validations are not supported for inter-PLMN service operations.

Copyright © 2019, 2025, Oracle and/or its affiliates.

3GPP specific attributes defined for different service operations plays a deciding role during inter-PLMN message routing.

There are two important terminology used in roaming mechanism:

- vNRF - visited or serving NRF when subscribers are roaming in non-home network.
- hNRF - NRF in home network of the subscribers.

Table 4-23 3GPP Attributes

Attribute Name	Service Operation	Details
requester-plmn-list	NFDiscover	If the requester-plmn-list matches with NRF PLMN list, it means that the NRF will function as vNRF. If the requester-plmn-list not matches with NRF PLMN list, it means that the NRF will function as hNRF.
target-plmn-list	NFDiscover	When an NRF is vNRF, the target-plmn-list becomes a mandatory attribute to decide the target PLMN. In case the NRF is hNRF, this value can be optional but if it is present, then the value matches with hNRF PLMN.
requesterPlmnList	AccessToken	If the requesterPlmnList matches with the NRF PLMN list, it means that the NRF will function as vNRF. If the requesterPlmnList does not match with the NRF PLMN list, it means that the NRF will function as hNRF.
requesterPlmn	AccessToken	If the requesterPlmn matches with the NRF PLMN list, it means that the NRF will function as vNRF. If the requesterPlmn does not match with the NRF PLMN list, it means that the NRF will function as hNRF. In case both the requesterPlmnList and requesterPlmn attributes are present, then combined PLMN values are used.
targetPlmn	AccessToken	When an NRF is vNRF, the targetPlmn is considered to decide the target PLMN. In case the NRF is hNRF, this value must match with the hNRF PLMN.
nfType	AccessToken	When an NRF is hNRF, then this value is used for NRF NfAccessToken Authorization Feature. But in case this value is not present, then the user-agent header is used. User-Agent header is present with 3GPP defined nfType. In case this header is not present, see `userAgentMandatory` attribute for more details.
reqPlmnList	NFStatusSubscribe	If the reqPlmnList matches with the NRF PLMN list, it means that the NRF will function as vNRF. If the reqPlmnList does not match with the NRF PLMN list, it means that the NRF will function as hNRF.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-23 (Cont.) 3GPP Attributes

Attribute Name	Service Operation	Details
nfStatusNotificationURI	NFStatusSubscribe	In case reqPlmnList attribute is not present in Subscription data, then NRF will check that nfStatusNotificationURI is in inter-PLMN format which is `5gc.mnc(\d\d\d).mcc(\d\d\d).3gppnetwork.org format`. Sample: 5gc.mnc310.mcc314.3gppnetwork.org In case nfStatusNotificationURI is in this format, then this is used to find the role of NRF. However, if reqPlmnList is present, then this attribute have to be in inter-PLMN format. This will help when hNRF generate the notification towards visited or serving network.
plmnId	NFStatusSubscribe	When an NRF is vNRF, then the plmnId is considered to decide the target PLMN. In case the NRF is hNRF, this value can be optional but if it is present, then the value shall match with the hNRF PLMN.
subscriptionId	NFStatusSubscribe, NFStatusUnSubscribe	subscriptionId also plays an important role during roaming cases. Subscription Id generated by hNRF will have 'roam' keyword prefixed in beginning of subscriptionId. It helps to identify the inter-PLMN request for subsequent service operations NFStatusSubscribe - Update and NFStatusUnSubscribe.

Enable

You can enable the Roaming Options feature using REST API or CNC Console.

- **Enable using REST API**: Set `featureStatus` to `ENABLED` in **Roaming Options** configuration API. For more information about API path, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Set **Feature Status** to **ENABLED** on the Roaming Options page. For more information about enabling the feature using CNC Console, see Roaming Options.

> **Note**
>
> Before enabling the `featureStatus` attribute, ensure that the corresponding configurations are completed.

Configure

You can configure the Roaming Options using REST API or CNC Console:

- **Configure NRF Roaming Options using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- Create or update {apiRoot}/nrf/nf-common-component/v1/egw/peerconfiguration. Update host with SEPP host.

```
curl -v -X PUT "http://10.75.226.126:30747/nrf/nf-common-
component/v1/egw/peerconfiguration" -H  "Content-Type: application/
json"  -d @peer.json

peer.json sample:-
[
  {
    "id": "peer1",
    "host": "sepp-stub-service",
    "port": "8080",
    "apiPrefix": "/",
  }
]
```

- Create or update the {apiRoot}/nrf/nf-common-component/v1/egw/peersetconfiguration to assign these peers.
- Create or update {apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerrorcriteriasets.
- Create or update {apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerroractionsets.
- Update the {apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration as mentioned below:

```
curl -v -X PUT "http://10.75.226.126:32247/nrf/nf-common-
component/v1/egw/routesconfiguration" -H  "Content-Type: application/
json"  -d @header.json

sample header.json:-
[
    {
        "id":"egress_sepp_proxy1",
        "uri": "http://localhost:32068/",
        "order": 0,
        "sbiRoutingConfiguration": {
            "enabled": true,
            "peerSetIdentifier": "set0"
        },
        "httpRuriOnly": true,
        "httpsTargetOnly": true,
        "predicates": [{
            "args": {
                "header": "OC-MCCMNC",
                "regexp": "310014"
            },
            "name": "Header"
        }],
        "filters": [{
            "name": "SbiRouting"
        }, {
            "args": {
                "retries": "3",
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
                "methods": "GET, POST, PUT, DELETE, PATCH",
                "statuses": "BAD_REQUEST, INTERNAL_SERVER_ERROR,
BAD_GATEWAY, NOT_FOUND, GATEWAY_TIMEOUT",
                "exceptions":
"java.util.concurrent.TimeoutException,java.net.ConnectException,java.ne
t.UnknownHostException"
            },
            "name": "SBIReroute"
        },{
            "args": {
                "name": "OC-MCCMNC"
            },
            "name": "RemoveRequestHeader"
        }],
        "metadata": {
        }
    },
    {"id":"default_route","uri":"egress://
request.uri","order":100,"predicates":[{"args":{"pattern":"/
**"},"name":"Path"}]}
]
```

For more information about the routes configuration on different deployments, see Egress Gateway Route Configuration for Different Deployments.

- **Configure NRF Roaming Options using CNC Console**: Perform the feature configurations as described in Roaming Options.

Observe

Following are the NRF Roaming Options specific metrics filters:

- ocnrf_roaming_nfStatusSubscribe_rx_requests_total
- ocnrf_roaming_nfStatusSubscribe_tx_responses_total
- ocnrf_roaming_nfStatusSubscribe_tx_requests_total
- ocnrf_roaming_nfStatusSubscribe_rx_responses_total
- ocnrf_roaming_nfStatusSubscribe_rx_requests_total
- ocnrf_roaming_nfStatusSubscribe_tx_responses_total
- ocnrf_roaming_nfStatusSubscribe_tx_requests_total
- ocnrf_roaming_nfStatusSubscribe_rx_responses_total
- ocnrf_roaming_nfStatusUnSubscribe_rx_requests_total
- ocnrf_roaming_nfStatusUnSubscribe_tx_responses_total
- ocnrf_roaming_nfStatusUnSubscribe_tx_requests_total
- ocnrf_roaming_nfStatusUnSubscribe_rx_responses_total
- ocnrf_roaming_nfDiscover_rx_requests_total
- ocnrf_roaming_nfDiscover_tx_responses_total
- ocnrf_roaming_nfDiscover_tx_requests_total
- ocnrf_roaming_nfDiscover_rx_responses_total
- ocnrf_roaming_accessToken_rx_requests_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

- ocnrf_roaming_accessToken_tx_responses_total
- ocnrf_roaming_accessToken_tx_requests_total
- ocnrf_roaming_accessToken_rx_responses_total
- ocnrf_roaming_nfStatusNotify_tx_requests_total
- ocnrf_roaming_nfStatusNotify_rx_responses_total
- ocnrf_roaming_jetty_latency_seconds

For more information on NRF Roaming Options metrics and KPIs, see Roaming Support Metrics, and NRF KPIs sections.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.37 EmptyList in Discovery Response

NRF checks the NFStatus of the target-nf-type in the NRF state data. It identifies whether the current status of the matching NFProfile is SUSPENDED.

If the NFStatus of all matching profiles is SUSPENDED, then NRF modifies the NFStatus of these profiles as "REGISTERED". The modified NFStatus is sent in the discovery response with a shorter validity period so that a call can be established. Upon expiry of the validityPeriod, the Requester NF must rediscover the target NF.

> **Note**
>
> Change in the NFStatus of target-nf-type to REGISTERED is not stored in the database.

In case forwarding is ENABLED for a discovery request, NRF forwards the request to identify if there are any matching profiles in another region:

- If matching profiles are found, then that profile is sent in the discovery response.
- If the NFStatus of all the matching profiles are in SUSPENDED state, then these profiles are sent in the discovery response as REGISTERED with a shorter validity period.

> **Note**
>
> Once the profile moves from SUSPENDED state to NOT AVAILABLE state (after 7 days), NRF will start choosing the NFProfile containing `backupInfoAmfFailure` instead of the NFProfile containing `backupInfoAmfRemoval`.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-24 Discovering NF Profiles for AMF with Guami

Scenario	Profile A that contains match with the guami attribute.	Profile B that contains match with the backupInfoAmfFailure attribute	Profile C that contains match with the backupInfoAmfRemoval attribute	Expected Discovery Response	Notes

-	<pre>{
	  "guami": {
	    "plmnId": {
	      "mcc":
	"594",
	      "mnc":
	"75"
	    },
	    "amfId":
	"947d18"
	  }
	}</pre>
		<pre>{
		
		"backupInfoAmfFa
		ilure": [
		    {
		      "plmnId":
		{
		        "mcc":
		"594",
		        "mnc":
		"75"
		      },
		      "amfId":
		"947d18"
		    }
		  ]
		}</pre>
			<pre>{
			
			"backupInfoAmfRe
			moval": [
			    {
			      "plmnId":
			{
			        "mcc":
			"594",
			        "mnc":
			"75"
			      },
			      "amfId":
			"947d18"
			    }
			  ]
			}</pre>
				-	-

1	REGISTERED	NA	NA	Profile A	Profile A is registered and matches the guami.
2	SUSPENDED PROBABLE_SUSPENDED	REGISTERED	NA	Profile B	Profile B acts as the alternative to SUSPENDED Profile A

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-24 (Cont.) Discovering NF Profiles for AMF with Guami

Scenario	Profile A that contains match with the guami attribute.	Profile B that contains match with the backupInfoAmfFailure attribute	Profile C that contains match with the backupInfoAmfRemoval attribute	Expected Discovery Response	Notes
3	SUSPENDED (and NOT AVAILABLE) PROBABLE_SUSPENDED	SUSPENDED PROBABLE_SUSPENDED UNDISCOVERABLE DEREGISTERED NOT AVAILABLE	REGISTERED	Profile C	There is no alternate REGISTERED profile for the SUSPENDED Profile matching the backupInfoAmfFailure Hence the alternate for NOT AVAILABLE Profiles is Profile B.
4	SUSPENDED (and NOT AVAILABLE) PROBABLE_SUSPENDED	SUSPENDED PROBABLE_SUSPENDED UNDISCOVERABLE DEREGISTERED NOT AVAILABLE	SUSPENDED PROBABLE_SUSPENDED UNDISCOVERABLE DEREGISTERED NOT AVAILABLE	Profile A	No Alternate REGISTERED Profiles for SUSPENDED and NOT AVAILABLE profile are present. Hence empty response feature is applied and Profile A is selected.
5	UNDISCOVERABLE DEREGISTERED NOT AVAILABLE	NA	REGISTERED	Profile C	Profile C is the alternate to NOT AVAILABLE profiles.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-24 (Cont.) Discovering NF Profiles for AMF with Guami

Scenario	Profile A that contains match with the guami attribute.	Profile B that contains match with the backupInfoAmfFailure attribute	Profile C that contains match with the backupInfoAmfRemoval attribute	Expected Discovery Response	Notes
6	UNDISCOVERABLE DEREGISTERED NOT AVAILABLE	NA	SUSPENDED PROBABLE_SUSPENDED	Profile C	There are no REGISTERED profiles or alternates for it. Empty response feature is applied on Profile C.
7	UNDISCOVERABLE DEREGISTERED NOT AVAILABLE	NA	UNDISCOVERABLE DEREGISTERED NOT AVAILABLE	NO Profile	There are no alternate REGISTERED profiles for SUSPENDED or NOT AVAILABLE profiles. Empty List feature can't be applied to Profile A or Profile C as they are not in suspended state. Hence no profile is sent in discovery response.
8	Few: SUSPENDED Few: DEREGISTERED or UNDISCOVERABLE	REGISTERED	SUSPENDED or PROBABLE_SUSPENDED or DEREGISTERED or NOT AVAILABLE or UNDISCOVERABLE	Profile B	Profile B acts as the alternative to SUSPENDED Profile A

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-24 (Cont.) Discovering NF Profiles for AMF with Guami

Scenario	Profile A that contains match with the guami attribute.	Profile B that contains match with the backupInfoAmfFailure attribute	Profile C that contains match with the backupInfoAmfRemoval attribute	Expected Discovery Response	Notes
9	Few: SUSPENDED Few: DEREGISTERED or UNDISCOVERABLE	SUSPENDED or PROBABLE_SUSPENDED or DEREGISTERED or NOT AVAILABLE or UNDISCOVERABLE	REGISTERED	Profile C	There is no alternate REGISTERED Profile for SUSPENDED profiles. Profile C is REGISTERED alternate for NOT AVAILABLE Hence Considering Profile C.
10	Few: SUSPENDED Few: DEREGISTERED or UNDISCOVERABLE	SUSPENDED PROBABLE_SUSPENDED	SUSPENDED or PROBABLE_SUSPENDED or DEREGISTERED or NOT AVAILABLE or UNDISCOVERABLE	Profile A	There are no REGISTERED alternates for any state. Hence empty response feature is applied on Profile A.

Managing EmptyList Feature

Enable

You can enable EmptyList feature using the REST API or CNC Console.

- **Enable using REST API**: Set `emptyListFeatureStatus` to `ENABLED` in **nfDiscovery Options** configuration API. Also, set `featureStatus` to `ENABLED` to configure EmptyList for a particular nfType. For more information about API path, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Set **Feature Status** to **ENABLED** for Empty Discovery Response. Also, set **Feature Status** to **ENABLED** to configure EmptyList for a particular nfType. For more information about enabling the feature using CNC Console, see NF Discovery Options.

Observe

Metrics

Copyright © 2019, 2025, Oracle and/or its affiliates.

ocnrf_nfDiscover_emptyList_total metric is added for the EmptyList feature.

Alert
OcnrfNFDiscoveryEmptyListObservedNotification alert is added for the EmptyList feature.

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.38 Overload Control

Overload occurs when 100% of the planned capacity is exhausted. It can be due to uneven distribution of traffic towards a given NRF service instance, network fluctuations leading to traffic bursts, or unexpected high traffic volume at any given point of time.

The Overload Control feature protects the system from getting overloaded due to unexpected surges in traffic, traffic failures, or high latency and maintains the overall health of NRF. The system needs to not only detect overload conditions but also protect against the same. Further, it needs to mitigate against and avoid the system from entering into an overload condition by taking necessary actions for recovering from overload. Using Ingress Gateway and Perf-Info microservices, NRF provides options to control network congestion and handle the overload scenarios.

NRF provides the following means for overload management:

- It provides a predefined set of threshold load levels.
- It monitors various overload indicators such as CPU utilization, pending message count, and failure message count.
- In case the threshold load level is breached for any overload indicator, enforce load shedding to protect the service.

The overload indicator for each service is monitored by the Overload Manager module in Perf-Info service. It performs overload calculations based on the following indicators:

- **CPU Utilization**: Perf-Info monitors the CPU utilization of each service by querying Prometheus for the metric '`cgroup_cpu_nanoseconds`'.
- **Pending Message Count:** Ingress Gateway microservice monitors and maintains the number of messages pending to be processed by Ingress Gateway for each service. Perf-Info queries the Ingress Gateway microservice periodically to get the pending message count.
- **Failure Message Count**: Ingress Gateway microservice monitors the number of requests returning failure responses for each service. Perf-Info queries Ingress Gateway microservice periodically to get the failure message count.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> NRF has deprecated memory-based overload control as it is already covered by pending message count overload control.

The overload level is configured for the following NRF microservices:

- Registration
- Discovery
- AccessToken

Overload control is triggered if the threshold for one or more indicator is reached.

Perf-Info calculates the overload levels based on the values reported for each overload indicator and the configured thresholds for the overload indicator. If one or more overload indicators have breached their thresholds, the highest overload level is considered.
For example, if the CPU utilization of the discovery microservice has crossed L2 threshold, and its pending message count has crossed L3 threshold, then the overload level is L3. The overload level is communicated to Ingress Gateway periodically to enforce load shedding. A configurable value is available for sampling interval as `ocPolicyMapping.samplingPeriod` based on which Ingress Gateway calculates the rate per service in the current sampling period and applies appropriate discard policies and actions in the subsequent sampling period.

The Ingress Gateway Overload Controller monitors all the service requests received against the overload control configuration. If the feature is enabled, Ingress Gateway computes the percentage of the service requests that should be discarded by using the rate of incoming requests per service for a configurable sampling period.

When Perf-Info reports that the service is in an overload state, Ingress Gateway matches the service requests to the configured service name and the configured discard policy for a particular service, and takes the appropriate action.

> **Note**
>
> When the percentage-based overload control discard policy is enabled, the number of requests to be dropped in the current sampling period is computed based on the configured percentage of requests that should be discarded and the "rate of requests outgoing of Ingress Gateway" in the previous sampling period for the service. Once the number of requests to be dropped in the current sampling period is computed, the Ingress Gateway does not drop all the new traffic to meet the discard count. Instead, Ingress Gateway decides if a request is to be discarded or not in a random way. If the random function returns the value as *true*, the request is discarded in the current sampling period with the discard action "`RejectWithErrorCode`". In a sampling period, the requests are discarded in a distributed way. The percentage dropped is not precisely the configured percentage because the number of requests to be dropped in the current sampling period is calculated based on the number of requests sent to the NRF discovery microservice in the previous sampling period and not based on the total requests received at Ingress Gateway.

Pod Level Traffic Rejections

Currently, NRF uses percentage-based discard for overload control. NRF rejects the traffic (configured percentage) with an error code when overload thresholds are breached.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Depending on the number of requests to be rejected, a token is fetched from a cache shared across all Ingress Gateway pods.

This approach has certain accuracy limitations, particularly in environments with:

- Higher number of minimum token request values
- Large number of Ingress Gateway pods
- Uneven traffic distribution across Ingress Gateway pods
- Low Transactions Per Second (TPS) traffic pattern

From Release 25.1.200 onwards, NRF rejects the incoming requests at pod level for percentage-based overload control by removing the dependency on cache-based coordination across pods. When the overload control level is breached, the number of requests to be rejected is calculated based on the requests received at each Ingress Gateway pod. This ensures a more accurate and consistent request rejection even in scenarios with low Transactions Per Second (TPS) and uneven traffic distribution.

Managing Overload Control Feature

Enable

You can enable Overload Control and Pod Level Traffic Rejections feature using the following Helm configuration:

1. Open the `ocnrf_custom_values_25.1.201.yaml` file.
2. Check the value of `global.performanceServiceEnable` parameter to `true` in `ocnrf_custom_values_25.1.201.yaml` file.
3. Check the value of `perf-info.overloadManager.enabled` parameter to `true` in `ocnrf_custom_values_25.1.201.yaml` file.
4. Check the value of `ingressgateway.overloadControlLocalDiscardEnable` parameter is `true` in `ocnrf_custom_values_25.1.201.yaml` file for the pod level traffic rejections.
5. Configure the Prometheus URI in `perf-info.configmapPerformance.prometheus` in `ocnrf_custom_values_25.1.201.yaml` file.
6. Save the `ocnrf_custom_values_25.1.201.yaml` file.
7. Ensure that the `autoscaling` and `apps` apiGroups are configured under service account. For more information on configuration, see "Creating Service Account, Role and Role Binding Resources" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
8. Run `helm upgrade`, if you are enabling this feature after NRF deployment. For more information on upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Configure

- **Configure using REST API**: The Overload Control feature related configurations are performed at Ingress Gateway and Perf-Info.
  The following REST APIs must be configured for this feature:
  - `{apiRoot}/nrf/nf-common-component/v1/perfinfo/overloadLevelThreshold`
  - `{apiRoot}/nrf/nf-common-component/v1/igw/errorcodeprofiles`
  - `{apiRoot}/nrf/nf-common-component/v1/igw/ocdiscardpolicies`
  - `{apiRoot}/nrf/nf-common-component/v1/igw/ocpolicymapping`
  - `{apiRoot}/nrf/nf-common-component/v1/igw/errorcodeserieslist`

Copyright © 2019, 2025, Oracle and/or its affiliates.

- `{apiRoot}/nrf/nf-common-component/v1/igw/routesconfiguration`

For more information about APIs, see "Common Services REST APIs" section in the *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*

- **Configure using CNC Console**: Perform the following configurations:
  - Create or update Error Code Profile.
  - Create or update the Overload Control Discard Policy.
  - Create or update Overload Control Policy Mapping.
  - Create or update Error Code Series.
  - Update the Routes Configuration as mentioned above.

Disable

1. Use `{apiRoot}/nrf/nf-common-component/v1/igw/ocpolicymapping` API and set `enabled` to **false** as follows:

```
{
    "enabled": false,
    "mappings": [
        {
            "svcName": "ocnrf-nfdiscovery",
            "policyName": "nfdiscoveryPolicy"
        },
        {
            "svcName": "ocnrf-nfaccesstoken",
            "policyName": "nfaccesstokenPolicy"
        },
        {
            "svcName": "ocnrf-nfregistration",
            "policyName": "nfregistrationPolicy"
        }
    ],
    "samplingPeriod": 200
}
```

2. Open the `ocnrf_custom_values_25.1.201.yaml` file.
3. Set `perf-info.overloadManager.enabled` parameter to `false` in `ocnrf_custom_values_25.1.201.yaml` file.
4. (Optional) Set `global.performanceServiceEnable` parameter to `false` in `ocnrf_custom_values_25.1.201.yaml` file.

> **Note**
>
> Perform this step only if you want to disable complete Perf-Info service.

5. Save the `ocnrf_custom_values_25.1.201.yaml` file.
6. Run `helm upgrade`, if you are disabling this feature after NRF deployment. For more information on upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Observe

Metrics

The following metrics are added for the Overload Control feature:

- `load_level`
- `service_resource_stress`
- `service_resource_overload_level`

For more information about the metric, see Overload Control.

Alert

The following alerts are added for the Overload Control feature:

- OcnrfOverloadThresholdBreachedL1
- OcnrfOverloadThresholdBreachedL2
- OcnrfOverloadThresholdBreachedL3
- OcnrfOverloadThresholdBreachedL4

KPIs

There are no KPIs for this feature.

For more information on the entire list of KPIs, see NRF KPIs.

Identifying Kubernetes Tag for Overload Control

The following procedure explains how to identify the tags for different Kubernetes version:

1. Log in to Prometheus.
2. Enter `cgroup_cpu_nanoseconds` query in Prometheus as shown:

Figure 4-23 Prometheus Search

3. In the response, search for the tags that contain the values for name of the container, NRF deployment namespace, and name of the NRF services:

Figure 4-24 Tag Search

4. Use the tags from the response to configure the following parameters under Perf-Info microservice:
   - `tagNamespace`
   - `tagContainerName`
   - `tagServiceName`

   For example, in the Figure 4-24 image, `namespace` is the tag that contain the value for NRF deployment namespace. You need to set `namespace` as the value for `tagNamespace`

Copyright © 2019, 2025, Oracle and/or its affiliates.

parameter in Perf-Info microservice. For more information on the parameters, see Perf-Info section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide.*

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

4.39 DNS NAPTR Update

In the 5G core networks, AMFs are added or removed dynamically for scalability or planned maintenance. NRF supports updating the Name Authority Pointer (NAPTR) record in Domain Name System (DNS) during Access and Mobility Functions (AMF) registration, update, and deregistration. The AMFs available within an AMF set is provisioned within NAPTR records in the DNS.

> **Note**
>
> - NRF considers the priority of the peer only for peer selection.
> - This feature can be used in a NRF deployment where service-mesh is NOT present.

Enable

You must enable Artisan microservice for DNS NAPTR Update as follows:

1. Open the `ocnrf_custom_values_25.1.201.yaml` file.
2. Set `enableNrfArtisanService` to **true** to enable Artisan microservice.
3. Save the file.
4. Run Helm install. For more information about installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
5. If you are enabling this parameter after NRF deployment, run helm upgrade. For more information about upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
6. Set the DNS configuration as described in the "DNS NAPTR Configuration in Alternate Route Service" section of *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
7. After enabling Artisan microservice, enable DNS NAPTR Update Options using the REST API or CNC Console:

Copyright © 2019, 2025, Oracle and/or its affiliates.

- **Enable using REST API**: Set `featureStatus` to `ENABLED` in dnsNAPTRUpdateOptions configuration API. For more information about API path, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Set **Feature Status** to **ENABLED** on the DNS NAPTR Update Options page. For more information about enabling the feature using CNC Console, see DNS NAPTR Update Options Options.

Configure

You can configure DNS NAPTR Update options feature using the REST API or CNC Console.

- **Configure using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Perform the feature configurations as described in DNS NAPTR Update Options Options.

Observe

Metrics

Following metrics are added in the DNS NAPTR Update Metrics section:

- ocnrf_dns_naptr_tx_requests_total
- ocnrf_dns_naptr_rx_responses_total
- ocnrf_dns_naptr_audit_tx_requests_total
- ocnrf_dns_naptr_audit_rx_responses_total
- ocnrf_dns_naptr_failure_rx_responses
- ocnrf_dns_naptr_round_trip_time_seconds
- ocnrf_dns_naptr_nfRegistration_tx_requests_total
- ocnrf_dns_naptr_nfRegistration_rx_responses_total
- ocnrf_dns_naptr_nrfAuditor_tx_requests_total
- ocnrf_dns_naptr_nrfAuditor_rx_responses_total
- ocnrf_dns_naptr_trigger_rx_requests_total
- ocnrf_dns_naptr_trigger_tx_responses_total
- oc_alternate_route_upstream_dns_request_timeout_total

Alert

Following alerts are added for the DNS NAPTR Update feature:

- OcnrfDnsNaptrFailureResponseStatus
- OcnrfAlternateRouteUpstreamDnsRetryExhausted

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information about how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

2. **Raise a service request**: See My Oracle Support for more information about how to raise a service request.

4.40 Notification Retry

NRF supports notification retry mechanism in case of the following scenarios in sending `NfStatusNotify` message. Based on the configured Notification Retry Profile, the NRF attempts to re-send the `NfStatusNotify` message on failures until the retry attempt is exhausted.

Following are the scenarios where retry can be performed:

- 4xx, 5xx Response Codes from notification callback server of the NF
- Connection Timeout between NRF Egress Gateway and notification callback server of the NF
- Request Timeout at NRF Egress Gateway

To enable notification request retry, the NRF Subscription microservice sends the `NfStatusNotify` message to the Egress Gateway. Upon failure of the request, Egress Gateway retries towards the notification callback server of the NF depending on the configuration. The retries will happen depending upon the response codes, exception list or timeout configuration in the `nfManagementOptions`.

> **Note**
>
> This is applicable only when direct routing is used for routing notification messages.

Managing Notification Retry Feature

Enable
You can enable the Notification Retry feature using the CNC Console or REST API.

- **Enable using REST API**: Set `requestRetryDetails.featureStatus` to **ENABLED** in NF Management Options configuration API. For more information about API path, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Set **Feature Status** to **ENABLED** under **Notification Retry** section on the NF Management Options page. For more information about enabling the feature using CNC Console, see the NF Management Options section.

Configuring DefaultRouteRetry in Egress Gateway

After NRF upgrade from 23.3.x to 23.4.0, to enable notification retry feature, `DefaultRouteRetry` filter must be added to the Egress Gateway default route configuration.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Note

- The value of `nfsubscription.jetty.request.idleTimeout` parameter must be greater than total request timeout value (`totalRequestTimeout = ((retryCount+1) * requestTimeout) + 1000`) if notification retry feature is enabled. For more information about the parameter, see the "NF Subscription Microservice (`nfsubscription`)" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
- If attempts are done due to request timeouts, then it is possible that the backend services may timeout before all the attempts are completed. In order to avoid this, the backend requestTimeout attributes must be configured considering the total number of Egress reroutes and the Egress Gateway requestTimeout. The discovery requestTimeout values are configured using `nfdiscovery.jetty.request.timeout` parameter.

Following is the sample command to configure `DefaultRouteRetry`:

```
curl -X PUT "http://ocnrf-nrfconfiguration:8080/nrf/nf-common-
component/v1/egw/routesconfiguration" -H 'Content-Type:application/json' -d
'[{
    "id": "default_route",
    "uri": "egress://request.uri",
    "order": 100,
    "filters": [{
        "name": "DefaultRouteRetry"
    }],
    "predicates": [{
        "args": {
            "pattern": "/**"
        },
        "name": "Path"
    }]
}]'
```

This updates the hostname and port of the `ocnrf-nrfconfiguration` and reflects the details to access the NRF configuration.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> To disable the notification retry, remove the `DefaultRouteRetry` filter from the `{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration`, and retain the `default_route` configuration. This `default_route` configuration is available by default during fresh installation or upgrade. Following is the sample default route configuration:
>
> ```
> {
>     "id": "default_route",
>     "uri": "egress://request.uri",
>     "order": 100,
>     "predicates": [{
>         "args": {
>             "pattern": "/**"
>         },
>         "name": "Path"
>     }]
> }
> ```

Configure

You can configure Notification Retry feature using the REST API or CNC Console.

- **Configure using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Perform the feature configurations as described in the NF Management Options section.

Observe

Metrics

Following dimensions are added for `ocnrf_nfStatusNotify_rx_responses_total` metric in the NRF NF Metrics section:

- NotificationHostPort
- NumberOfRetriesAttempted

Alert

Following alerts are added for the Notification Retry feature:

- OcnrfNotificationRetryExhausted
- OcnrfNotificationFailureOtherThanRetryExhausted

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

2. **Raise a service request**: See My Oracle Support for more information about how to raise a service request.

4.41 NRF Message Feed

NRF supports copying of both request and response HTTP messages routed through Ingress and Egress Gateways, to a Data Director.

Data Director receives the messages from Gateways with a correlation-id and feeds the data securely to an external monitoring system.

The correlation-id in the messages is used as a unique identifier for every transaction. If the request does not have the correlation-id, the Gateway generates the correlation-id and then passes it to Data Director.

Upon receiving an Ingress Gateway requests containing a correlation-Id, the NRF microservices includes this correlation-Id in following Egress Gateway requests that may be generated by the NRF based on the Ingress Gateway transaction:

- Inter-PLMN
- SLF
- Forwarding
- Notification

The communication between the Gateways and the Data Director is encrypted using TLS. Also, the Gateways authenticate itself to Data Director using Simple Authentication and Security Layer (SASL). For more information on configuring SASL, see Configuring SASL.

> **Note**
>
> - NRF does not copy the following messages to Data Director:
>   - NRF Discovery for SLF (Artisan to Discovery query for filling SLF Candidate List)
>   - NRF Getting AccessToken for SLF (Egress Gateway Microservice to AccessToken for fetching Token for SLF request)
> - This feature can be used in a NRF deployment where service-mesh is NOT present.

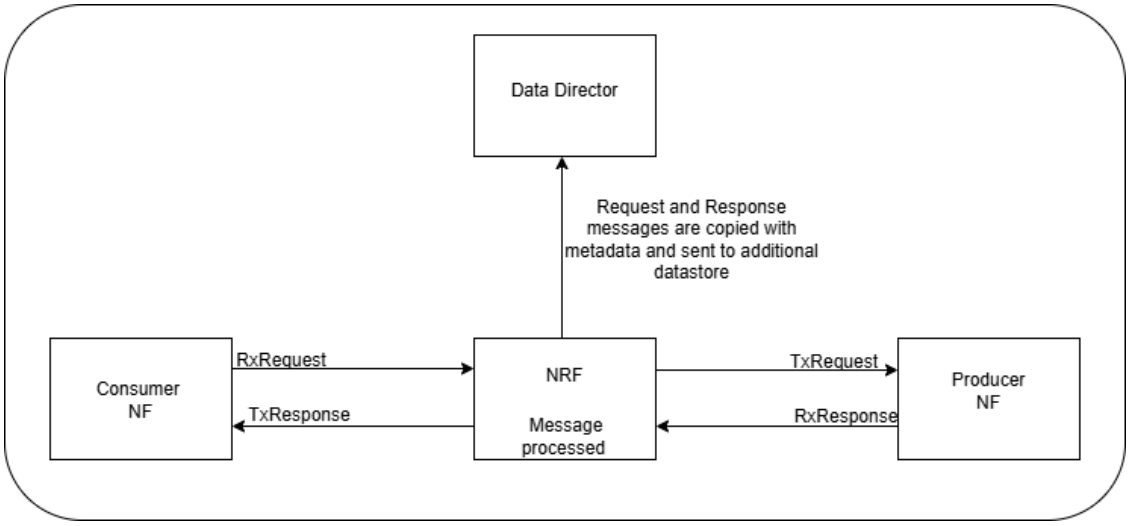
Additional Metadata Sent from NRF to Data Director

From release 25.1.200 onwards, along with the existing attributes, the following additional attributes about the message are also copied as part of the metadata list:

- `source-ip`
- `destination-ip`
- `source-port`
- `destination-port`
- `pod-instance-id`

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-25 Sending Metadata from NRF to Data Director



Request Messages:

- **RxRequest:** NRF receives a request sent by an external entity, such request is handled first by the Ingress Gateway microservice, it creates a message with additional metadata which is communicated to Data Director and the request is forwarded to a specific NRF backend microservice.
- **TxRequest:** NRF transmits a request to communicate an external entity, such request is sent by NRF backend microservice to Egress Gateway microservice, it creates a message with additional metadata which is communicated to Data Director and then forwards such requests to the specific address.

Response Messages:

- **RxResponse:** NRF receives a response from an external entity, such response is handled by Egress Gateway microservice, it creates a message with additional metadata which communicated to Data Director and then it forwards the response.
- **TxResponse:** NRF transmits a response to an external entity, such response is sent by an NRF microservice to Ingress Gateway microservice, it creates a message with additional metadata which is communicated to Data Director and then sends back the response to the external entity which initiated the communication.

The following table defines the attributes of the message copy feed template:

Table 4-25 OracleNfFeed

Attribute	Data Type	Description
`version`	String	NRF Producer data model version.
`metadata-list`	MetaData	The metadata added by NRF is used by Data Director for filtering, aggregation, and routing.
`header-list`	Http2Header	The http2 message header.
`5g-sbi-message`	Http2Body	The http2 message body. Supported formats: • JSON • gzip data (content-encoding:gzip)

Copyright © 2019, 2025, Oracle and/or its affiliates.

The following are the descriptions of the attributes that are copied to the Data Director as part of metadata:

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-26 Attributes Copied to Data Director

Attribute name	Data type	Description	Source of data
correlation-id	String	This is a mandatory parameter. This is a unique identifier for the message (both request and response). The possibilities to populate this field in the metadata is as follows: • If the x-request-id header is present in the message, then the correlation-id will be filled with the x-request-id header value. • If this header is not present, then the following processing is done: – At Ingress Gateway Microservice: As soon as the message is received and the x-request-id header is not present, then Ingress Gateway generates a UUID, and set correlation-id to this UUID. Copy the message towards Data Director with correlation-id in metadata, but no x-request-id header. After copying the message to Data Director, this x-request-id header is added by Ingress Gateway to the incoming request. For response coming from the NRF backend microservice, Ingress Gateway microservice again picks the same UUID as set in the request from its context and set in the metadata. For response, it does not matter if the x-request-id header is present or not, as Ingress Gateway microservice already has it in its request context which is re-used while processing the corresponding response. – At Egress Gateway Microservice: After receiving a request from the NRFbackend microservice, if the x-request-id header is not present, Egress Gateway	x-request-id header or autogenerate.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-26 (Cont.) Attributes Copied to Data Director

Attribute name	Data type	Description	Source of data
		microservice first generates a UUID and set correlation-id metadata to this UUID, and also add the x-request-id header to the request message. Then it should copy the message to Data Director and then send it out on the wire. In the response received from outside NF, Egress Gateway microservice reuses the UUID it generated during the request and set it to correlation-id of response metadata. Egress Gateway microservice need not check or add the x-request-id header in the response.	
`consumer-id`	String	This is an optional parameter. 5G NF Instance ID of the NF originated the received message. For Ingress Gateway microservice, this is peer-nf-id For Egress Gateway microservice, this is self-nf-id For Notifications at Egress Gateway, this is peer-nf-id	peer-nf-id for Ingress Gateway is the NF-Instance ID extracted from the user-agent header if available in the request. User-Agent: <NF Type>-<NF Instance ID> <NF FQDN> self-nf-id will be extracted from Helm configurations (nfInstanceId).
`producer-id`	String	This is an optional parameter. 5G NF Instance ID of the next destination NF.	self-nf-id will be extracted from Helm configurations (nfInstanceId). **Note**: Egress Gateway doesn't populate producer-id.
`consumer-fqdn`	String	This is an optional parameter. It is the FQDN of the orginating NF. For Ingress Gateway, this is peer-nf-fqdn. For Egress Gateway, this is self-nf-fqdn. For Notifications at Egress Gateway, this is peer-nf-fqdn.	peer-nf-fqdn for Ingress Gateway is the NF-FQDN extracted from user-agent header. User-Agent: <NF Type>-<NF Instance ID> <NF FQDN> self-nf-fqdn will be extracted from Helm configurations (nfFqdn). peer-nf-fqdn for Egress Gateway is extracted from 3gpp-sbi-target-apiroot header.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-26 (Cont.) Attributes Copied to Data Director

Attribute name	Data type	Description	Source of data
`producer-fqdn`	String	This is an optional parameter. FQDN of the destination NF For Egress Gateway, this is peer-nf-fqdn For Ingress Gateway, this is self-nf-fqdn For Notifications at Egress Gateway, this is self-nf-fqdn	peer-nf-fqdn is extracted from3gpp-sbi-target-apiroot header. self-nf-fqdn will be extracted from Helm configurations (nfFqdn).
`source-ip`	String	This is an optional parameter. The IP address of the source NF that has sent the message.	NA
`destination-ip`	String	This is an optional parameter. The IP address of the target NF to which the message has to be sent.	NA
`source-port`	String	This is an optional parameter. The port of the source NF that receives response from the target NF. TCP port of the NF on the source IP address.	NA
`destination-port`	String	This is an optional parameter. The port of the target NF that sends response to the source NF. TCP port of the NF on the destination IP address.	NA
`timestamp`	String	This is a mandatory parameter. Identifies the timestamp when the message arrives at the producer. The timestamp format would be long and in nanoseconds.	The time at which the message is created with nanoseconds precision. Example: 1656499195571109210 nanoseconds
`message-direction`	String	This is a mandatory parameter. The direction of the message. This can be either incoming or outgoing.	Parameter to indicate whether a message is an ingress to or egress from NF. It can be indicated by putting the traffic feed trigger point name. • RxRequest (Ingress Request) • TxRequest (Egress Request) • RxResponse (Egress Response) • TxResponse (Ingress Response)
`feed-source`	FeedSource	This is a mandatory parameter. Information about the NF details. **Note:** The details are given in the following table Table 4-27.	The details are given in the following table Table 4-27.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 4-27 FeedSource

Attribute name	Data type	Description	Source of data
nf-type	String	This is a mandatory parameter. Identifies a type of producer NF.	The information from the Helm configuration from the parameter global.nfTypeMsgCpy. If the NFs haven't configured this, then unknown would be attached as a value.
nf-instance-id	String	This is an optional Parameter. Identifies a producer NF Instance.	The information from Helm configuration from the parameter global.nfTypeMsgCpy.
nf-fqdn	String	This is an optional Parameter. Identifies a producer NF FQDN.	The information from Helm configuration from the parameter global.nfFqdn.
pod-instance-id	String	This is an optional parameter. Identifies a producer NF's pod. This is the unique identifier or name assigned to the pod.	NA

Sample JSON file with message feed details for TxResponse transaction:


```
{
    "version": "1.0.0",
    "header-list": {
        ":status": "201",
        "date": "Tue, 22 Apr 2025 01:56:52 GMT",
        "location": "http://nrf-18199363-ingressgateway:80/nnrf-nfm/v1/nf-
instances/17380efd-6b77-4c9a-8ed9-f820371d842b",
        "content-type": "application/json",
        "nettylatency": "1745287012718",
        "requestmethod": "PUT"
    },
    "metadata-list": {
        "correlation-id": "7967c76c-15fb-4c3c-b324-c9f408698930",
        "producer-id": "6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c",
        "producer-fqdn": "NRF-d5g.oracle.com",
        "timestamp": "1745287012820818431",
        "message-direction": "TxResponse",
        "source-ip": "xxx.xxx.xxx.xxx",
        "source-port": "8081",
        "destination-ip": "xxx.xxx.xxx.xxx",
        "destination-port": "51244",
        "feed-source": {
            "nf-type": "NRF",
            "nf-instance-id": "6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c",
            "nf-fqdn": "NRF-d5g.oracle.com",
            "pod-instance-id": "nrf-18199363-ingressgateway-55bdcb779c-clvwg"
        }
    },
    "5g-sbi-message": {
        "nfInstanceId": "17380efd-6b77-4c9a-8ed9-f820371d842b",
        "nfType": "AMF",
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
"nfStatus": "REGISTERED",
"heartBeatTimer": 30,
"plmnList": [
    {
        "mcc": "310",
        "mnc": "14"
    }
],
"sNssais": [
    {
        "sd": "4ebaaa",
        "sst": 124
    },
    {
        "sd": "dc8aaa",
        "sst": 54
    },
    {
        "sd": "f46aaa",
        "sst": 73
    }
],
"nsiList": [
    "slice-1",
    "slice-2"
],
"fqdn": "AMF.d5g.oracle.com",
"interPlmnFqdn": "AMF-d5g.oracle.com",
"ipv4Addresses": [
    "xxx.xxx.xxx.xxx",
    "xxx.xxx.xxx.xxx",
    "xxx.xxx.xxx.xxx",

    "xxx.xxx.xxx.xxx"
],
"ipv6Addresses": [
    "2001:0db8:85a3:0000:0000:8a2e:0370:7334"
],
"capacity": 2000,
"load": 10,
"locality": "US East",
"amfInfo": {
    "amfSetId": "1ab",
    "amfRegionId": "23",
    "guamiList": [
        {
            "plmnId": {
                "mcc": "594",
                "mnc": "75"
            },
            "amfId": "947d18"
        },
        {
            "plmnId": {
                "mcc": "602",
                "mnc": "42"
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
                        },
                        "amfId": "5f3259"
                    },
                    {
                        "plmnId": {
                            "mcc": "135",
                            "mnc": "19"
                        },
                        "amfId": "f37f1a"
                    },
                    {
                        "plmnId": {
                            "mcc": "817",
                            "mnc": "80"
                        },
                        "amfId": "f7a026"
                    }
                ],
                "taiList": [
                    {
                        "plmnId": {
                            "mcc": "641",
                            "mnc": "72"
                        },
                        "tac": "ccc0"
                    },
                    {
                        "plmnId": {
                            "mcc": "594",
                            "mnc": "45"
                        },
                        "tac": "77db"
                    }
                ]
            },
            "nfServices": [
                {
                    "serviceInstanceId": "fe137ab7-740a-46ee-aa5c-951806d77b0d",
                    "serviceName": "namf-mt",
                    "versions": [
                        {
                            "apiVersionInUri": "v1",
                            "apiFullVersion": "1.0.0",
                            "expiry": "2018-12-03T18:55:08.871Z"
                        }
                    ],
                    "scheme": "http",
                    "nfServiceStatus": "REGISTERED",
                    "fqdn": "AMF.d5g.oracle.com",
                    "interPlmnFqdn": "AMF-d5g.oracle.com",
                    "apiPrefix": "",
                    "defaultNotificationSubscriptions": [
                        {
                            "notificationType": "LOCATION_NOTIFICATION",
                            "callbackUri": "http://somehost.oracle.com/callback-
uri"
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
                        },
                        {
                            "notificationType": "N1_MESSAGES",
                            "callbackUri": "http://somehost.oracle.com/callback-
uri",
                            "n1MessageClass": "SM"
                        },
                        {
                            "notificationType": "N2_INFORMATION",
                            "callbackUri": "http://somehost.oracle.com/callback-
uri",
                            "n2InformationClass": "NRPPa"
                        }
                    ],
                    "allowedPlmns": [
                        {
                            "mcc": "904",
                            "mnc": "47"
                        },
                        {
                            "mcc": "743",
                            "mnc": "47"
                        },
                        {
                            "mcc": "222",
                            "mnc": "23"
                        },
                        {
                            "mcc": "521",
                            "mnc": "11"
                        }
                    ],
                    "allowedNfTypes": [
                        "NRF",
                        "AMF",
                        "AUSF",
                        "BSF",
                        "UDM",
                        "UDR",
                        "PCF"
                    ],
                    "allowedNfDomains": [
                        "oracle.com",
                        "xyz.com"
                    ],
                    "allowedNssais": [
                        {
                            "sd": "dbaaaa",
                            "sst": 14
                        },
                        {
                            "sd": "3caaa9",
                            "sst": 153
                        },
                        {
                            "sd": "5faaad",
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
                        "sst": 132
                    }
                ],
                "capacity": 500,
                "load": 0,
                "supportedFeatures": "80000000"
            }
        ]
    }
},
```

Managing Data

The messages are managed according to the type of content.

The following table describes the behavior of different content type:

Table 4-28 Managing Data

Content-Type	Content-Encoding	Behavior
• application/json • application/problem+json • application/json-patch+json • application/3gppHal+json	NA	The payload is converted into a JSON object and attached to the message copy feed payload as a JSON node.

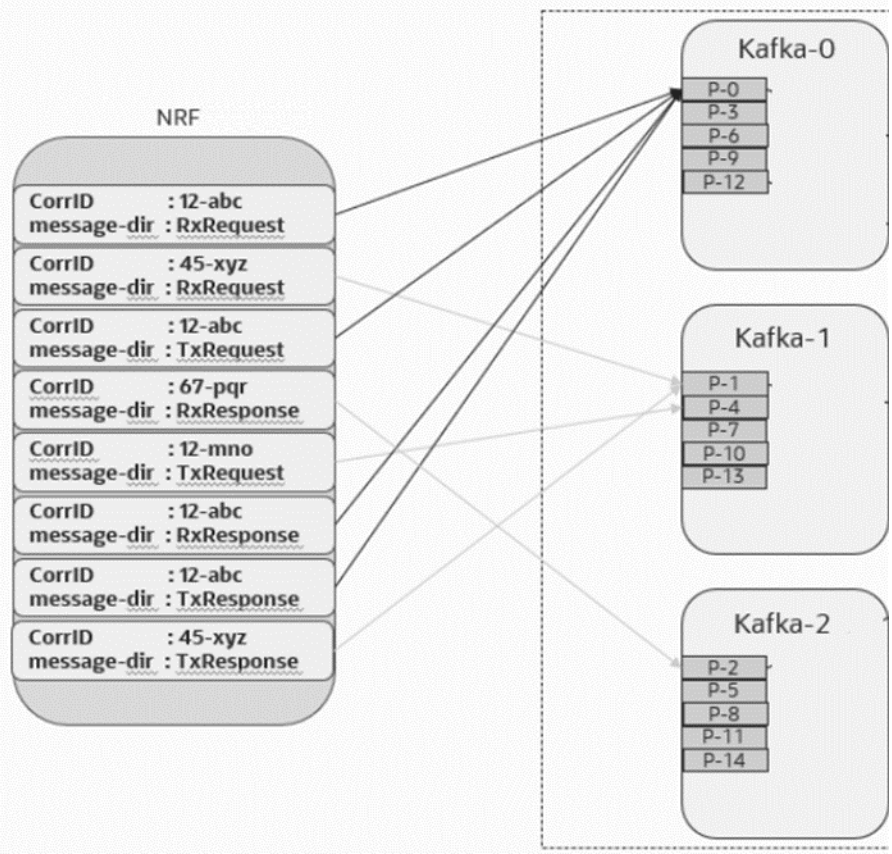
Writing Messages of the Same Transaction in the Same Kafka Partition

From release 25.1.200 onwards, while sending messages to Data Director, NRF supports copying the request and response messages of same transaction to the same Kafka partition. This reduces the latency used to process the same transaction data. This feature uses correlation-id (unique-identifier) metadata as message key for the correlation of the messages for a transaction.

The following diagram represents the enhancement of selecting the same Kafka partition for the messages of the same transaction type:

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 4-26 Same Kafka Partition for Writing the Messages of the Same Transaction Type



Data Director uses the corelation-id received from the Gateways to group the messages belonging to the same transaction. Data Director feeds all these messages to the same Kafka partition instead of multiple partitions.

Managing the feature

Managing NRF Message Feed Feature

Enable

You can enable Message Feed and the feature using the Helm:

1. Open the `ocnrf_custom_values_25.1.201.yaml` file.
2. Set `ingressgateway.messageCopy.enabled` to **true** to enable message copying at the Ingress Gateway. For more information on enabling the parameter, see the "Ingress Gateway Parameters" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
3. Set `egressgateway.messageCopy.enabled` to **true** to enable message copying at the Egress Gateway. For more information on enabling the parameter, see the "Egress Gateway Parameters" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
4. Set `ingressgateway.messageCopy.keybasedKafkaProducer` to **true** to enable copying messages of the given transaction to the same Kafka partition in Ingress Gateway. For more information on enabling the parameter, see the "Ingress Gateway Parameters"

Copyright © 2019, 2025, Oracle and/or its affiliates.

section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

5. Set `egressgateway.messageCopy.keybasedKafkaProducer` to **true** to enable copying messages of the given transaction to the same Kafka partition in Egress Gateway. For more information on enabling the parameter, see the "Egress Gateway Parameters" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
6. Save the file.
7. Install NRF. For more information about installation procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
8. If you are enabling this parameter after NRF deployment, upgrade NRF. For more information about upgrade procedure, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

> **Note**
>
> Enabling the Message Feed feature on both the Ingress and Egress Gateways may affect pod capacity, depending on your usage model and configuration.

Configure

There is no configuration to be performed using the REST API or CNC Console.

Configuring SASL

1. Generate your SSL Certificates.

> **Note**
>
> Creation process for private keys, certificates and passwords is based on discretion of user or operator.

2. Before copying the certificates into the secret, add the DD Root certificates contents into the CA certificate (**caroot.cer**) generated for NRF as follows:

> **Note**
>
> Make sure you have added 8 hyphens "**-**"between 2 certificates.
>
> ```
> -----BEGIN CERTIFICATE-----
> <existing caroot-certificate content>
> -----END CERTIFICATE-----
> --------
> -----BEGIN CERTIFICATE-----
> <DD caroot-certificate content>
> -----END CERTIFICATE-----
> ```


Copyright © 2019, 2025, Oracle and/or its affiliates.

3. Create secrets for both Ingress and Egress Gateway for authentication with Data Director. To create a secret you need to store the password in a text file and using same file need to create a new secret. Run the following command to create secret:

   ```
   kubectl create secret generic ocingress-secret --from-
   file=ssl_ecdsa_private_key.pem --from-file=rsa_private_key_pkcs1.pem --
   from-file=ssl_truststore.txt --from-file=ssl_keystore.txt --from-
   file=caroot.cer --from-file=ssl_rsa_certificate.crt --from-
   file=ssl_ecdsa_certificate.crt --from-file=sasl.txt -n <namespace>

   kubectl create secret generic ocegress-secret --from-
   file=ssl_ecdsa_private_key.pem --from-file=ssl_rsa_private_key.pem --from-
   file=ssl_truststore.txt --from-file=ssl_keystore.txt --from-
   file=ssl_cabundle.crt --from-file=ssl_rsa_certificate.crt --from-
   file=ssl_ecdsa_certificate.crt --from-file=sasl.txt -n <namespace>
   ```

   For more information on creating secrets, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
4. Configure the SSL section as described in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
5. Configure the message copy feature as described in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
6. Configure the **SASL_SSL** port in the **kafka.bootstrapAddress** attribute.

Observe

Metrics

Following metrics are added in the NRF Gateways Metrics section:

- `oc_ingressgateway_msgcopy_requests_total`
- `oc_ingressgateway_msgcopy_responses_total`
- `oc_ingressgateway_dd_unreachable`
- `oc_egressgateway_msgcopy_requests_total`
- `oc_egressgateway_msgcopy_responses_total`
- `oc_egressgateway_dd_unreachable`

Alert

Following alerts are added in the NRF Alerts section:

- OcnrfIngressGatewayDDUnreachable
- OcnrfEgressGatewayDDUnreachable

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

Copyright © 2019, 2025, Oracle and/or its affiliates.

4.42 Subscription Limit

In 5G architecture, Network Functions (NFs) subscribe to be notified for a change in the producer NF Profile in a network using the NFStatusSubscribe service Operation. This subscription is managed by NRF Management microservice and maintained in the database for a particular period. A notification is triggered when there is change in the producer NF Profile to the Consumer NFs by the NRF subscription microservice.

If the number of subscriptions created is very high, it may trigger a huge number of notification requests. This might lead to the overload of the NRF subscription microservice.

NRF restricts the number of allowed subscriptions to avoid overload condition at NRF subscription microservice. NRF regulates the maximum number of allowed subscriptions in the database using a configurable Global Subscription Limit. In case of georedundant NRF, the limit is applied across all the mated sites.

> **Note**
>
> The subscription limit must be configured with the same across all georedundant site.

NRF evaluates the global subscription limit condition every time an NF tries to create a new subscription or update an existing subscription. If the limit is breached, new subscriptions are not created or renewed, and the subscription requests are rejected with configured error code. Upon the global subscription limit is breached as well as approaching the breach threshold, specific alerts are raised based on the threshold level.

Managing NRF Subscription Limit Feature

Prerequisites

Following are the prerequisites to enable the feature:

1. All the georedundant sites must be upgraded to NRF 25.1.201 release.
2. Replication link between all the georedundant sites must be up. The `OcnrfDbReplicationStatusInactive` alert indicates if the replication link is inactive. If this alert is raised in any of the sites, wait till it is cleared.
3. Wait until all the subscription records are migrated. The OcnrfSubscriptionMigrationInProgressWarn and OcnrfSubscriptionMigrationInProgressCritical alerts indicate whether the migration is complete. If this alert is raised in any of the sites, wait till it is cleared.

Enable

You can enable the Subscription Limit feature using the CNC Console or REST API.

- **Enable using REST API**: Set `subscriptionLimit.featureStatus` to **ENABLED** in NF Management Options configuration API. For more information about API path, see *Oracle Communications Cloud Native Core Network Repository Function REST Specification Guide*.
- **Enable using CNC Console**: Set **Feature Status** to **ENABLED** under **Subscription Limit** section on the NF Management Options page. For more information about enabling the feature using CNC Console, see the NF Management Options section.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Configure

You can configure Subscription Limit feature using the REST API or CNC Console.

- **Configure using REST API**: Perform the feature configurations as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*
- **Enable using CNC Console**: Perform the feature configurations as described in the NF Management Options section.

Observe

Metrics

Following dimension is added in the NRF Metrics section:

- `RejectionReason=SubscriptionLimitExceeded` dimension is added for `ocnrf_nfStatusSubscribe_tx_responses_total` metric.

Following metrics are added for the Subscription Limit feature:

- ocnrf_nfset_active_subscriptions
- ocnrf_nfset_limit_level
- ocnrf_subscription_migration_status

Alerts

Following alerts are added for the Subscription Limit feature:

- OcnrfSubscriptionGlobalCountWarnThresholdBreached
- OcnrfSubscriptionGlobalCountMinorThresholdBreached
- OcnrfSubscriptionGlobalCountMajorThresholdBreached
- OcnrfSubscriptionGlobalCountCriticalThresholdBreached
- OcnrfSubscriptionMigrationInProgressWarn
- OcnrfSubscriptionMigrationInProgressCritical

KPIs

Following KPIs are added for the Subscription Limit feature:

- Global Subscription Limit Level
- Subscription Limit Breached

Maintain

If you encounter alerts at system or application levels, see NRF Alerts section for resolution steps.

In case the alerts still persists, perform the following:

1. **Collect the logs**: For more information on how to collect logs, see *Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide*.
2. **Raise a service request**: See My Oracle Support for more information on how to raise a service request.

Copyright © 2019, 2025, Oracle and/or its affiliates.

4.43 Automated Test Suite Support

NRF provides Automated Test Suite (ATS) for validating the functionalities. ATS allows you to run NRF test cases using an automated testing tool, and then compares the actual results with the expected or predicted results. In this process, there is no intervention from the user. For more information on installing and configuring ATS, see *Oracle Communications Cloud Native Core, Automated Test Suite User Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

5

Configuring NRF using CNC Console

This chapter describes how to configure different services in NRF using Oracle Communications Cloud Native Configuration Console (CNC Console).

5.1 Support for Multiple Cluster Deployment

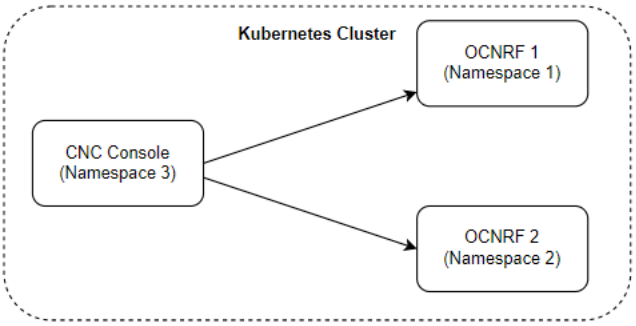
The CNC Console supports both single and multiple cluster deployments.

In a single cluster deployment, the CNC Console can manage NFs and Oracle Communications Cloud Native Core, Cloud Native Environment (CNE) common services deployed in the local Kubernetes clusters.

In a multiple instances deployment, the CNC Console can manage multiple NRF instances and CNE common services deployed within a Kubernetes cluster. For more information about single and multiple cluster deployments, see *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide*.

The following image represents a Kubernetes cluster with one instance of CNC Console and two instances of NRF. The single instance of the CNC Console is configuring two instances of NRF with different namespaces.

Figure 5-1 Support for Multiple Instance Deployment

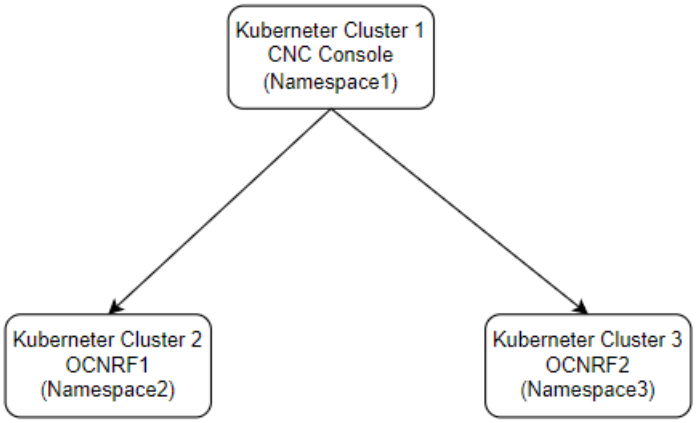


With the support of multicluster deployment, NRF deployed in multiple Kubernetes clusters can be accessed using CNC Console. In a multicluster deployment, the CNC Console can manage NRF and CNE common services deployed in the remote Kubernetes clusters.

The following image represents multiple Kubernetes clusters with one CNC Console and two NRF deployments. The single instance of the CNC Console is configuring two instances of NRF with different namespaces deployed in different clusters.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 5-2 Support for Multicluster Deployment



5.2 CNC Console Interface

This section provides an overview of the CNC Console, which includes an interface to configure NRF features.

You use NRF integrated with the CNC Console after logging in to the CNC Console application. To log in to the CNC Console, you need to make the following updates to the hosts file available at the **C:\Windows\System32\drivers\etc** location.

1. In the Windows system, user needs to open the hosts file in the notepad as an Administrator and append the following set of lines at the end:

   ```
   <CNCC Node IP> cncc-iam-ingress-gateway.cncc.svc.cluster.local
   <CNCC Node IP> cncc-core-ingress-gateway.cncc.svc.cluster.local
   ```

   For example:

   ```
   10.75.212.88 cncc-iam-ingress-gateway.cncc.svc.cluster.local
   10.75.212.88 cncc-core-ingress-gateway.cncc.svc.cluster.local
   ```

   > **Note**
   >
   > The IP Address in the above lines may change when deployment cluster changes.

2. Save and close the hosts file.
   Before logging into CNC Console, create a CNC user and password. Log in to the CNC console application using the same credentials. For information on creating a CNC Console user and password, see *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide*.

CNC Console Log in

Following is the procedure to log in to CNC Console:

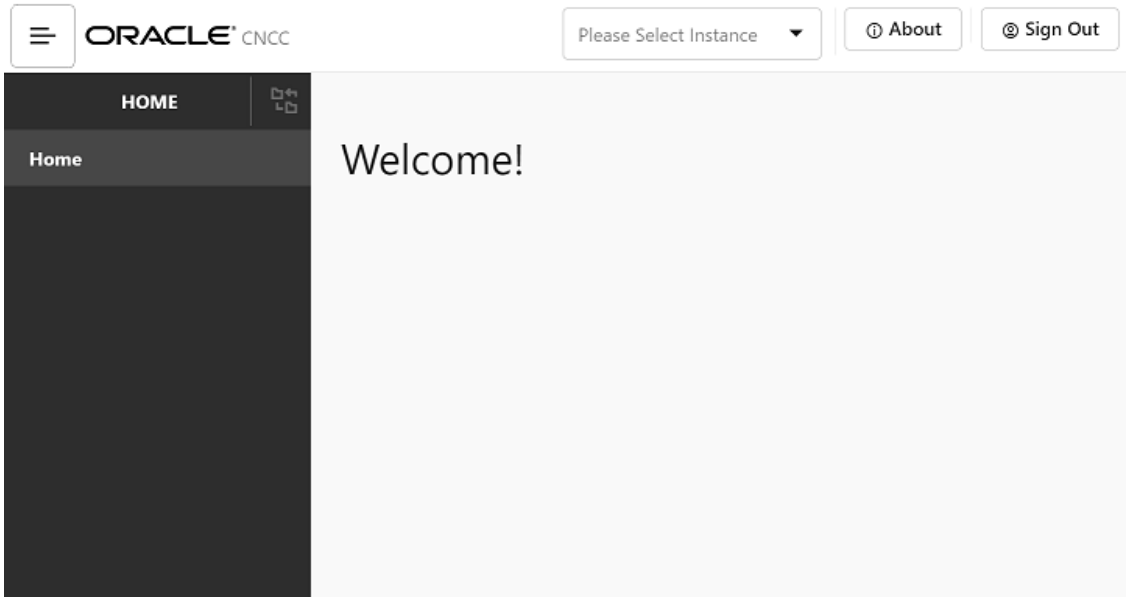
1. Open any browser.
2. Enter the URL: ***http://<host name>:<port number>***.

Copyright © 2019, 2025, Oracle and/or its affiliates.

where, *host name* is cncc-iam-ingress-ip and *port number* is cncc-iam-ingressport.

3. Enter valid credentials.
4. Click **Log in**. The CNC Console interface is displayed.

Figure 5-3 CNC Console



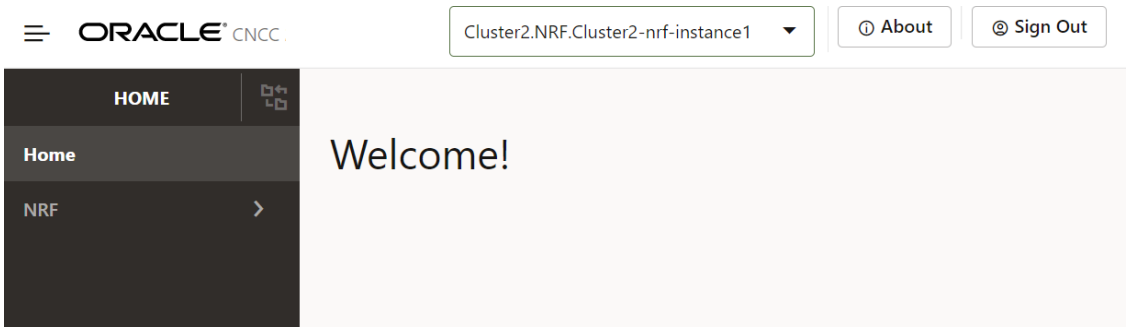
Select the required NF instance from the **Please Select Instance** drop-down list. The left pane displays the selected network function.

5.3 NRF Configuration

This section describes how to configure different features and services using CNC Console.

On selecting NRF instance from the drop-down list, the following screen appears:

Figure 5-4 NRF Welcome Screen



Copyright © 2019, 2025, Oracle and/or its affiliates.

5.3.1 cnDBTier APIs

1. From the left navigation menu, navigate to **NRF** and then click **cnDBTier** tab.
   The **cnDBTier** page is displayed.

5.3.1.1 cnDBTier Backup Status

1. From the left navigation menu, navigate to **NRF** and then click **cnDBTier** tab.
   The **cnDBTier** page is displayed.
2. Click **cnDBTier Backup Status** to view the cnDBTier backup status.
   The **cnDBTier Backup Status** page is displayed.

> **Note**
>
> The parameters listed in the below table are read-only.

Table 5-1 cnDBTier Backup Status

Attribute	Description
Current Timestamp	This attribute displays the current system time.
Is Backup In Progress	This attribute displays the current cnDBTier backup status.
Next Backup Scheduled Timestamp	This attribute displays the time for the next scheduled backup.

5.3.1.2 cnDBTier Health

1. From the left navigation menu, navigate to **NRF** and then click **cnDBTier** tab.
   The **cnDBTier** page is displayed.
2. Click **cnDBTier Health** to view the health status of the microservices like replication, backup manager, monitor services, and NDB services.
   The **cnDBTier Health** page is displayed.
   - Click the **Backup Manager Health Status** to view the health status of the backup manager.
     The **Backup Manager Health Status** page is displayed.

> **Note**
>
> TThe parameters listed in the below table are read-only.

Table 5-2 Backup Manager Health Status

Fields	Description
Service Name	This attribute displays the service name of the backup manager microservice.
Service Status	This attribute displays the service status of the backup manager microservice. Possible values are UP and DOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-2 (Cont.) Backup Manager Health Status

Fields	Description
DB Connection Status	This attribute displays the database connection status of the backup manager microservice. Possible values are UP and DOWN.
Overall Backup Manager Service Health	This attribute displays the overall health status of the backup manager microservice. Possible values are UP and DOWN.
Backup Executor Health Status	This attribute displays the following information like node id and DB connection status of the backup executor. – Node Id: This attribute displays the id of the node. – DB Connection Status: This attribute displays the backup executor database connection status with the nodes. Possible values are UP and DOWN.

- Click the **Monitor Health Status** to view the health status of the services.
  The **Monitor Health Status** page is displayed.

> **Note**
>
> The parameters listed in the below table are read-only.

Table 5-3 Monitor Health Status details

Attribute	Description
Service Name	This attribute displays the service name of the monitor microservice.
DB Connection Status	This attribute displays the database connection status of the monitor microservice. Possible values are UP and DOWN.
Metric Scrape Status	This attribute displays the status of the metric scrape, that is if the metrics are fetched or not. If the metrics are fetched then the service is up and vice versa. Possible values are UP and DOWN.
Overall Monitor Service Health	This attribute displays the overall health status of the monitor microservice. Possible values are UP and DOWN.

- Click the **NDB Health Status** to view the health status of the network database.
  The **NDB Health Status** page is displayed.

> **Note**
>
> The parameters listed in the below table are read-only.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-4 NDB Health Status details

Attribute	Description
Local Site Name	This attribute displays the name of the current site. For example, site 1, site 2.
NDB Health Status Details	This attribute displays the health status of the network database like name of the NDB service, status of the service, health status of PVC.
Service Name	This attribute displays the service name. For example, ndbmgmd-0, ndbmtd-0, ndbmyappsqld-1, ndbmysqld-2.
Service Status	This attribute displays the status of the service. Possible values are UP and DOWN.
PVC Health Status	This attribute displays the health status of the PVC. Possible values are UP, DOWN, and NA. **Note**: This attribute is set to NA when some of the database pods are not connected to the PVC.

- Click the **Replication Health Status** to view the health status of the replication sites. The **Replication Health Status** page is displayed.

> **Note**
>
> The parameters listed in the below table are read-only.

Table 5-5 Replication Health Status details

Attribute	Description
Local Site Name	This attribute displays the name of the current site (site 1, site 2).
Health Status Details	This attribute displays the health status details of the local site like replication service name, replication service status, database connection status of the replication service, and the overall health status of the replication micorservices. The number of rows in this table varies depending on the type of deployment (for example, two-site, three-site deployments).
Service Name	This attribute displays the name of the available replication service.
Service Status	This attribute displays the status of the available replication service. Possible values are UP and DOWN.
DB Connection Status	This attribute displays the database connection status of the replication microservice. Possible values are UP and DOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-5 (Cont.) Replication Health Status details

Attribute	Description
Overall Replication Service Health	This attribute displays the overall health status of the replication microservice. Possible values are UP and DOWN.

5.3.1.3 cnDBTier Version

1. From the left navigation menu, navigate to **NRF** and then click **cnDBTier** tab.
   The **cnDBTier** page is displayed.
2. Click **cnDBTier Version** to view the version.
   The **cnDBTier Version** page is displayed.

> **Note**
>
> The parameters listed in the below table are read-only.

Table 5-6 cnDBTier Version Attributes

Attribute	Description
cnDBTier Version	This attribute displays the cnDBTier version.
NDB Version	This attribute displays the network database (NDB) version.

5.3.1.4 Backup List

1. Click the **Backup List** to view the list of completed backups along with backup ID, backup size, and backup creation timestamp.
   The **Backup List** page is displayed.

> **Note**
>
> The parameters listed in the below table are read-only.

Table 5-7 Backup List

Fields	Description
Site Name	This attribute displays the name of the current site to which NRF is connected.
Backup Details	This attribute displays the following information like backup id, backup size, and backup creation timestamp.
Backup Id	This attribute displays the ID of the stored backup.
Backup Size (bytes)	This attribute displays the size of the stored backup.
Creation TimeStamp	This attribute displays the time recorded when the backup was stored.

Copyright © 2019, 2025, Oracle and/or its affiliates.

5.3.1.5 Database Statistics Report

1. From the left navigation menu, navigate to **NRF** and then click **cnDBTier** tab.
   The **cnDBTier** page is displayed.
2. Click **Database Statistics Report** to view the available databases.
   The **Database Statistics Report** page is displayed.

> **Note**
>
> The parameters listed in the below table are read-only.

Table 5-8 Database Statistics Report

Fields	Description
Database Count	This attribute displays the number of available database.
Database Tables Count	This attribute displays the available database names and their table count.
Database Name	This attribute displays the database name.
Table Count	This attribute displays the table count for each database.
Database Table Rows Count	This attribute displays the table rows present in each table.
Database Name	This attribute displays the database name.

a. Click **View** available next to the database name to view the **View Database Table Rows Count** screen.
   The **View Database Table Rows Count** page is displayed.

Table 5-9 View Database Table Rows Count

Fields	Description
Database Name	This attribute displays the database name.
Tables	This attribute displays the table names and the corresponding rows in each table.
Table Name	This attribute displays the table name.
Row Count	This attribute displays the table rows present in each table.

5.3.1.6 Georeplication Status

1. From the left navigation menu, navigate to **NRF** and then click **cnDBTier** tab.
   The **cnDBTier** page is displayed.
2. Click **Georeplication Status** to view the local site and remote site name to which NRF is connected.
   The **Georeplication Status** page is displayed.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> The parameters listed in the below table are read-only.

Table 5-10 Georeplication Status

Attribute	Description
Local Site Name	This attribute displays the local site name to which NRF is connected. **Note:** The number of local site names may vary depending on the type of georeplication used in NRF.
Remote Site Name	This attribute displays the remote site name. **Note:** The number of remote site names may vary depending on the type of georeplication used in NRF.
Replication Status	This attribute displays the replication status with corresponding sites. **Note:** The number of replication statuses may vary depending on the type of georeplication used in NRF.
Seconds Behind Remote Site	This attribute displays the number of seconds that the last record read by the local site is behind the latest record written by the remote site for all the replication groups. **Note:** The number of replication statuses may vary depending on the type of georeplication used in NRF.

a. Click **View** option in the **Actions** menu, to view the **View Georeplication Status** screen.
The **Georeplication Status** page is displayed.

Table 5-11 Georeplication Status

Attribute	Description
Replication Group Delay	This attribute displays the seconds behind the remote site for individual replication groups.
Replication Channel Group Id	This attribute displays the ID of the replication channel group.

b. Click on the option to view the **Replication Group Delay** attributes.
The **Replication Group Delay** page is displayed.

Table 5-12 View Replication Group Delay

Attribute	Description
Channel Details	This attribute displays the channel details such as Remote Replication IP and Role.
Remote Replication IP	This attribute displays the IP of the remote replication channel.
Role	This attribute displays the role of the replication channel IP.

Copyright © 2019, 2025, Oracle and/or its affiliates.

5.3.1.7 Georeplication Recovery

Perform the following procedure to mark cnDBTier cluster as failed, execute georeplication recovery, and monitor their status:

1. From the left navigation menu, navigate to **NRF** and then click **cnDBTier** tab.
2. Click **Georeplication Recovery** to access the Georeplication Recovery Status of the cnDBTier cluster. This includes options such as Update Cluster As Failed, Start Georeplication Recovery, and Georeplication Recovery Status.
   - Click **Update Cluster As Failed** to mark the cnDBTier cluster as **FAILED**.
     The **Update Cluster As Failed** page is displayed.

     Table 5-13 Update Cluster As Failed

Fields	Description
Cluster Names	This field displays a list of cnDBTier clusters that can be marked as failed.
Failed Cluster Names	This field displays a cnDBTier cluster that is marked as failed.

     Click **Update Cluster**. The selected cnDBTier cluster name is updated in the **Failed Cluster Names** field.
   - Click **Start Georeplication Recovery** to start the georeplication recovery process for a failed site.
     The **Start Georeplication Recovery** page is displayed.

     Table 5-14 Start Georeplication Recovery

Attribute	Description
Failed Cluster Name	This field displays a list of all the cnDBTier clusters that have been marked as failed.
Backup Cluster Name (Optional)	This field displays a list of all the healthy cnDBTier clusters. If no cnDBTier cluster is selected, the system uses the first available healthy cnDBTier cluster for the backup.

     Click **Start Georeplication Recovery** to initiate georeplication recovery.
   - Click **Georeplication Recovery Status** to view the status of georeplication recovery for cnDBTier clusters.
     The **Georeplication Recovery Status** page is displayed.

     Table 5-15 Georeplication Recovery Status

Attribute	Description
Local Cluster Name	This field displays the name of the local cnDBTier cluster.
Georeplication Recover Status Details	This field displays the details of the georeplication recovery status of cnDBTier clusters.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-15 (Cont.) Georeplication Recovery Status

Attribute	Description
Cluster Name	This field displays the cnDBTier clusters by name.
Georeplication Recovery Status	This field displays the current georeplication recovery status of the corresponding cnDBTier cluster. For more information about the georeplication recovery status, see *Oracle Communications Cloud Native Core, cnDBTier User Guide.*

5.3.1.8 Replication HeartBeat Status

1. From the left navigation menu, navigate to **NRF** and then click **cnDBTier** tab.
   The **cnDBTier** page is displayed.
2. Click **Replication HeartBeat Status** to view the connectivity between local site and remote site name to which NRF is connected.
   The **Replication HeartBeat Status** page is displayed.

> **Note**
>
> The parameters listed in the below table are read-only.

Table 5-16 Replication HeartBeat Status Details

Fields	Description
Site Name	This attribute displays the name of the current site to which NRF is connected.
HeartBeat Details	This attribute displays the following information like remote site name, heartbeat status, heartbeat lag, and replication channel group id.
Remote Site Name	This attribute displays the remote site name.
Heartbeat Status	This attribute displays the connectivity status with corresponding sites.
Heartbeat Lag	This attribute displays the lag or latency in seconds it took to synchronize between sites.
Replication Channel Group Id	This attribute displays the ID of the replication channel group.

5.3.1.9 Local Cluster Status

1. From the left navigation menu, navigate to **NRF** and then click **cnDBTier** tab.
   The **cnDBTier** page is displayed.
2. Click **Local Cluster Status** to view the local cluster status for the current site.
   The **Local Cluster Status** page is displayed.

> **Note**
>
> The parameters listed in the below table are read-only.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-17 Local Cluster Status

Attribute	Description
Cluster Name	This attribute displays the name of the current site to which NRF is connected.
Cluster Status	This attribute displays the local cluster status for the current site.

5.3.1.10 On Demand Backup

1. From the left navigation menu, navigate to **NRF** and then click **cnDBTier** tab.
   The **cnDBTier** page is displayed.
2. Click the **On Demand Backup** to create a new backup and view the status of initiated on-demand backups.
   The **On Demand Backup** page is displayed.

> **Note**
>
> The parameters listed in the below table are read-only.

Table 5-18 On Demand Backup Details

Fields	Description
Site Name	This attribute displays the name of the current site to which NRF is connected.
DR Status	This attribute displays the disaster recovery status .
Backup Id	This attribute displays the ID of the stored backup.
Backup Status	This attribute displays the status of backup.
Remote Transfer Status	The attribute displays the status of remote transfer.
Initiate Backup	The attribute displays whether the backup is initiated or not. **Note:**You can read and write this API.

- Click **Edit**. The **Edit** option On Demand Backup page appears.

> **Note**
>
> The **Edit** mode is available only for Initiate Backup.

- Use the Toggle option to Initiate the backup and click **Save**. A confirmation message "Save successfully" appears.
- Click **Cancel** to navigate back to the On Demand Backup page.
- Click **Refresh** to reload the On Demand Backup page.

Copyright © 2019, 2025, Oracle and/or its affiliates.

5.3.2 General Options

Perform the following procedure to configure the **General Options**:

1. From the left navigation menu, navigate to **NRF** and then click **General Options**. The **General Options** page is displayed.
2. Click **Edit** from the top right side to edit or update **General Options** parameter. The page is enabled for modification.
3. Configure **General Options** fields as described in the following table:

Table 5-19 General Options

Field Name	Description
Default Load Assignment	If NFProfile does not have Load attribute, **Default Load** value is set in NF Load attribute of NFProfile while sending the NFDiscover response and NFProfile sent in NFNotify operation. If the value of the **Service Priority Update Feature Status** feature flag is enabled in NF Discovery Options and if NFService/NFProfile does not have any value for the Load attribute, the **Default Load** value is set in NF Service and NF Profile load attribute while sending the NFDiscover response. For more information, about the inheritance logic, see NFService Priority Update.
Default Priority Assignment	If NFProfile does not have Priority attribute, **Default Priority** value is set in NF Load attribute of NFProfile while sending the NFDiscover response and NFProfile sent in NFNotify operation. If the value of the **Service Priority Update Feature Status** feature flag is enabled in NF Discovery Options and if NFService/NFProfile does not have any value for the Priority attribute, the **Default Priority** value is set in NF Service and NF Profile Priority attribute while sending the NFDiscover response. For more information, about the inheritance logic, see NFService Priority Update.
Default Load	If the value of this attribute is set to true, then defaultLoad value is set in NF load attribute of NFProfile. This value is sent in NFDiscover response and NFProfile sent in NFNotify operation, in case NFProfile does not have load attribute.
Default Priority	This attribute is default value of NF Priority and will be used if NFProfile does not have priority attribute set by NF.
Enable 29510 v15.3.0	functions as per 3GPP 29510 v15.3 specification, if this flag is set to true. If it is set to true, then NRF will compliant to 3GPP 29510 v15.3. If it is set to false, NRF will compliant to 3GPP 29510 v15.2.
Enable 29510 v15.5.0	NRF functions as per 3GPP 29510 v15.5 specification, if this flag is set to true. If it is set to false, NRF functions as per 3GPP 29510 v15.2 or v15.3 specification (depends on enableF3 flag).

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-19 (Cont.) General Options

Field Name	Description
Maximum Hop Count	Maximum number of Nodes (SLF/NRF's) that NRF can communicate, to service a request. **Note**: The value of this parameter must be greater than the value configured for the **Maximum SLF Attempts** parameter in the SLF Options.
OCNRF Scheme	NRF Host's Scheme
OCNRF Host	ocnrfHost needs to be NRF's External Routable FQDN (for example, ocnrf.oracle.com) or External Routable IpAddress
OCNRF Port	NRF Host's Port
NRF PLMN List	This value will have at least one PLMN supported by NRF and this value is set before using NRF.
Add 3GPP SBI Correlation Info Header	This attribute indicates whether the `3gpp-Sbi-Correlation-Info` can be added to the request. If this flag is set to ENABLED, then `3gpp-Sbi-Correlation-Info` header is added.
OCNRF User-Agent Header	This attribute indicates the value of the User-Agent header that is added to the outgoing requests for SLF query and NFStatusNotify. The recommended value starts with NRF followed by -<operator specific string/value>. Few examples, NRF-<NrfInstanceId>, For example: NRF-4947a69a-f61b-4bc1-b9da-47c9c5d14b64 NRF-<NRF FQDN>, For example: NRF-nrf05.testnetwork.org NRF-<NrfInstanceId> <NRF FQDN>, For example: NRF-4947a69a-f61b-4bc1-b9da-47c9c5d14b64 nrf05.testnetwork.org **Note**: a. The length of the value can be 1-300 characters. b. String containing only blank spaces is not supported. However, it supports blank spaces in between. c. The feature is disabled for an empty string. d. 3GPP validation is not performed for the configured value.

For more information on parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

4. Click **Add** provided under **NRF PLMN List** section to add MCC and MNC.
5. Enter **MCC** and **MNC** values in the field:

Table 5-20 Add NRF PLMN List

Field Name	Description
MCC	Enter the MCC value.
MNC	Enter the MNC value.

6. Click **Save** on **Add NRF PLMN List** page.

Copyright © 2019, 2025, Oracle and/or its affiliates.

7. Click **Save** on **Edit General Options** page to save **General Options**.

For more information on parameter values, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

> **Note**
>
> Use the **Edit** or **Delete** icons available in the next column of the PLMN to update or delete the **NRF PLMN List** information.

5.3.3 NF Management Options

Perform the following procedure to configure the **NF Management Options**:

1. From the left navigation menu, navigate to **NRF** and then click **NF Management Options**. The **NF Management Options** page is displayed.
2. Click **Edit** from the top right side to edit or update **NF Management Options** parameter. The page is enabled for modification.
3. Configure **NF Management Options** fields as described in the following table:

Table 5-21 NF Management Options

Field Name	Description
NF Notify Load Threshold	NRF generates the notification trigger when the difference between the last notified load value and the current reported load value is equal or greater than the configured value of `nfNotifyloadThreshold` attribute. **Note**: • NRF generates the notification trigger if the nfProfile level load is added or removed. • This feature applies to nfProfile level load only. NRF triggers a notification for every change in the service level load. • This feature is applicable with NfHeartBeat and NfUpdate (partial and full) service operation.
NRF Support For Profile Changes In Response	When this flag is set to true and the nfprofile contains *nfProfileChangesSupportInd* set to true, NRF will send only the changes in the Profile in the response. When this value is set to false, the complete profile will be sent in the response.
Default Subscription Validity Time	If the validity time attribute is not received in SubscriptionData during NFStatusSubscribe, this default value will be used for calculation of validity time (current time + default duration). If the validity time attribute is received in SubscriptionData during NFStatusSubscribe, this is minimum value will be used for validation and limit purpose. It means if value provided is less than (current time + minimum possible range value), then minimum range value will be considered as validity time for subscription and similarly in case validity time is more than (current time + maximum possible range value), then maximum range value will be considered as validity time for subscription.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-21 (Cont.) NF Management Options

Field Name	Description
NRF Support For Profile Changes In Notification	NRF sends profileChanges attribute instead of NFProfile in Notification, if this flag is enabled.
NF Profile Suspend Duration	Indicates the duration for which the NF is suspended, before it is deleted from NRF database. The value is in pHqMrS format. Where p,q,r are integers and H,M,S or h,m,s denote hours, minutes and seconds respectively.
Accept Additional Attributes	NRF preserves additional attributes that are not defined by 3GPP in NFProfile/NFService based on this attribute value.
Allow Duplicate Subscriptions	This attribute specifies if NRF allows creation of duplicate subscriptions. • If this value is set as true, for every subscription request, NRF will create a new Subscription without checking if there is already a subscription request present. For more use cases, see Use Cases for Allow Duplicate Subscriptions. • If this value is set as false, for every subscription create request, all the attributes of the subscription data JSON Object, except the validityTime attribute, is checked against all the existing subscriptions to see if there is an exact match: – If a duplicate subscription is found, NRF returns "201 Created along with the existing SubscriptionId and the existing validityTime". By this NRF will be accepting the subscription create request, but will not create a duplicate subscription. – If a duplicate subscription is not found, NRF creates the subscription and returns "201 Created with the new SubscriptionId". **Note**: If the value of Allow Duplicate Subscriptions is set as false, NRF would check for duplicate subscription by matching the current subscription request with every subscription present in NRF. Hence, this causes a performance degradation of around 50% during NFStatusSubscribe and NfStatusNotify service operation.
NF Heartbeat Timers	This attribute is used to configure the heartbeat related information of the NF. It allows to configure the heartbeat information per NFType. By default, the nfHeartbeatTimer information for ALL_NF_TYPE is present.
Notification Retry	This attribute is used to configure the Notification Retry feature.
Subscription Limit	This attribute is used to configure the Subscription Limit feature.
NF Profile Limit	This atttribute is used to configure the NF Profile Size Limit feature.

4. Click **Add** provided under **NF Heartbeat Timers** section to add the Heartbeat Timers parameters.
5. Enter the values for heartbeat timers as described in the *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-22 Add NF Heartbeat Timers

Attribute	Description
NF Type	Enter the supported NFType. All nftypes supported in 29.510 Rel 15.5.
Minimum Heartbeat Timer	The minimum HeartbeatTimer allowed for the NF.
Maximum Heartbeat Timer	The maximum HeartbeatTimer allowed for the NF.
Default Heartbeat Timer	This default heartBeatTimer value to be used when the network functions does not provide the heartBeatTimer value in NFProfile.
Number of Heartbeats NF type can miss	The allowed number of missed HeartBeat(s) after which the NFProfile is marked as suspended.

6. Click **Save** on **Add NF Heartbeat Timers** page.
7. Configure **Notification Retry** parameters as described in the following table:

Table 5-23 Notification Retry

Attribute	Description
Feature Status	This flag enables or disables the *NfStatusNotify* service operation request. **Note**: If any attribute is null in request body, then the previously saved values are retained and considered. If Feature Status value is ENABLED, Error Response Code List and Exception Response List cannot be simultaneously null, either of the list must be present. However, if Feature Status value is DISABLED, both Error Response Code List and Exception Response List can be simultaneously empty. Feature Status value cannot be ENABLED, if both Error Response Code List and Exception Response List are null.
Retry Count	The number of retries that happens at Egress Gateway upon the following scenarios: • 4xx, 5xx Response Error Codes: In case of the error response from the notification callback server, http status codes are matched with the statuses. • Connection Failure/Timeout: In case of connection failure/timeout, error occurred at Egress Gateway is matched with the Exception Response List. • Request Timeout: In case of request timeout, error occurred at Egress Gateway is matched with the Exception Response List.
Request Timeout	This configuration decides the request timeout if response from notification callback server is not received. **Note:** The value here corresponds in milliseconds. This value is used only if java.util.concurrent.TimeoutException is configured in Exception Response List attribute.
Error Response Code List	This configuration is the list of HTTP error codes that are authorized for retry. In case of the 4xx, 5xx Response Error Codes from the notification callback server, the HTTP status codes from error response are matched with this configuration.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-23 (Cont.) Notification Retry

Attribute	Description
Exception Response List	This configuration is the list of exceptions that are authorized for retry. Below is the set of possible exceptions: • java.net.UnknownHostException: Exception is raised when the address of a host could not be determined, or host address is invalid • javax.net.ssl.SSLHandshakeException:Exception is raised when the client and server could not negotiate the desired level of security. • java.nio.channels.ClosedChannelException: Exception is raised when an attempt is made to invoke or complete an I/O operation upon channel that is closed, or at least closed to that operation • java.net.ConnectException: Exception is raised when an error occurred while attempting to connect a socket to a remote address and port • java.util.concurrent.RejectedExecutionException: Exception is raised when a task cannot be accepted by the executor • java.util.concurrent.TimeoutException: Exception is raised when the NFStatusNotify request is timed out • java.net.SocketTimeoutException: Exception is raised when a timeout has occurred on a socket read or accept. This occurs in case of connection is not established within time.

8. Configure **Subscription Limit** parameters as described in the following table:

Table 5-24 Subscription Limit

Attribute	Description
Feature Status	This attribute is used to enable or disable the Subscription Limit feature.
Global Max Limit	This attribute is used to set the maximum number of subscriptions allowed for the NRF. **Note**: The value of this attribute must be same across all georedundant site.
Reject Subscription Renewal When Limit Breached	This flag is used to indicate whether Subscription Renewal is allowed when the Global Subscription limit is breached.
Limit Thresholds	This attribute is used to configure the subscription limit thresholds for which alerts are raised. **Note**: Duplicate level name is not allowed in limitThresholds.
Error Responses	Indicates the error response that are generated when a subscription request is rejected.

9. Click **Add** provided under **Error Responses** section to add the error response details. The **Add Error Responses** page is displayed.
10. Enter the values for error response configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
11. Click **Save** on **Add Error Responses** page.
12. Click **Add** provided under **Limit Thresholds** section to add the limit thresholds details. The **Add Limit Thresholds** page is displayed.
13. Enter the values for limit thresholds configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
14. Click **Save** on **Add Limit Thresholds** page.

Copyright © 2019, 2025, Oracle and/or its affiliates.

15. Configure **NF Profile Limit** parameters as described in the following table:

Table 5-25 NF Profile Limit

Attribute	Description
Feature Status	This attribute is used to enable or disable the NF Profile Size Limit feature.
NF Profile Size Limits	This attribute is used to set the maximum size for the NF Profile.
Error Responses	This attribute provides the error response that are generated when a NF Profile exceeds the maximum defined size.

16. Click **Add** provided under **NF Profile Size Limits** section to add the NF Profile limit details. The **Add NF Profile Size Limits** page is displayed.

Table 5-26 NF Profile Size Limits

Attribute	Description
NF Type	This attribute indicates the NF type for which the profile size should be limited.
Max Size	This attribute indicates the maximum allowed size of NF Profile being registered or updated in bytes. If the value is set as 0, then there is no restriction in the maximum size of the NF Profile. NRF allows the NF Profiles as received from the Consumer NF to the database.

17. Click **Save** on **Add NF Profile Size Limits** page.
18. Enter the values for NF Profile size limits configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
19. Click **Add** provided under **Error Responses** section to add the error response details. The **Add Error Responses** page is displayed.
20. Enter the values for error response configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
21. Click **Save** on **Edit NF Management Options** page to save **NF Management Options**.

For more information on parameter values, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

> **Note**
>
> Use the **Edit** or **Delete** icons available in the next column of the heartbeat timer to update or delete the **NF Heartbeat Timers** information.

5.3.4 NF Discovery Options

Perform the following procedure to configure the **NF Discovery Options**:

1. From the left navigation menu, navigate to **NRF** and then click **NF Discovery Options**. The **NF Discovery Options** page is displayed.
2. Click **Edit** from the top right side to edit or update **NF Discovery Options** parameter.

Copyright © 2019, 2025, Oracle and/or its affiliates.

The page is enabled for modification.

3. Configure **NF Discovery Options** fields as described in the following table:

Table 5-27 NF Discovery Options

Field Name	Description
Discovery Result Load Threshold	NFDiscover response contains NF profiles with load attribute value less than or equal to this configured value. In case there are no profiles matching this criteria, the profiles with load greater than the configured value are included in the response. If service-names attribute is present in the discovery query and the NF profile has only one NF service instance and that NF service load value is non-null, then that load value is considered. Otherwise, the NF profile load value is considered, if its non-null. If the considered load value is non-null and has value greater than the load threshold value, then that NF Profile gets filtered out. Value 0 indicates this feature is disabled. Increment or decrement the **Discovery Result Load Threshold** parameter using the arrow keys.
Profile Count In Discovery Response	This value restricts NF profile count in NFDiscover response. If value of this attribute is 0, it means this functionality is disabled. In that case all of the NF profiles after the discovery filtering will be returned in NFDiscover response. Increment or decrement the **Profiles Count In Discovery Response** parameter using the arrow keys. **Note**: If Limit attribute is present in SearchData URI, then this attribute is not used.
Discovery Validity Period	This attribute mentions the validity period of a discovery request for a specific target-nf-type after which requester NF must perform discovery again to get the latest values. By default, the validityPeriod information for ALL_NF_TYPE is present.
Empty Discovery Response	This attribute defines the configuration for EmptyList feature.
Extended Preferred Locality	This attribute is used for Extended Preferred Locality feature.
Service Priority Update Feature Status	This attribute determines if the NFService Priority Update feature is enabled or disabled. If this feature flag is **ENABLED**, NRF updates the NFService level priority along with the NFProfile level priority while processing the discovery query. If this feature flag is **DISABLED**, NRF does not update the NFService level priority along with the NFProfile level priority while processing the discovery query.

4. Select the **Service Priority Update Feature Status** value as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
5. Click **Add** provided under **Discovery Validity Period** section to add the validity information.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-28 Add Discovery Validity Period

Attribute	Description
NF Type	Enter the supported NFType. All nftypes supported in 29.510 Rel 15.5.
Validity Period	Specify the validity period of a discovery request of a specific target-nf-type after which requester NF must perform discovery again to get the latest values. The value is in pHqMrS format.
Empty List Validity Period	This attribute mentions the validity period for an empty list response of a discovery request of a specific target-nf-type. This value is sent as `validityPeriod` in the discovery response when NRF is generating empty response, irrespective of `emptyListFeatureStatus` is ENABLED or DISABLED. In case the specific nfType is not configured, then `emptyListValidityPeriod` configured for ALL_NF_TYPE is considered in the discovery response. Upon expiry of the `emptyListValidityPeriod`, Producer NF must send the discovery request again. The value is in pHqMrS format, where p, q, r are integers and H, M, S or h, m, s denotes hours, minutes, and seconds respectively.

6. Click **Save** on **Add Discovery Validity Period** page.
7. Click **Add** provided under **Empty List Configuration** section to the nfTypes for which EmptyList must be configured.
8. Configure the **NF Type** and **Feature Status** per nfType on the **Add Empty List Configuration** page.
9. Click **Save** on **Add Empty List Configuration** page.
10. Select the **Feature Status** and **Location Types** values provided under **Extended Preferred Locality** section as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
11. Click **Add** provided under **Location Type Mapping** section to add the location information.
12. Enter the location type mapping values as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Table 5-29 Add Location Type Mapping

Attribute	Description
NF Type	NF type of particular Network Function. This value is derived from target-nf-type of Discovery Search query.
NF Services	NF Services belonging to Network Function
Location Type	Location Type to which NF is mapped

13. Click **Save**.
14. Click **Add** provided under **Preferred Location Details** section to add the preferred location information.

Copyright © 2019, 2025, Oracle and/or its affiliates.

15. Provide the configuration for Preferred location details as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Table 5-30 Add Preferred Location Details

Attribute	Description
Preferred Location	This value is matched with preferredLocation from consumer NF in Discovery Search Query.
Target Location Type	This value is derived from mapped locationType attribute value of LocationTypeMapping table.
Maximum number of NF Profiles from First Matching Location	This value is configured to define the maximum number of NF profiles that can be selected from the first matching location from the Target Preferred Locations.
Target Preferred Locations	Preferred Locations decided by operator for particular preferredLocation and targetLocationType.

a. Click **Add** provided under **Target Preferred Locations** section to add the target preferred location information:

Table 5-31 Add Target Preferred Locations

Attribute	Description
Priority	Set the priority using the arrows
Location	Enter the target location or location set

b. In case **Location** is a set of locations, configure **Location Sets**. Click **Add** provided under **Location Sets** section to add the location information:

Table 5-32 Location Sets

Attribute	Description
Location Set Name	An identifier to distinguish among other locations. It must be alphanumeric and allowed special characters are **-** and _. **Note**: It is recommended to append **Set** as a keyword for every `locationSetName` specifying group of locations, for easy identification between a single location and group of locations. The length of the location name can be in the range of 5 to 100 characters.
Locations	Set of unique locations, where location is of type string, and n is predefined maximum locations per location set. A maximum of 10 locations can be configured.

c. Click **Save** to save the location set.

d. Click **Save** on **Add Target Preferred Locations** page to save the target preferred location.

16. Click **Save** on **Add Preferred Location Details** page to save the preferred location.

17. Click **Save** on **Edit NF Discovery Options** page to save **NF Discovery Options**.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> Use the **Edit** or **Delete** icons displayed on the next column of the specific entry to update or delete the **Discovery Validity Period**, **Location Type Mapping**, and **Preferred Location Details** information.

5.3.5 NF Access Token Options

Perform the following procedure to configure the **NF Access Token Options**:

1. From the left navigation menu, navigate to **NRF** and then click **NF Access Token** .
2. Click **AccessToken Options** under **NF AccessToken**
   The **NF AccessToken Options** page is displayed.
3. Click **Edit** from the top right side to edit or update **NF AccessToken Options** parameter.
   The page is enabled for modification.
4. Configure **AccessToken Options** fields as described in the following table:

Table 5-33 AccessToken Options

Field Name	Description
Audience Type	Select the value for the AudienceType in AccessTokenClaim. NRF considers this value only if targetnfInstanceId is not received in AccessTokenRequest. NF_INSTANCE_ID: NF Instance Id(s) in audience IE of AccessTokenClaim NF_TYPE: NF Type in audience IE of AccessTokenClaim
Authorize Requester NF	Select the attribute to validate the requester NF is registered with NRF If NF is registered, then check if NFtype in Access Token Request is same as in NF profile registered with NRF and requesterPlmn received in the Access Token Request request is same as Registered Profile. If the value is DISABLED, NRF will issue token to non-registered NFs as well.
Logical Operator For Scope	Select the value to decide whether values in scope will have relationship AND or OR. If value is AND, while looking for producer network function profiles, token will be issued for profiles matching all the services-names present in scope. If value is OR, token will be issued for profiles matching any of the services-names present in scope.
OAuth Token Expiry Time	Set the Oauth token expiry time. **Note**: In case NRF signed certificate expiry duration is less than this attribute, then certificate expiry duration is used in Access Token Expiry Time.
Access token authorization feature	The attribute contains the parameters required to enable and configure NfAccessToken Authorization Feature.
Token Signing Details	This attribute allows user to configure all of the details required to sign the token generated by NRF.
Access token authorization error responses	This attribute allows user to update details for different error conditions.

5. Select **Feature status** provided under **Access token authorization feature** section.

Copyright © 2019, 2025, Oracle and/or its affiliates.

6. Click **Add** provided under **Access token authorization feature rules configuration** section to add the rules.
7. Enter the values for rules configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

> **Note**
>
> The fields **Feature status**, **Access token authorization feature rules configuration** and **Add Access token authorization error responses** can be independently configured in any order. However, when the **Feature status** is ENABLED, the **Access token authorization feature rules configuration** must be preconfigured or be present in the current request.

Table 5-34 AuthConfig

Attribute	Description
Target NF Type	Enter the attribute to define the NF Type of the target NF.
Requester NF Type	Enter the attribute to define the NF Type of the requester NF that is authorized to access the target NF Type and its services.
Service Names	Enter the attribute to define the NF services that is authorized to be accessed by the requester NF type. The value "*" indicates that all the services are authorized to be accessed the requester NF Type. If "*" is to be used, the services contain only a single entry in the list with this value.

8. Click **Save** on **Add Access token authorization feature rules configuration** page.
9. Enter the **Token Signing Details** as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
10. Enter the **Default K8s Secret Details** as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
11. Click **Add** provided under **Key Details List** section to add the Key IDs details.
12. Enter the values for rules configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Table 5-35 OauthKeyDetails

Attribute	Description
Key ID	Unique value in list of keys. Key Details are known by this value.
Algorithm	Algorithm value will be used to sign the oauth token.
Private Key	Enter the NRF Private key details: • K8s Secret Name • K8s Secret NameSpace • File Name
Certificate	Enter the NRF Public certificate details. • K8s Secret Name • K8s Secret NameSpace • File Name

Copyright © 2019, 2025, Oracle and/or its affiliates.

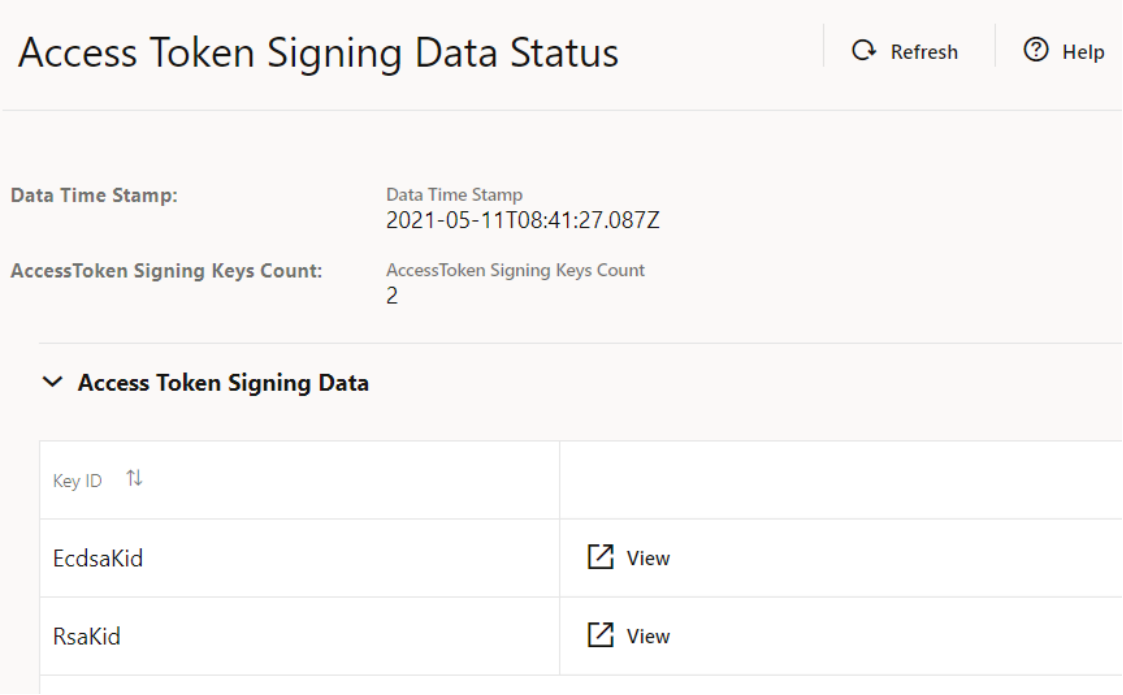
13. Click **Save** on **Add Key Details List** page.
14. Click **Add** provided under **Access token authorization error responses** section to add the error response details.
15. Enter the values for rules configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
16. Click **Save** on **Add Access token authorization error responses** page.
17. Click **Save** to save the NF AccessToken Options.

> **Note**
>
> Use the **Edit** or **Delete** icons available in the next column of the specific entry to update or delete the **Access token authorization feature rules configuration**, **Token Signing Details**, **Key Details List** and **Access token authorization error responses** information.

5.3.5.1 Access Token Signing Data Status

This screen displays the status of Access Token signing data. The data is fetched from NF Access Token Options.

Figure 5-5 Access Token Signing Data Status



5.3.6 Forwarding Options

Perform the following procedure to configure the **Forwarding Options**:

1. From the left navigation menu, navigate to **NRF** and then click **Forwarding Options**. The **Forwarding Options** page is displayed.

Copyright © 2019, 2025, Oracle and/or its affiliates.

2. Click **Edit** from the top right side to edit or update **Forwarding Options** parameter. The page is enabled for modification.
3. Configure **Forwarding Options** fields as described in the following table:

Table 5-36 Forwarding Options

Field Name	Description
NF Profile Retrieval Status	This attribute controls the forwarding of NFProfileRetrieval service operation messages.
Subscription Status	This attribute controls the forwarding of NFStatusSubscribe, NFStatusUnsubscribe service operation messages. **Note**: NFStatusSubscribe forwarding is supported only if Subscription Condition is NfInstanceIdCond in the NFStatusSubscribe request.
Discovery Status	This attribute controls the forwarding of NFDiscover service operation messages.
AccessToken Status	This attribute controls the forwarding of AccessToken service operation messages.
NRF Host Config	This is used to configure Primary and alternate NRF Details which is used for forwarding various requests. **Note**: The value of this attribute can be FQDN, IPv4 or IPv6.
NRF Reroute On Response HTTP Status Codes	This configuration is used to determine if the service operation message needs to forwarded to alternate NRF.
Forwarding Rules Feature Config	This attribute provide details for Forwarding Rules feature configuration.
Error Responses	This attribute defines the error responses which may be sent during NRF Forwarding scenarios.

4. Click **Add** provided under **NRF Host Config** section to add the host information. The **Add NRF Host Config** page is displayed.
5. Enter the values for rules configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Table 5-37 NfConfig

Field Name	Description
Scheme	URI schema supported by NF
Host	Host of NF
Port	Port of NF
apiPrefix	ApiPrefix
Priority	Priority of NF
NF Instance ID	NF Instance Id of NF

   a. Click **Add** provided under **API Versions** section.
   b. Enter **API Full Version** and **API Version In Uri** values.
   c. Click **Save** on **Add API Versions** page.
6. Click **Save** on **Add NRF Host Config** page.
7. Select the **Feature Status** value provided under **Forwarding Rules Feature Config** section to set the feature configuration.

Copyright © 2019, 2025, Oracle and/or its affiliates.

This flag can be **ENABLED** only if **Discovery Status** or **AccessToken Status** attributes are **ENABLED**.

> **Note**
>
> Once **Feature Status** flag is **ENABLED** for **Forwarding Rules Feature Config**, both **Discovery Status** and **AccessToken Status** forwarding cannot be disabled.

8. Click **Add** provided under **Forwarding Rules Feature Config** section to add the Forwarding Rules feature configuration.
   The **Add Forwarding Rules Config** page is displayed.

   Enter the values for rules configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Table 5-38 Add Forwarding Rules Config

Field Name	Description
Target NF Type	This attribute defines target NF type in a forwarding rule.
Service Names Match Type	This attribute provides details on how the service names are evaluated, based on the defined constraints. **Exact**: Service Names in incoming request must be present in the configured service names. **Anyone**: Service Names in incoming request must match with any one of the configured service names. **Note**: If serviceNames is a wildcard attribute, then this attribute can be skipped.
Service Names	List of services allowed for target NF type.

9. Click **Save** on **Add Forwarding Rules Config** page.
10. Click **Add** provided under **Error Responses** section to add the error response details.
    The **Add Error Responses** page is displayed.
11. Enter the values for rules configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
12. Click **Save** on **Add Error Responses** page.
13. Click **Save** to save the **Forwarding Options**.

> **Note**
>
> Use the **Edit** or **Delete** icons available in the next column of the specific entry to update or delete the **NRF Host Config**, **Forwarding Rules Feature Config**, and **Error Responses** information.

5.3.7 SLF Options

Perform the following procedure to configure the **SLF Options**:

1. From the left navigation menu, navigate to **NRF** and then click **SLF Options**.
   The **SLF Options** page is displayed.
2. Click **Edit** from the top right side to edit or update **SLF Options** parameter.

Copyright © 2019, 2025, Oracle and/or its affiliates.

The page is enabled for modification.

3. Configure **SLF Options** fields as described in the following table:

Table 5-39 SLF Options

Field Name	Description
Feature Status	Enables or disables the SLF Feature. • If the GroupId is already present in Search Query, then the value of featureStatus is also ENABLED. NRF uses the Group Id received and not communicate to SLF or UDR. • If the Subscriber Id is present in Search Query, it is ignored and not be used to perform discovery search in NRF.
UseOAuthToken	This attribute is used while performing the SLF query to SLF or UDR to query the Access Token service to get Oauth access token details. If the value of this attribute is **true**, the SLF function of NRF accesses the Access Token service. If the value of this attribute is **false**, the SLF function does not access the Access Token service and performs the SLF query without Oauth Access Token. **Note**: This attribute is considered only if the ***oauth2Required*** attribute is not configured in the registered UDRs. For more information about ***oauth2Required*** attribute, see 3GPP 29.510 v16.
Cache AccessToken	When this attribute is **true**, NRF will cache the oAuth2 token for SLF communication and use it until the token is expired. When this attribute is **false**, for each SLF query NRF generates a new oAuth2 token. **Note**: Operator must set this field to **true** after NRF is upgraded from previous version and the upgrade process is complete and successful.
Preferred Port from IPEndPoint	This attribute indicates if the ports defined in the ipEndpoints of the NfService must be used for routing where chosen routing parameter is NfProfile or NfService FQDN, or NfProfile IPv6Addresses or IPv4Addresses. If set to **ENABLED**, the port is picked from the ipEndpoints and used with the FQDN in the NfProfile or NfService or IPv6Addresses or IPv4Addresses in the NfProfile. If the port is not present in the ipEndPoints, then NRF will continue to use the scheme for determining the port. If set to **DISABLED**, the scheme is used for determining the port. If NfService.scheme is set to http, port 80 is used. If NfService.scheme is set to https, port 443 is used. **Note**: For port selection either from NfProfile or NfService, the first ipEndPoints configured with only the port (with no IPv4/Ipv6) address is used for port selection.
Preferred Routing Parameter	This attribute indicates the priority of the parameters Fqdn, Ipv4Address, and Ipv6Address are used for routing SLF requests. If the most preferred attribute is not present in the NfProfile, then the next available preferred attribute is selected for routing. **Note**: The highest preference is given for the routing parameter to which the lowest priority value is set.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-39 (Cont.) SLF Options

Field Name	Description
SLF Configuration Mode	This attribute decides whether the SLF lookup can be performed based on preconfigured **SLF Host Config** configuration or SLF or UDR registered at the NRF. STATIC_SLF_CONFIG_MODE: If this value is set, the SLF lookup is performed based on preconfigured **SLF Host Config**. DISCOVERED_SLF_CONFIG_MODE: If this value is set, the SLF lookup is performed based on SLF or UDR registered with NRF. **Note**: The **SLF Host Config** must be configured before or while setting the **SLF Configuration Mode** as STATIC_SLF_CONFIG_MODE when **Feature Status** is ENABLED. Once the **Feature Status** is ENABLED, and the **SLF Configuration Mode** is set to STATIC_SLF_CONFIG_MODE, the **SLF Host Config** cannot be empty. If **SLF Configuration Mode** is set to DISCOVERED_SLF_CONFIG_MODE, **SLF Host Config** is not considered for discovery query. The **SLF Configuration Mode** can be set to DISCOVERED_SLF_CONFIG_MODE only if there is atleast one slfCandidate present in the **SLF Discovered Candidate List**. To trigger the population of **SLF Discovered Candidate List**, **Populate SLF Discovered Candidate List** must be set to **true**.
Maximum SLF Attempts	Indicates the maximum SLF attempts that can be made when the SLF request fails. **Note**: The value of this parameter must be less than or equal to the value configured for **Maximum Hop Count** parameter in General Options.
Use Alternate SCP	Indicates whether the reroute for the SLF query can be performed using alternate SCP. If the value is set to **true**, the alternate SCP that is configured under the Egress Gateway route configuration is used for routing the SLF query. If the value is set to **false** the alternate SCP is not used for routing. **Note**: • When **Use Alternate SCP** is set to **true**, alternate SCP selection will not guaranteed if SCP Health APIs feature is enabled. Hence, it is recommended to disable SCP Health APIs feature when **Use Alternate SCP** is set to **true**. • When **Use Alternate SCP** is set to **true**, **Custom Peer Selector Enabled** parameter in Routes Configuration must also be set to **true**.
Preferred SLF Locality	Indicates the preferred locality for SLF or UDR. When this attribute is configured, the SLF or UDR profile belonging to this locality is given the highest priority.
SLF Lookup Configuration	This attribute defines details for SLF lookup. Different exception lists, preferredSubscriberIdType, and SLF lookup skip can be configured per NFType. While enabling the SLF feature, this attribute is configured before or while enabling the feature.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-39 (Cont.) SLF Options

Field Name	Description
SLF Host Config	This is used to configure Primary and Secondary SLF details which are used for forwarding various requests. It allows to configure details of SLF like apiVersion, scheme, host, port, etc. The only supported value for apiVersionInUri is v1. Hence the apiVersions attribute must have at least one data record with apiVersionInUri attribute values set as v1. This configuration allows you to configure more than two SLF details. SLF with highest priority is considered as Primary SLF for forwarding messages. SLF with second highest priority is considered as Secondary SLF for forwarding. If supportedNfTypeList is set, then operator must set this attribute. This is because this value will be used to contact the network function hosting the SLF. To reset this attribute, send an empty array, for example: "slfHostConfig": [ ] If this attribute is already set, then there is no need to provide the value again. **Note**: The value of this attribute can be FQDN, IPv4, or IPv6. This attribute is used only if **SLF Configuration Mode** is set to STATIC_SLF_CONFIG_MODE.
Reroute On Response HTTP Status Codes	This attribute is used after getting a response from primary SLF (SLF Config with the highest priority), if the response code from primary SLF is present or matches this configuration, then NRF reroutes the SLF query to secondary SLF (SLF Config with second highest priority).
Populate SLF Discovered Candidate List	This attribute triggers the creation of the **SLF Discovered Candidate List** when set to **true**. **Note**: If **SLF Configuration Mode** is set to DISCOVERED_SLF_CONFIG_MODE, this attribute cannot be set back to **false**.
SLF Discovered Candidate List	This attribute contains the list of SLF or UDR profiles registered with the NRF which is used to send the SLF query when the SLF feature is enabled and **SLF Configuration Mode** is set to DISCOVERED_SLF_CONFIG_MODE. **Note**: This is a read-only attribute.
Error Responses	This attribute defines the error responses which may be sent during SLF processing. This attribute will allow to update the error response code and error response description for preloaded error conditions.

4. Configure **Preferred Port from IPEndPoint** and **Preferred Routing Parameter** option.
5. Click **Add** provided under **Preferred Routing Parameter** section to add the routing parameter details.
   The **Preferred Routing Parameter** page is displayed.
6. Enter the values for **Preferred Routing Parameter** as described.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-40 Preferred Routing Parameter

Parameter	Description	Details
Priority	Indicates the priority for the routing parameter. **Note**: No duplicate priority values are allowed.	**DataType**: string **Constraints**: 1, 2, 3 **Default Value**: NA
Routing Parameter	Indicates the routing parameter assigned for the specific priority. **Note**: Routing parameter value should be either Ipv4Address, Ipv6Address or Fqdn.	**DataType**: integer **Constraints**: Ipv4Address, Ipv6Address, Fqdn **Default Value**: NA

7. Click **Save** on **Preferred Routing Parameter** page.
8. Click **Add** SLF lookup configuration under **SLF Lookup Configuration** section.
9. Enter the values for SLF lookup configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Table 5-41 SLF Lookup Configuration

Field	Description
NF Type	NF Type for which SLF need to be supported.
Preferred Subscriber Id Type	This attribute is only used to pick one Subscriber Identifier if more than one subscriber identifiers (SUPI, GPSI) are present in the NFDiscover service operation message.
Exception List For Missing Mandatory Parameter	This attribute indicates if the mandatory attributes (SUPI or GPSI) are present in the discovery query and if any of the discovery query parameter matches with the value configured in this attribute, NRF continues to perform SLF Lookup. If the mandatory attributes (SUPI or GPSI) are not present in the discovery query parameters and if NFDiscover search query attribute is not present in the configured **Exception List For Missing Mandatory Parameter** attribute then the discovery query is rejected. If any of the discovery query parameter matches with the value configured in this attribute, NRF will not reject the request and continues to process the discovery request based on the other query parameters.
Skip SLF Lookup Parameter List	If the discovery query parameter matches with any of the values configured in this attribute, SLF lookup is to be skipped. In this case, NRF ignores the SUPI or GPSI if present in the discovery search query and processes the discovery query based on the remaining query attributes. **Note:** • NRF does not check if the mandatory parameters (SUPI or GPSI) are present in the discovery query if any of its parameters matches with this attribute. • If SUPI or GPSI is configured in **Skip SLF Lookup Parameter List**, NRF will not skip SLF lookup for the discovery request. • If the **Enable Value Based Skip SLF Lookup** parameter is set as true, NRF ignores the **Skip SLF Lookup Parameter List** parameter and proceeds with the value based skip SLF lookup.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-41 (Cont.) SLF Lookup Configuration

Field	Description
Value Based Skip SLF Lookup Parameters	This attribute configures the discovery parameter name and their corresponding values for which SLF lookup is to be skipped. The entries in the **Value Based Skip SLF Lookup Parameters** are used only when **Enable Value Based Skip SLF Lookup** flag is set to true. **Note:** • NRF does not check if the mandatory parameters (SUPI or GPSI) are present in the discovery query if any of its parameters matches with this attribute. • If SUPI or GPSI is configured in Value Based Skip SLF Lookup Parameters, NRF will not skip SLF lookup for the discovery request.
Enable Value Based Skip SLF Lookup	This attribute is used to enable or disable the Discovery Parameter Value Based Skip SLFLookup feature. If the value of **Enable Value Based Skip SLF Lookup** parameter is set as true, NRF validates the discovery query parameters and the values received in the discovery request against the values configured in **Value Based Skip SLF Lookup Parameters** parameter to determine if the SLF lookup is to be skipped. If the value of **Enable Value Based Skip SLF Lookup** parameter is set as false, NRF falls back to the existing skip SLF lookup feature (**Skip SLF Lookup Parameter List**) based on the list of discovery parameters configured.

Table 5-42 Value Based Skip SLF Lookup Parameters List

Field	Description
Skip Parameter Name	This parameter is a valid discovery query parameter supported by NRF for which the SLF lookup is to be skipped. **Note**: In 25.1.100, Discovery Parameter Value Based Skip SLF Lookup feature supports only dnn based skipSlfLookup.
Skip Parameter Value	This is the value for the corresponding discovery parameter name for which the SLF lookup is skipped and this value can be configured as a regular expression. .* indicates all and NRF skips the SLF lookup for any discovery parameter value. **Note:** • The Parameter Value string cannot be empty or null. • Maximum length of string is 200 characters.

10. Click **Save** on **Add SLF Lookup Configuration** page.
11. In **Edit** mode, click **Add** HTTP status code information under **Reroute On Response HTTP Status Codes** section.
12. Enter the values for HTTP status codes as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
13. Click **Add** provided under **SLF Host Config** section to add the host information.
14. Enter the values for SLF Host configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

15. Click **Save** on **Add SLF Host Config** page.
16. Set the value to *true* for **Populate SLF Discovered Candidate List** from the drop-down list to create the **SLF Discovered Candidate List**.
17. Click **Add** provided under **Error Responses** section to add the error response details. The **Add Error Responses** page is displayed.
18. Enter the values for **Error Responses** as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
19. Click **Save** on **Add Error Responses** page.
20. Click **Save** to save the SLF Options.

> **Note**
>
> Use the **Edit** or **Delete** icons available in the next column of the specific entry to update or delete the **SLF Lookup Configuration**, **Reroute On Response HTTP Status Codes**, **SLF Host Config**, and **Error Responses**.

5.3.8 Roaming Options

Perform the following procedure to configure the **Roaming Options**:

1. From the left navigation menu, navigate to **NRF** and then click **Roaming Options**. The **Roaming Options** page is displayed.
2. Click **Edit** from the top right side to edit or update **Roaming Options** parameter. The page is enabled for modification.
3. Configure **Roaming Options** fields as described in the following table:

Table 5-43 Roaming Options

Field Name	Description
Feature Status	Flag to control roaming feature. If value is ENABLED, roaming specific routing occurs. If value is DISABLED, roaming specific routing not occurs.
Generic InterPlmn FQDN	Generic FQDN required to be added in Inter-PLMN headers request. Every NRF, whether it is vNRF or hNRF, adds the genericInterPlmnFqdn in the header. This helps to detect the loop. This is a mandatory attribute. **Note**: It is mandatory to configure this FQDN value in inter-PLMN format. Example: nrf.5gc.mnc012.mcc345.pub.3gppnetwork.org
User Agent Mandatory	This flag regulates the consideration to relax or reject the request, if the user-agent is not present in the request at home NRF for Access Token Request Validation feature processing. **Note**: User-agent header is only applicable when the nfType attribute of the access token request is not present at home NRF.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-43 (Cont.) Roaming Options

Field Name	Description
Notification API Version	Required as part of 3gpp-Sbi-Callback header input which specifies the version of Notification Server. This is a mandatory attribute for Roaming feature. Example: v1
3GPP API Root Scheme	This attribute defines the scheme which is used while constructing the 3GPP API root header.
Error Responses	This attribute defines the error responses which may be sent during NRF Roaming traffic processing. This attribute allows the user to configure the error response code and error response description for preloaded error conditions.

4. Click **Add** under **Error Responses** section to add the error response details. The **Add Error Responses** page is displayed.
5. Enter the values for **Error Responses** as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
6. Click **Save** on **Add Error Responses** page.
7. Click **Save** to save the **Roaming Options**.

> **Note**
>
> Use the **Edit** or **Delete** icons available in the next column of the specific entry to update or delete the **Error Responses** information.

5.3.9 Georedundancy Options

Perform the following procedure to configure the **Georedundancy Options**:

1. From the left navigation menu, navigate to **NRF** and then click **Georedundancy Options**. The **Georedundancy Options** page is displayed.
2. Click **Edit** from the top right side to edit or update **Georedundancy Options** parameter. The page is enabled for modification.
3. Configure **Georedundancy Options** fields as described in the following table:

Table 5-44 Georedundancy Options

Field Name	Description
Feature Status	Enables or disables the georedundancy feature in OCNRF.
Replication DownTime Tolerance	This attribute specifies the minimum time the reported replication status shall remain as INACTIVE before OCNRF considers the replication status as INACTIVE.
Replication UpTime Tolerance	This attribute specifies the minimum time the reported replication status shall remain as ACTIVE before OCNRF considers the replication status as ACTIVE.
Replication Latency Threshold	This attribute defines the maximum allowed replication latency beyond which the replication channel shall be considered INACTIVE.
Replication Latency	The default replication Latency of the replication channel.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-44 (Cont.) Georedundancy Options

Field Name	Description
Replication Status Uri	The URI that is used to query the DB replication status.
Use Remote Data When Replication Down	This attribute specifies whether the remote NRF records are considered for OCNRF service operations when the DB replication is down. **Note**: This attribute does not have impact the **read-only** service requests. The service requests in this category are NfDiscovery, NfAccessToken, NfProfileRetrieval, and NfListRetrieval. When replication channel status is down, the mate NRF records are not used for processing the service requests.
Monitor NRF Service Status Interval	This attribute defines the time interval for monitoring the aggregated Nf_Management service status (combined status of nfRegistration, nfSubscription and nrfAuditor service).
Monitor DB Replication Status Interval	This attribute defines the time interval for monitoring the DB replication status.
Site Name To NRF InstanceId Mapping List	It contains an array of SiteNameToNrfInstanceIdMapping.

4. Enter the values for rules configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
5. Click **Add** provided under **Site Name To NRF InstanceId Mapping List** section to add the site ID details.

Table 5-45 Add Site Name To NRF InstanceId Mapping List

Attribute name	Description
Site Name	Represents site name
NRF Instance Id	Represents nrfInstanceId of NRF

6. Click **Save** on **Add Site Name To NRF InstanceId Mapping List** page.
7. Click **Save**.

5.3.10 NF Authentication Options

Perform the following procedure to configure the **NF Authentication Options**:

1. From the left navigation menu, navigate to **NRF** and then click **NF Authentication Options**.
   The **NF Authentication Options** page is displayed.
2. Click **Edit** from the top right side to edit or update **NF Authentication Options** parameter. The page is enabled for modification.
3. Configure **NF Authentication Options** fields as described in the following table:

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-46 NF Authentication Options

Field Name	Description
NF Registration Status	This attribute controls the authentication of consumer NF for NFRegister, NFUpdate and NFDeregister service operations. If value of this attribute is ENABLED, then identity of consumer NF is validated. If value of this attribute is DISABLED, then validation is not performed for consumer NF.
NF Subscription Status	This attribute controls the authentication of consumer NF for NFStatusSubscribe and NFStatusUnsubscribe service operations. If value of this attribute is ENABLED, then identity of consumer NF is validated and NRF allows the subscription only if the NF is registered with NRF. If value of this attribute is DISABLED, then validation is not performed for consumer NF.
NF Discovery Status	This attribute controls the authentication of consumer NF for NFDiscover service operation. If value of this attribute is ENABLED, then identity of consumer NF is validated. If value of this attribute is DISABLED, then validation is not performed for consumer NF. In case NF identity is not present in discovery request messages then validation is performed as per checkIfNfIsRegistered attribute.
Access Token Status	This attribute controls the authentication of consumer NF for AccessToken service operation. If value of this attribute is ENABLED, then identity of consumer NF is validated. If value of this attribute is DISABLED, then validation is not performed for consumer NF.
NF Profile Retrieval Status	This attribute controls the authentication of consumer NF for NF Profile Retrieval service operation. If value of this attribute is ENABLED, then identity of consumer NF is validated. If value of this attribute is DISABLED, then validation is not performed for consumer NF.
NF List Retrieval Status	This attribute controls the authentication of consumer NF for NF List Retrieval service operation. If value of this attribute is ENABLED, then identity of consumer NF is validated. If value of this attribute is DISABLED, then validation is not performed for consumer NF.
Check if NF is registered	This attribute controls the mechanism if there is need to check if NF is registered or not with NRF. If value of this attribute is ENABLED, then validation is performed. If value of this attribute is DISABLED, then validation is not performed for consumer NF. • discovery request does not contain requester-nf-instance-fqdn and value of nfDiscoveryAuthenticationStatus is ENABLED.
errorResponses	This attribute defines the error responses which may be sent for NF Authentication scenarios. This attribute will allow to update the response code, error response description, retryAfter and redirectUrl for preloaded error conditions.

4. Click **Add** provided under **Error Responses** section to add the error response details.
5. Enter the values for rules configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
6. Click **Save** on **Add Error Responses** page.
7. Click **Save** to save the NF Authentication Options.

> **Note**
>
> Use the **Edit** or **Delete** icons available in the next column of the specific entry to update or delete the **Error Responses**.

Copyright © 2019, 2025, Oracle and/or its affiliates.

5.3.11 Logging Level Options

Perform the following procedure to configure the **Logging Level Options**:

1. From the left navigation menu, navigate to **NRF** and then click **Logging Level Options**. The **Logging Level Options** page is displayed.
2. Click **Edit** from the top right side to edit or update **Logging Level Options** parameter. The page is enabled for modification.
3. Select the **Service Type** from the drop-down list to select anyone of the following NF Service type.
   a. **nfAccessToken**
   b. **nfDiscovery**
   c. **nfRegistration**
   d. **nfSubscription**
   e. **nrfAuditor**
   f. **nrfConfiguration**
   g. **nrfArtisan**
   h. **nrfCacheData**
   i. **ingressGateway**
   j. **egressGateway**
   k. **appInfo**
   l. **perfinfo**
   m. **alternateroute**

   For more information, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Table 5-47 Logging Level Options

Attribute	Description
nfAccessToken	Specifies the log level options for nfAccessToken microservice
nfDiscovery	Specifies the log level options for nfDiscovery microservice
nfRegistration	Specifies the log level options for nfRegistration microservice
nfSubscription	Specifies the log level options for nfSubscription microservice
nrfAuditor	Specifies the log level options for nrfAuditor microservice
nrfConfiguration	Specifies the log level options for nrfConfiguration microservice
nrfArtisan	Specifies the log level options for nrfArtisan microservice
nrfCacheData	Specifies the log level options for Cache Data Service
ingressGateway	Specifies the log level options for Ingress Gateway

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-47 (Cont.) Logging Level Options

Attribute	Description
egressGateway	Specifies the log level options for Egress Gateway
appInfo	Specifies the log level options for app info
perfinfo	Specifies the log level options for Perf Info
alternateroute	Specifies the log level options for alternate-route service

4. Select the **Application Log Level** from the dropdown menu to set the constraints of the log. Possible values are:
   - **TRACE**
   - **DEBUG**
   - **INFO**
   - **WARN**
   - **ERROR**

> **Note**
>
> The value for the **Application Log Level** field is the mandatory value, and the package log level is the optional value.

5. Select the appropriate **Log Rate Control Log Level** option from the drop-down menu, in the **Log Rate Control** section.

Table 5-48 Log Rate Control Options

Log Level	Description
ALL	No filtering; logs everything (not common for BurstFilter).
DEBUG	Filters logs at `DEBUG` and above.
ERROR	Applies burst filtering to error messages and above.
FATAL	Only filters fatal errors.
INFO	Filters logs at `INFO`, `WARN`, `ERROR`, and `FATAL`.
OFF	Disables the logLevel.
TRACE	Filters everything, including the most detailed logs.
WARN	Applies filtering to warnings and above.

6. Set the value of **Rate**. This value indicates the number of log messages that are throttled per second.

> **Note**
>
> The **Log Rate Control** and **Rate** parameters are displayed only for NRF microservices.

7. Set the status of **Additional Error Logging** as **ENABLED** from the drop-down menu to add the additional attributes in the ERROR log.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> This attribute is available only for the following microservices:
>
> - nfregistration
> - nfsubscription
> - nfdiscovery
> - nfaccesstoken
> - nrfconfiguration
> - nrfauditor
> - nrfcachedata
> - nrfartisan
> - ingressGateway
> - egressGateway
> - alternateroute

8. Set the status of **Log Subscriber Info** as **ENABLED** from the drop-down menu to add the subscriberId details in the ERROR log.

> **Note**
>
> This attribute is available only for the following microservices:
>
> - nfregistration
> - nfsubscription
> - nfdiscovery
> - nfaccesstoken
> - nrfconfiguration
> - nrfauditor
> - nrfcachedata
> - nrfartisan
> - ingressGateway
> - egressGateway
> - alternateroute

9. Click **Add** provided under **Package Log Level** section to specify the name of the package and its log level.

> **Note**
>
> This section is only applicable when Oracle Engineering is trying to isolate an issue and requests one or more package names be added and logs collected after the reproduction of an issue.

Copyright © 2019, 2025, Oracle and/or its affiliates.

a. Select the **Package Name** in the **Package Log Level** section. Possible values are:
   - **root**
   - **cache**

   > **Note**
   >
   > **cache** is for nfDiscovery microservice only.

b. From the **Log Level** drop-down list, select the log level for the package. Possible values are:
   - **TRACE**
   - **DEBUG**
   - **INFO**
   - **WARN**
   - **ERROR**

c. Click **Save** on **Add Package Log Level** page.
   The package log level information for the selected service is saved.

10. Click **Save** to save the **Logging Level Options**.

> **Note**
>
> Use the **Edit** or **Delete** icons available in the next column of the specific entry to update or delete the **Package Log Level**.

5.3.12 NF Screening Options

Perform the following procedure to configure the **NF Screening Options**:

1. From the left navigation menu, navigate to **NRF** and then click **NF Screening Options**. The **NF Screening Options** page is displayed.
2. Click **Edit** from the top right side to edit or update **NF Screening Options** parameter. The page is enabled for modification.
3. Configure the values for the attributes as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*:
   a. Enter the **Response Code** value in the field.
   b. Select the **Feature Status** from the drop-down list.
4. Click **Save**.

5.3.12.1 Screening Rules

NF Screening supports the functionality to screen the service requests received from 5G Network Functions (NFs) before allowing access to NRF services. In this feature, NRF screens the incoming service operations from NFs on the basis of some attributes against set of rules configured at NRF. NRF processes the required services only if screening is successful. This feature provides extra security by restricting the NF that can use the service of NRF.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Using the screening lists, operator can decide which NF can access the services provided by NRF by configuring attributes based on the requirement.

5.3.12.1.1 CALLBACK URI

Screening list type for callback URIs in NF Service and nfStatusNotificationUri in SubscriptionData.

NRF screens the callback URI present in the request before allowing access to management service. Host present in callback URI (FQDN+port or IP+port) must be used for screening. In CALLBACK URI, the attributes that can be modified are FQDN, Port and IP address.

5.3.12.1.1.1 Configuring Callback URI Parameters

Perform the following procedure to configure the **CALLBACK URI**:

1. From the left navigation menu, navigate to **NRF** and then click **Screening Rules**. Under **Screening Rules**, click **CALLBACK URI**.
   The **CALLBACK URI** page is displayed.
2. Click **Edit** from the top right side to edit or update a CALLBACK URI parameter.
   The page is enabled for modification.
3. Set the values for the fields as follows:
   a. Select **Status** from drop-down list.
   b. Select the required **NF Screening Type** from drop-down list.
   c. Choose the **Failure Action** for specific *Rules Data* from drop-down list. For more information on NF Types, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
   d. Click **Add** provided under **NF Callback URIs** section based on the requirement for each *Rules Data* to add NF Callback URIs.
   e. Enter the parameter values in **Add NF Callback URIs** page as described in the following table:

Table 5-49 Add NF Callback URIs

Attribute Name	Description
FQDN	Exact FQDN to be matched. This is conditional, at least one attribute is present.
Pattern	Regular Expression for FQDN. This is conditional, at least one attribute is present.
IPv4 Address	Enter the IPv4 address to be matched.
IPv4 Address Range	Enter the range of IPv4 addresses.
IPv6 Address	Enter the IPv6 address to be matched.
IPv6 Address Range	Enter the range of IPv6 addresses.
Ports	Enter the Ports. If this attribute is not configured then it will not be considered for validation
Port Ranges	Enter the range of ports. If this attribute is not configured then it will not be considered for validation

For more information on parameter values and description, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

f. Click **Save** on **Add NF Callback URIs** page to save the **NF Callback URIs** details.

4. Click **Save**.

> **Note**
>
> Use the **Edit** or **Delete** icons available in the next column of the specific entry to update or delete the **NF Callback URIs**.

5.3.12.1.2 NF FQDN

NRF screens the Fully Qualified Domain Name (FQDN) present in the request before allowing access to management service.

In NF FQDN, the attributes that can be modified are pattern, fqdn in NFProfile and fqdn in NFService.

5.3.12.1.2.1 Configuring NF FQDN Parameters

Perform the following procedure to configure the **NF FQDN**:

1. From the left navigation menu, navigate to **NRF** and then click **Screening Rules**. Under **Screening Rules**, click **NF FQDN**.
   The **NF FQDN** page is displayed.
2. Click **Edit** from the top right side to edit or update the NF FQDN parameter.
   The page is enabled for modification.
3. Set the values for the fields as follows:
   a. Select **Status** from drop-down list.
   b. Select the required **NF Screening Type** from drop-down list.
   c. Choose the **Failure Action** for specific *Rules Data* from drop-down list.
   d. Enter the **Pattern** and **FQDN** values under **NF FQDN** section based on the requirement for each *Rules Data*.
4. Click **Save**.

> **Note**
>
> Repeat the above steps for all the **Rules Data**.

5.3.12.1.3 NF IP Endpoint

NRF screens the IP endpoint(s) present in the request before allowing access to management service.

5.3.12.1.3.1 Configuring NF IP Endpoint Parameters

Perform the following procedure to configure the **NF IP Endpoint**:

1. From the left navigation menu, navigate to **NRF** and then click **Screening Rules**. Under **Screening Rules**, click **NF IP Endpoint**.
   The **NF IP Endpoint** page is displayed.

Copyright © 2019, 2025, Oracle and/or its affiliates.

2. Click **Edit** from the top right side to edit or update NF IP Endpoint parameters.
   The page is enabled for modification.
3. Set the values for the fields as follows:
   a. Select **Status** from drop-down list.
   b. Select the required **NF Screening Type** from drop-down list.
   c. Choose the **Failure Action** for specific *Rules Data* from drop-down list.
   d. Click **Add** provided under **NF IP Endpoint** section based on the requirement for each *Rules Data* to add NF IP Endpoint.
   e. Enter the parameter values in **Add NF IP Endpoint** page as described in the following table:

Table 5-50 Add NF IP Endpoint

Attribute Name	Description
IPv4 Address	IPv4 address to be matched. See NOTE below
IPv4 Address Range	Range of IPv4 addresses. See NOTE below
IPv6 Address	IPv6 address to be matched. See NOTE below
IPv6 Address Range	Range of IPv6 addresses. See NOTE below
port	If this attribute is not configured then it will not be considered for validation
portRange	If this attribute is not configured then it will not be considered for validation

   For more information on parameter values and description, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

   f. Click **Save** on **Add NF IP Endpoint** page to save the **NF IP Endpoint** details.
4. Click **Save**.

> **Note**
>
> Use the **Edit** or **Delete** icons available in the next column of the specific entry to update or delete the **NF IP Endpoint**.

5.3.12.1.4 NF Type Register

NRF screens the NF type present in the incoming service request.

5.3.12.1.4.1 Configuring NF IP Type Register Parameters

Perform the following procedure to configure the **NF IP Type Register**:

1. From the left navigation menu, navigate to **NRF** and then click **Screening Rules**. Under **Screening Rules**, click **NF IP Type Register**.
   The **NF IP Type Register** page is displayed.
2. Click **Edit** from the top right side to edit or update a NF IP Type Register parameters.
   The page is enabled for modification.
3. Set the values for the fields as follows:

Copyright © 2019, 2025, Oracle and/or its affiliates.

a. Select **Status** from drop-down list.
b. Select the required **NF Screening Type** from drop-down list.
c. Select the **Failure Action** from drop-down list for *Global Screening Rules Data*.
d. Enter the NF Type in the **NF Type List** field for *Global Screening Rules Data*.

4. Click **Save**.

5.3.12.1.5 PLMN ID Parameters

NRF screens the PLMN ID present in the request before allowing access to management service.

5.3.12.1.5.1 Configuring PLMN ID Parameters

Perform the following procedure to configure the **PLMN ID**:

1. From the left navigation menu, navigate to **NRF** and then click **Screening Rules**. Under **Screening Rules**, click **PLMN ID**.
   The **PLMN ID** page is displayed.
2. Click **Edit** from the top right side to edit or update a PLMN ID parameters.
   The page is enabled for modification.
3. Set the values for the fields as follows:
   a. Select **Status** from drop-down list.
   b. Select the required **NF Screening Type** from drop-down list.
   c. Choose the **Failure Action** for specific *Rules Data* from drop-down list.
   d. Click **Add** provided under **PLMN List** section based on the requirement for each *Rules Data* to add PLMN List.
   e. Enter the **MCC** and **MNC** parameter values in **Add PLMN List** page.
      For more information on parameter values and description, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
   f. Click **Save** on **Add PLMN List** page to save the **PLMN List** details.
4. Click **Save**.

> **Note**
>
> Use the **Edit** or **Delete** icons available in the next column of the specific entry to update or delete the **PLMN List**.

5.3.13 DNS NAPTR Update Options

Perform the following procedure to configure the **DNS NAPTR Update Options**:

1. From the left navigation menu, navigate to **NRF** and then click **DNS NAPTR Update Options**.
2. Click **DNS NAPTR Update Options** under **DNS NAPTR Update**.
   The **DNS NAPTR Update Options** page is displayed.

Copyright © 2019, 2025, Oracle and/or its affiliates.

3. Click **Edit** from the top right side to edit or update **DNS NAPTR Update Options** parameter.
   The page is enabled for modification.
4. Configure **DNS NAPTR Update Options** fields as described in the following table:

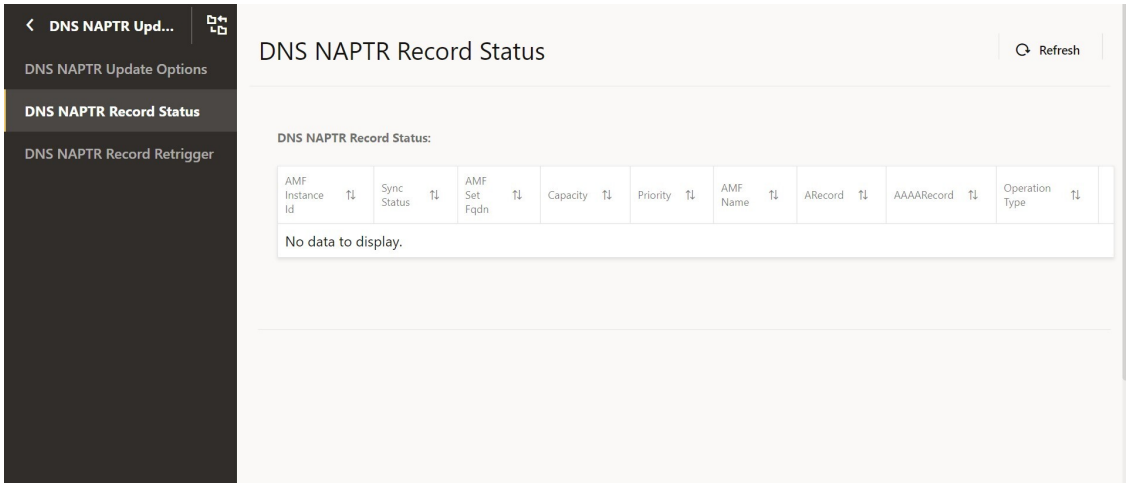
Table 5-51 DNS NAPTR Update Options

Field Name	Description
Feature Status	Enables or disables the DNS NATPR update. If the value of this attribute is ENABLED, then NRF performs DNS NAPTR update during AMF NFRegister, NFUpdate, and NFDeregister requests. If the value of this attribute is DISABLED, then NRF does not perform any DNS NAPTR update during AMF NFRegister, NFUpdate, and NFDeregister requests.
Max Retry Count	This attribute defines the maximum number of retries when a DNS NAPTR update fails.
Default Priority	The value of this attribute is considered as NF priority in case NFProfile does not have a priority attribute.
Default Capacity	The value of this attribute is considered as NF capacity in case NFProfile does not have the capacity attribute.

5.3.13.1 DNS NAPTR Record Status

This screen displays the status of DNS NAPTR record.

Figure 5-6 DNS NAPTR Record Status



Click on **View** icon to view the status of specific DNS NAPTR Record.

5.3.13.2 DNS NAPTR Record Retrigger

Perform the following procedure to retrigger NFInstanceIds using **DNS NAPTR Record Retrigger** option.

1. From the left navigation menu, navigate to **DNS NAPTR Record Retrigger**.
   The **DNS NAPTR Record Retrigger** page is displayed.

Copyright © 2019, 2025, Oracle and/or its affiliates.

2. Click **Edit** from the top right side to edit or update **DNS NAPTR Record Retrigger** parameter.
   The page is enabled for modification.
3. Enter the `NFInstanceIds` in **Retrigger NAPTR Update Records** field.
4. Click **Save**.

5.3.14 Pod Protection Options

Perform the following procedure to configure the **Pod Protection Options**:

1. From the left navigation menu, navigate to **NRF** and then click **Pod Protection Options**.
   The NF Subscription Pod Protection Options page is displayed.
2. Click **Edit** from the top right side to edit or update Pod Protection Options.
   The page is enabled for modification.
3. Set the value to `true` for **Enabled** option from the drop-down to enable the Pod Protection Options.
4. Set the value to `true` from the drop-down for **Enabled** option under Congestion Control section to enable the congestion control.
5. Click **Save** to save the Pod Protection Options.

Table 5-52 Pod Protection Options

Attribute	Description	Details
Enabled	This attribute indicates if the Pod Protection feature is enabled or disabled.	**DataType**: Boolean **Constraints**: true, false **Default Value**: false
Monitoring Interval	This attribute indicates the periodicity at which the overload state is monitored. **Unit**: Milliseconds **Note:** This is a read-only attribute.	**DataType**: Integer **Constraints**: NA **Default Value**: 1200
Congestion Control	This attribute allows the congestion control configuration. For more information about congestion control parameters, see Table 5-53.	**DataType**: Object **Constraints**: NA **Default Value**: NA

Table 5-53 CongestionControlConfig

Attribute	Description	Details
Enabled	This attribute allows the configuration of pod protection attributes for the nfSubscription pods. **Note**: This must be set to true for Pod Protection feature.	**DataType**: Boolean **Constraints**: true, false **Default Value**: false

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-53 (Cont.) CongestionControlConfig

Attribute	Description	Details
Action Sampling Period	This attribute indicates the interval at which the configured action must be considered. The actions are configured under entryAction.action attribute. The interval is calculated as (`actionSamplingPeriod * monitoringInterval`). **Note:** This is a read-only attribute.	**DataType**: Integer **Constraints**: NA **Default Value**: 2
State Change Sample Count	This attribute indicates the number of times the pod must be in the particular congestion state before transitioning to that state. For example, if the current state is normal, and the new state is DoC, then NRF moves the pod to DoC only if the state is reported for 10 times in 1 second (`stateChangeSampleCount * monitoringInterval`). **Note:** This is a read-only attribute.	**DataType**: Integer **Constraints**: NA **Default Value**: 2
States	This attribute indicates the congestion states, the thresholds, and corresponding actions. For more information about congestion states, see Table 5-54.	**DataType**: Object **Constraints**: NA **Default Value**: NA

Table 5-54 CongestionStates

Attribute	Description	Details
Name	The name of the congestion state. • Normal: The pod is not in overload state. • Danger of Congestion (Doc): The pod is about to go into the congested state. Actions configured in the entryAction is performed. • Congested state: The pod is in congested state. Actions configured in the entryAction is performed. **Note:** This is a read-only attribute.	**DataType**: String **Constraints**: Normal, Doc, Congested **Default Value**: NA
Weight	The weight of the congestion state. The weight indicates the criticality of the congestion state. The lower the value, the lower the criticality. **Note:** This is a read-only attribute.	**DataType**: Integer **Constraints**: NA **Default Value**: Normal= 0, DoC= 1, and Congested= 2
Entry Action	This attribute indicates the actions for the congestion state. For more information about the entry action configuration, see Table 5-55. **Note:** This is a read-only attribute.	**DataType**: List **Constraints**: NA **Default Value**: NA

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-54 (Cont.) CongestionStates

Attribute	Description	Details
Resource Threshold	This attribute indicates the resource thresholds for the given congestion state. This configuration is mandatory for the 'DoC' and 'Congested' states. For more information about the threshold for each resources, see Table 5-56. **Note:** This is a read-only attribute.	**DataType**: Object **Constraints**: NA **Default Value**: NA

Table 5-55 EntryActionConfig

Attribute	Description	Details
Action	This attribute indicates the action for the congestion state. • AcceptIncomingConnections: The action indicates whether the incoming new connection is accepted or rejected based on the overload state. • MaxConcurrentStreamsUpdate: The action indicates whether to increase or decrease the max concurrent stream for all incoming connections till the maxConcurrentStreamsPerCon is reached.	**DataType**: String **Constraints**: MaxConcurrentStreamsUpdate, AcceptIncomingConnections **Default Value**: NA
Arguments	This attribute indicates the actions for the congestion state. For more information about the arguments, see Table 5-57.	**DataType**: Map <String, Object> **Constraints**: NA **Default Value**: NA

Table 5-56 ResourceThreshold

Attribute	Description	Details
CPU	The CPU threshold is expressed in percentage. **Note:** This is a read-only attribute.	**DataType**: Integer **Constraints**: NA **Default Value**: For DoC, the default CPU is 60. For Congested, the default CPU is 75.
Pending Message Count	The number of messages pending to be processed, expressed in absolute value. **Note:** This is a read-only attribute.	**DataType**: Integer **Constraints**: NA **Default Value**: For DoC, the default count is 100. For Congested, the default count is 150.

Table 5-57 Possible Arguments

Attribute	Description	Details
Accept	The attribute indicates if the incoming connection should be accepted or not. Applicable when the action is **Accept Incoming Connections**. True: The incoming connection is accepted. False: The incoming connection is rejected.	**DataType**: Boolean **Constraints**: true, false **Default Value**: NA

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-57 (Cont.) Possible Arguments

Attribute	Description	Details
Increment By	The attribute indicates the factor by which the current concurrent streams value will be incremented till it reaches maxConcurrentStreamsPerCon. **Note**: This is preconfigured for Normal and DoC state.	**DataType**: Integer **Constraints**: NA **Default Value**: 30
Decrement By	The attribute indicates the factor by which the current concurrent streams value will be decremented till it reaches maxConcurrentStreamsPerCon. **Note**: This is preconfigured for DoC and Congested state.	**DataType**: Integer **Constraints**: NA **Default Value**: 30
Max Concurrent Streams Per Connection	The attribute indicates the maximum number of concurrent streams per connection.	**DataType**: Integer **Constraints**: NA **Default Value**: For Normal, the default value is 100. For DoC, the default count is 50. For Congested, the default count is 10.
Decrement By Action Sampling Period	The attribute indicates the time interval at which the decrementBy is applied to reach **Max Concurrent Streams Per Connection**. If not provided, the **Action Sampling Period** is used. **Note**: This is preconfigured for DoC and Congested state.	**DataType**: Integer **Constraints**: NA **Default Value**: 1
Increment By Action Sampling Period	The attribute indicates the time interval at which the incrementBy is applied to reach **Max Concurrent Streams Per Connection**. If not provided, the **Action Sampling Period** is used. **Note**: This is preconfigured for DoC and Congested state.	**DataType**: Integer **Constraints**: NA **Default Value**: 30

5.3.15 Controlled Shutdown

Perform the following procedure to configure the **Controlled Shutdown**:

1. From the left navigation menu, navigate to **NRF** and then click **Controlled Shutdown**. The **Controlled Shutdown Options** page is displayed.
2. Click Edit from the top right side to edit or update **Operational State** parameter. The page is enabled for modification.
3. Configure **Operational State** field as described in the following table:

Table 5-58 Operational State

Field Name	Description
Operational State	Determines the value of the current operational state. Allowed values are NORMAL, COMPLETE_SHUTDOWN.

4. Click on **Operational State History** field to view the last 5 operational state changes.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-59 Operational State History

Field Name	Description
Operational State History	Provides the list of the last five operational state changes.

5.3.16 NRF Growth Options

Perform the following steps to configure the NRF Growth feature.

1. From the left navigation menu, navigate to **NRF** and then click the **Growth Feature**.
2. Select **Feature Options** under **Growth Feature** in the left navigation menu.
   The **Feature Options** page is displayed.
3. Click **Edit** from the top right side to edit or update Growth Feature Options.
   The page is enabled for modification.
4. Set the value to **ENABLED** for the **Feature Status** option from the drop-down to enable the Growth feature.
5. Configure **NRF Host Config** details. Enter the values for host configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
   a. Click **Add** provided to add a NF set ID.
   b. Click **Save** on the **Add NF Set ID** page.
6. Configure the reroute profile. Enter the values for reroute configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.16.1 Forwarding Options

In case you want to enable Forwarding Options for the NRF Growth feature, perform the following configurations:

> **Note**
>
> When the NRF Growth feature is enabled, **NRF-NRF Forwarding Options** configurations is not considered. For more information about the forwarding options, see the "Forwarding Options" section.

1. From the left navigation menu, navigate to **NRF** and then click the **Growth Feature**.
2. Select **Forwarding Options** under **Growth Feature** in the left navigation menu.
   The **Forwarding Options** page is displayed.
3. Click **Edit** from the top right side to edit or update the **Forwarding Options** parameter.
   The page is enabled for modification.
4. Configure **Forwarding Options** fields as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
5. Configure **NRF Host Config** details. Enter the values for host configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

a. Click **Add** provided to add a segment ID.

b. Click **Save** on the **Add Segment ID** page.

6. Select the **Feature Status** value provided under the **Forwarding Rules Feature Config** section to set the feature configuration.
This flag can be **ENABLED** only if Discovery Status or AccessToken Status attributes are ENABLED.

7. Click **Add** provided under the **Forwarding Rules Feature Config** section to add the Forwarding Rules feature configuration.
The **Add Forwarding Rules Config** page is displayed.

a. Enter the values for rules configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

b. Click **Save** on the **Add Forwarding Rules Config** page.

8. Click **Add** provided under the Error Responses section to add the error response details.
The **Add Error Responses page** is displayed.

a. Enter the values for rules configuration as described in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

b. Click **Save** on the **Add Error Responses** page.

9. Click **Save** to save the **Forwarding Options**.

5.3.17 Ingress Gateway

This section describes the Ingress Gateway configurations.

5.3.17.1 CCA Header

This configuration is used to update the CCA header configuration. By default, configurableErrorCodes will have value as provided in the feature configuration, if the configuration is disabled then default values are considered from Helm file.

Perform the following configurations:

1. From the left navigation menu, navigate to **Ingress Gateway** option to configure the Ingress Gateway APIs.
2. Click the **CCA Header** option under **Ingress Gateway** to configure the CCA header.
3. The **CCA header** page is displayed with default values.
4. Click **Edit** to edit the default configuration.

   The **Edit CCA header** page is displayed.
5. Configure the fields in the **Edit CCA header** page as mentioned in Table 5-60.
6. Click **Save** on the **Edit CCA header** page to save the details. Click **Cancel** to discard your progress and go back to **CCA header** page.

> **Note**
>
> - Use **Refresh** icon to refresh the configuration.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-60 CCA Header Validation Configuration

Field Name	Description	Details
Enabled	Indicates if the CCA validation feature flag is enabled or disabled. **Note**: This is a read-only parameter. If you want to enable this feature, use `metadata.ccaHeaderValidation.enabled` parameter in routesConfig for `accesstoken_mapping` id in custom values file.	**Data Type**: boolean **Constraints**: true, false **Default Value:** false **Presence:** M
Min Expiry Time	Indicates the buffer or additional time that is added to the current time to check it with the certificate expiry time.	**Data Type**: integer **Constraints**: NA **Default Value:** 0 **Presence:** M
Max Token Age	Indicates the maximum token age allowed. **Note:** It skips the check if the value is 0.	**Data Type**: integer **Constraints**: 0 - 86400 seconds **Default Value:** 0 **Recommended Value:** 60 for NRF. **Presence:** M
Role	Indicates the CCA token validator role to match the audience claim in the header. **Note:** This is a read-only parameter.	**Data Type**: string **Constraints**: NA **Default Value:** NRF **Presence:** M
Sub Key	Indicates if the certificate extension name to be read in the public key certificate received in CCA request for consumer NFs Instance Id. For NRF this value should be NF Instance Id. **Note**: This is a read-only parameter.	**Data Type**: string **Constraints**: NA **Default Value:** subjectAltName **Presence:** M
Validation Rule	Indicates the CCA validation rule. • strict: If "Issued At"+"Maximum Token Age Allowed "> current time, then the NRF validation rules are applied. • relaxed: "Issued At" validation is not mandatory. Validation rules are applied to Producer NF. For NRF, this value is always strict.	**Data Type**: string **Constraints**: strict, relaxed **Default Value:** strict **Presence:** M
K8 Secret Name	Indicates the name of the Kubernetes secret in which the CA bundle is present. **Note**: After fresh installation, these attributes must be configured with appropriate values as per the system requirement as these values are sample ones. Example `k8SecretName: ocingress-secret`	**Data Type**: string **Constraints**: NA **Default Value:** NA **Presence:** M

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-60 (Cont.) CCA Header Validation Configuration

Field Name	Description	Details
K8 NameSpace	Indicates the Kubernetes namespace in which CA bundle is present. **Note**: `changens` value is updated with NRF secret for CCA header. **Note**: After fresh installation, these attributes must be configured with appropriate values as per the system requirement as these values are sample ones. Example `k8NameSpace: ocingress-ns`	**Data Type**: string **Constraints**: NA **Default Value:** NA **Presence:** M
File Name	Indicates the name of the CA bundle file used for CCA. This is the file generated by certificate and key generation steps. **Note**: After fresh installation, these attributes can be configured with appropriate values as per the system requirement as these values are sample ones. Example: `fileName: caroot.cer`	**Data Type**: string **Constraints**: NA **Default Value:** NA **Presence:** M

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.17.2 Controlled Shutdown Error Mapping

This configuration is used for mapping between the routes and the error code profile that is used when the Ingress Gateway rejects incoming requests.

Perform the following configurations:

1. From the left navigation menu, navigate to **Ingress Gateway** option to configure the Ingress Gateway APIs.
2. Click **Controlled Shutdown Error Mapping** option under **Ingress Gateway**.

   The **Controlled Shutdown Error Mapping** page is displayed.
3. Click **Edit** to edit the configuration.

   The **Edit Controlled Shutdown Error Mapping** page is displayed.
4. Click **Add** in the **Route Error Profile List** section.

   The **Add Route Error Profile List** page is displayed.
5. Configure the fields in the **Add Route Error Profile List** page as mentioned in Table 5-61.
6. Click **Save** on the **Edit Controlled Shutdown Error Mapping** page to save the details. Click **Cancel** to discard your progress and go back to **Controlled Shutdown Error Mapping** page.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> - Use the **Edit** or **Delete** icon available in the next column of the specific entry to update or delete the error profile information.
> - Use **Refresh** icon to refresh the error profile information.

Table 5-61 Controlled Shutdown Error Mapping

Field Name	Description	Details
Route ID	The route id that is configured in routes configuration.	**DataType**: string **Constraints**: NA **Possible values:** <subscription_mapping, accesstoken_mapping, registration_mapping, disc_mapping> The possible values should match with the routeId of the routes configuration. **Default Value:** NA
Error Profile Name	The error profile name that is used to fetch the error from errorcodeprofiles.	**DataType**: string **Constraints**: NA **Default Value:** shutdownerror503

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.17.3 Copy Header on Gateway Error

Perform the following procedure to configure the **Copy Header on Gateway Error**:

1. From the left navigation menu, navigate to **Ingress Gateway** option to configure the Ingress Gateway APIs.
2. Click **Copy Header on Gateway Error** option under **Ingress Gateway**.
   The **Copy Header on Gateway Error** page is displayed with default values for a few fields.
3. Click **Edit** to modify the attributes in the **Copy Header on Gateway Error** page.
   The **Edit Copy Header on Gateway Error** page is displayed.
4. Use the toggle option to **Enabled** to turn on the feature.
5. Add the required Request Header Names in the **Request Header Names** field.
6. Click **Save** on the **Edit Copy Header on Gateway Error** page to save the details.
7. Click **Cancel** to discard your progress and go back to **Copy Header on Gateway Error** page.

5.3.17.4 Error Code Profile

This configuration is used to update the errorcodeprofiles that is used in Overload Control, Controlled Shutdown feature for populating details in error responses when a request is discarded.

Perform the following configurations:

Copyright © 2019, 2025, Oracle and/or its affiliates.

1. From the left navigation menu, navigate to **Ingress Gateway** option to configure the Ingress Gateway APIs.
2. Click **Error Code Profiles** option under **Ingress Gateway**.

   The **Error Code Profiles** page is displayed with default values for few fields.
3. Click **Add** to add the profiles.

   The **Create Error Code Profiles** page is displayed.
4. Configure the fields in the **Create Error Code Profiles** page as mentioned in Table 5-62.
5. Click **Save** on the **Create Error Code Profiles** page to save the details. Click **Cancel** to discard your progress and go back to **Error Code Profiles** page.

> **Note**
>
> - Use the **Edit** icon available in the next column of the specific entry to update the route error profile information.
> - Use **Refresh** icon to refresh the route error profile information.

Table 5-62 Error Code Profile Configuration

Field Name	Description	Details
Name	Indicate the error name.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M
Error Code	**Error Code** field in an errorScenario determines the HttpStatusCode that needs to be populated in ProblemDetails (HttpStatus field) response from Ingress Gateway when the exception occurred at Ingress Gateway matches the configured errorScenario's exceptionType field.	**Data Type**: integer **Constraints**: NA **Default Value**: NA **Presence**: M
Error Cause	**Error Cause** field in an errorScenario determines the error cause that needs to be populated in ProblemDetails (Cause field) response from Ingress Gateway when the exception occurred at Ingress Gateway matches the configured errorScenario's exceptionType parameter.	**Data Type**: integer **Constraints**: NA **Default Value**: NA **Presence**: O
Error Title	**Error Title** field in an errorScenario determines the title that needs to be populated in ProblemDetails (Title field) response from Ingress Gateway when the exception occurred at Ingress Gateway matches the configured errorScenario's exceptionType parameter.	**Data Type**: integer **Constraints**: NA **Default Value**: NA **Presence**: O

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-62 (Cont.) Error Code Profile Configuration

Field Name	Description	Details
Redirect URL	**Redirect URL** field in an errorScenario determines the redirection URL, this value is populated in LOCATION header while sending response from Ingress Gateway. The header is populated only when the exception occurred at Ingress Gateway matches the configured errorScenario's exceptionType parameter, the errorCode configured for the particular errorScenario lies in 3xx error series and the redirectUrl field for the particular errorScenario is configured appropriately.	**Data Type**: integer **Constraints**: NA **Default Value**: NA **Presence**: O
Retry After	**Retry After** field in an errorScenario determines the value in seconds or particular date after which the service should be retried, this value is populated in Retry-After header while sending response from Ingress Gateway. The header is populated only when the exception occurred at Ingress Gateway matches the configured errorScenario's exceptionType parameter, the errorCode configured for the particular errorScenario lies in 3xx error series and the retry-after field for the particular errorScenario is configured appropriately in seconds.	**Data Type**: integer **Constraints**: NA **Default Value**: NA **Presence**: O
Error Description	**Error Description** field in an errorScenario determines the description that needs to be populated in ProblemDetails (Detail field) response from Ingress Gateway when the exception occurred at Ingress Gateway matches the configured errorScenario's exceptionType field.	**Data Type**: integer **Constraints**: NA **Default Value**: NA **Presence**: O

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.17.5 Overload Control Policy Mapping

This configuration is used to update service names and corresponding policy names for the service that is mapped to "ocDiscardPolicies" based on "policyName" and enable or disable the Overload Control feature and the sampling period in overload control. The Overload Control feature is disabled by default, and the sampling period is 200. To enable the feature, invoke REST API and update the enabled flag to true.

Perform the following configurations:

1. From the left navigation menu, navigate to **Ingress Gateway** option to configure the Ingress Gateway APIs.
2. Click the **Overload Control Policy Mapping** option under **Ingress Gateway** to configure peers.

   The **Overload Control Policy Mapping** page is displayed.
3. Click **Add** to add the configuration.

   The **Create Overload Control Policy Mapping** page is displayed.

Copyright © 2019, 2025, Oracle and/or its affiliates.

4. Configure the fields in the **Create Overload Control Policy Mapping** page as mentioned in Table 5-63.
5. Click **Add** in the **Mappings** section.

   The **Add Mappings** page is displayed.
6. Configure the fields under the **Add Mappings** page as mentioned in Table 5-64.
7. Click **Save** on the **Add Mappings** page to save the details. Click **Cancel** to discard your progress and go back to **Create Overload Control Policy Mapping** page.
8. Click **Save** on the **Create Overload Control Policy Mapping** page to save the details. Click **Cancel** to discard your progress and go back to **Overload Control Policy Mapping** page.

> **Note**
>
> - Use the **Edit** icon available in the next column of the specific entry to update the policy mapping information.
> - Use **Refresh** icon to refresh the policy mapping information.

Table 5-63 Policy Mapping Configuration

Field Name	Description	Details
Enabled	To enable or disable Overload Control feature.	**Data Type**: boolean **Constraints**: NA **Default Value**: NA **Presence**: M
Sampling Period	Time frame for each cycle of Overload Control per service. Its value is in milliseconds.	**Data Type**: integer **Constraints**: NA **Default Value**: NA **Presence**: M

Table 5-64 Mappings

Field Name	Description	Details
Service Name	The service entry to determine a mapping between service and discard policy name per service. **Service Name** must be added in the following format:<deployment-name>-<servicename> **Note**: servicename is fixed and cannot be changed.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M
Policy Name	The discard policy entry to determine a mapping between service and discard policy name per service.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

5.3.17.6 Overload Control Discard Policy

This configuration is used to update discard policies that are used in overload control to select the appropriate policy from the configured list based on the load level of a particular service. By default, `ocdiscardpolicies` is null.

Perform the following configurations:

1. From the left navigation menu, navigate to **Ingress Gateway** option to configure the Ingress Gateway APIs.
2. Click the **Overload Control Discard Policy** option under **Ingress Gateway** to configure peers.

   The **Overload Control Discard Policy** page is displayed.
3. Click **Add** to add the configuration.

   The **Create Overload Control Discard Policy** page is displayed.
4. Configure the fields in the **Create Overload Control Discard Policy** page as mentioned in Table 5-65.
5. Click **Add** in the **Policies** section.

   The **Add Policies** page is displayed.
6. Configure the fields under the **Add Policies** page as mentioned in Table 5-66.
7. Click **Save** on the **Add Policies** page to save the details. Click **Cancel** to discard your progress and go back to **Create Overload Control Discard Policy** page.
8. Click **Save** on the **Create Overload Control Discard Policy** page to save the details. Click **Cancel** to discard your progress and go back to **Overload Control Discard Policy** page.

> **Note**
>
> - Use the **Edit** icon available in the next column of the specific entry to update the policy information.
> - Use **Refresh** icon to refresh the policy information.

Table 5-65 Discard Policy Configuration

Attribute Name	Description	Details
Name	Name of the discarded policy. **Note**: name must be the value configured in policyName under ocpolicymapping.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M
Scheme	Discarded policy scheme based on percentage.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-66 Policies

Attribute Name	Description	Details
Value	Value of priority above which requests are considered as potential candidates for drop. Percentage of requests to drop in the current sampling period over the calculated rate in the previous sampling period.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M
Action	Defines the action to be taken on selected requests rejection based on error code.	**Data Type**: string The value can be: RejectWithErrorCode **Constraints**: NA **Default Value**: NA **Presence**: M
Level	Defines the overload level.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M
Error Code Profile	Define the error code profiles.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.17.7 Error Code Series

This configuration is used to map errorCodeSeries id and the corresponding errorCodeSeries configuration. By default this configuration is null.

Perform the following configurations:

1. From the left navigation menu, navigate to **Ingress Gateway** option to configure the Ingress Gateway APIs.
2. Click **Error Code Series** option under **Ingress Gateway**.

   The **Error Code Series** page is displayed.
3. Click **Add** to add the error code series.

   The **Create Error Code Series** page is displayed.
4. Configure the fields in the **Create Error Code Series** page as mentioned in Table 5-67.
5. Click **Add** in the **Error Code Series** section.

   The **Add Error Code Series** page is displayed.
6. Configure the fields in the **Add Error Code Series** page as mentioned in Table 5-68.
7. Click **Save** on the **Add Error Code Series** page to save the details. Click **Cancel** to discard your progress and go back to **Create Error Code Series** page.
8. Click **Save** on the **Create Error Code Series**page to save the details. Click **Cancel** to discard your progress and go back to **Error Code Series** page.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> - Use the **Edit** icon or **Delete** available in the next column of the specific entry to update or delete the error code series information.
> - Use **Refresh** icon to refresh the error code series information.

Table 5-67 Error Code Series

Field Name	Description	Details
ID	Indicates the error code identifier.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M
Exception List	Lists the configurable exception or error for an error scenario in Ingress Gateway. The only supported values are: ConnectionTimeout, RequestTimeout, UnknownHostException, ConnectException, RejectedExecutionException, InternalError and NotFoundException, ClosedChannelException, BlackListIpException	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M
Error Code Series	Lists the error codes for a specific service. **Note**: "ErrorCodeSeries" is configured only if a set of error responses with specific error codes is expected in server header. If it is not configured then all the error responses will have server header.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M

Table 5-68 Error Code Series Configuration

Field Name	Description	Details
Error Set	Possible values for "errorSet" attribute: 5xx, 4xx, 3xx, 2xx, 1xx	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M
Error Codes	Possible values include all error codes in the respective HttpSeries value assigned for "errorSet". **Note**: Use single value of "-1" if all error codes in that HttpSeries are to be considered.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.17.8 Routes Configuration

This configuration is used to configure the route ID and server header in Ingress Gateway.

Perform the following configurations:

Copyright © 2019, 2025, Oracle and/or its affiliates.

1. From the left navigation menu, navigate to **Ingress Gateway** option to configure the Ingress Gateway APIs.
2. Click **Routes Configuration** option under **Ingress Gateway**.

   The **Routes Configuration** page is displayed.
3. Click **Add** to add the configuration.

   The **Create Routes Configuration** page is displayed.
4. Configure the fields in the **Create Routes Configuration** page as mentioned in Table 5-69.
5. Click **Save** on the **Create Routes Configuration** page to save the details. Click **Cancel** to discard your progress and go back to **Routes Configuration** page.

Table 5-69 Routes Configuration

Attribute Name	Description	Details
ID	Value of "id" attribute defines a specific service for route configuration.	**Data Type**: String **Constraints**: NA **Default Value**: NA **Presence**: M
FailureReqCountErrorCodeSeriesId	Indicates the ID that is used to map the service with the error code ID defined in errorcodeserieslist.	**Data Type**: String **Constraints**: NA **Default Value**: NA **Presence**: M
serverHeaderDetails	Indicates if the server header feature is available at Routes configuration. For more information see Server Header Details. **Note:** If server header is enabled at RoutesConfiguration, the configuration details like nfType and nfInstanceId are retrieved from the Global level. If `errorCodeSeriesId` is not configured at Route level, `errorCodeSeriesId` configured at Global level will be used as a fallback when serverHeaderDetails at RouteLevel is set as true.	**Data Type**: Array[Server Header Details.] **Constraints**: NA **Default Value**:NA **Presence**: O

5.3.17.9 Pod Protection

This configuration is used to configure the pod protection for Ingress Gateway pod.

Perform the following configurations:

1. From the left navigation menu, navigate to **Ingress Gateway** option to configure the Ingress Gateway APIs.
2. Click the **Pod Protection** option under **Ingress Gateway**.

   The **Pod Protection** page is displayed.
3. Click **Edit** to edit the configuration.

   The **Edit Pod Protection** page is displayed.
4. Configure the fields in the **Edit Pod Protection** page as mentioned in Table 5-70.
5. Configure the fields under the **Congestion Control Configuration** section as mentioned in Table 5-71.

Copyright © 2019, 2025, Oracle and/or its affiliates.

6. Click **Add** in the **States** section to configure different states for Pod Protection.

   The **Add States** page is displayed.
7. Configure the fields in the **Add States** page as mentioned in Table 5-72.
8. Configure the fields in the **Congestion Control Threshold** section in the **Add States** page as mentioned in Table 5-74.
9. Click **Add** in the **Error Action** section the **Add States** page.

   The **Add Error Action** page is displayed.
10. Configure the fields in the **Add Error Action** page as mentioned in Table 5-73.
11. Click **Add** in the **Entry Action** section the **Add Error Action** page.

    The **Add Entry Action** page is displayed.
12. Configure the fields in the **Add Entry Action** page as mentioned in Table 5-75.
13. Click **Save** on the **Add Error Action**page to save the details. Click **Cancel** to discard your progress and go back to **Error Action**page.
14. Click **Save** on the **Add States** page to save the details. Click **Cancel** to discard your progress and go back to **Edit Pod Protection** page.
15. Click **Save** on the **Edit Pod Protection** page to save the details. Click **Cancel** to discard your progress and go back to **Pod Protection** page.

> **Note**
>
> - Use the **Edit** icon available in the next column of the specific entry to update or delete the states information.
> - Use **Refresh** icon to refresh the states information.

Table 5-70 Pod Protection Options

Field Name	Description	Details
Enabled	This attribute indicates if the Pod Protection feature is enabled or disabled.	**Data Type**: Boolean **Constraints**: true, false **Default Value**: false
Monitoring Interval	This attribute indicates the periodicity at which the overload state is monitored. **Unit**: Milliseconds **Note:** The proposed value for this attribute is 1200.	**Data Type**: Integer **Constraints**: NA **Default Value**: 1200

Table 5-71 Congestion Control Configurations

Field Name	Description	Details
Enabled	This attribute allows the configuration of pod protection attributes for the Ingress Gateway pods. **Note**: This must be set to `true` for Pod Protection feature.	**Data Type**: boolean **Constraints**: true, false **Default Value**: false

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-71 (Cont.) Congestion Control Configurations

Field Name	Description	Details
Action Sampling Period	This attribute indicates the interval at which the configured action must be considered. The actions are configured under `entryAction.action` attribute. The interval is calculated as (`actionSamplingPeriod` * `monitoringInterval`). **Note:** The proposed value for this attribute is 3.	**Data Type**: integer **Constraints**: NA **Default Value**: NA
State Change Sample Count	This attribute indicates the number of times the pod must be in the particular congestion state before transitioning to another state. For example, if the current state is normal, and the new state is DoC, then NRF moves the pod to DoC only if the state is reported for 10 times in 1 second (`stateChangeSampleCount` * `monitoringInterval`). **Note:** The proposed value for this attribute is 10.	**Data Type**: integer **Constraints**: NA **Default Value**: NA

Table 5-72 States

Field Name	Description	Details
Name	The name of the congestion state. • Normal: The pod is not in overload state. • Danger of Congestion (Doc): The pod is about to go into the congested state. Actions configured in the entryAction is performed. • Congested state: The pod is in congested state. Actions configured in the entryAction is performed.	**Data Type**: string **Constraints**: Normal, Doc, Congested **Default Value**: NA
Weight	The weight of the congestion state. The weight indicates the critical of the congestion state. The lower the value, the lower the criticality. **Note:** The proposed value for this attribute is for Normal is 0, DoC is 1, and Congested is 2.	**Data Type**: integer **Constraints**: NA **Default Value**: NA

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-73 Entry Action

Field Name	Description	Details
Action	This attribute indicates the action for the congestion state. • `AcceptIncomingConnections`: The action indicates whether the incoming new connection is accepted or rejected based on the overload state. • `MaxConcurrentStreamsUpdate`: The action indicates whether to increase or decrease the max concurrent stream for all incoming connections till the maxConcurrentStreamsPerCon is reached.	**DataType**: string **Constraints**: MaxConcurrentStreamsUpdate, AcceptIncomingConnections **Default Value**: NA

Table 5-74 Resource Threshold

Field Name	Description	Details
CPU	The CPU threshold is expressed in percentage. **Note:**: The recommended value for this attribute is for DoC is 75 and for Congested is 85.	**Data Type**: integer **Constraints**: NA **Default Value**: NA
Pending Message	The number of pending messages to be processed, expressed in absolute count. **Note:**: The recommended value for this attribute is for DoC is 1500 and for Congested is 2000.	**Data Type**: integer **Constraints**: NA **Default Value**: NA

Table 5-75 Possible Arguments

Field Name	Description	Details
Accept	The attribute indicates if the incoming connection should be accepted or not. Applicable when the action is `AcceptIncomingConnections`. true: The incoming connection is accepted. false: The incoming connection is rejected. The proposed value for Normal state is `true`, and for DoC and Congested state is `false`.	**Data Type**: boolean **Constraints**: true, false **Default Value**: NA
Increment by	The attribute indicates the factor by which the current concurrent streams value will be incremented till it reaches `maxConcurrentStreamsPerCon`. **Note**: The proposed value for this attribute is 30 for Normal and DoC state.	**Data Type**: integer **Constraints**: NA **Default Value**: NA
Decrement by	The attribute indicates the factor by which the current concurrent streams value will be decremented till it reaches `maxConcurrentStreamsPerCon`. **Note**: The proposed value for this attribute is 30 for DoC and Congested state.	**Data Type**: integer **Constraints**: NA **Default Value**: NA

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-75 (Cont.) Possible Arguments

Field Name	Description	Details
Max Concurrent Streams Per Con	The attribute indicates the maximum number of concurrent streams per connection allowed. **Note**: The proposed value for this attribute is for Normal is 100, DoC is 10, and Congested is 1.	**Data Type**: integer **Constraints**: NA **Default Value**: NA
Decrement by Action Sampling Period	The attribute indicates the time interval at which the decrementBy is applied to reach `maxConcurrentStreamsPerCon`. If not provided, the `actionSamplingPeriod` is used. **Note**: The proposed value for this attribute is 1 for DoC and Congested state. Unit is seconds.	**Data Type**: integer **Constraints**: NA **Default Value**: NA
Increment by Action Sampling Period	The attribute indicates the time interval at which the incrementBy is applied to reach `maxConcurrentStreamsPerCon`. If not provided, the `actionSamplingPeriod` is used. **Note**: The proposed value for this attribute is 3 for Normal and DoC state. Unit is seconds.	**DataType**: integer **Constraints**: NA **Default Value**: NA

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.17.10 Server Header Details

Perform the following procedure to configure the Server Header Details:

1. From the left navigation menu, navigate to **Ingress Gateway** option to configure the Ingress Gateway APIs.
2. Click **Server Header Details** option under **Ingress Gateway**.
   The **Server Header Details** page is displayed with default values for a few fields.
3. Click **Edit** to modify the attributes in the **Server Header Details** page.
   The **Edit Server Header Details** page is displayed.

Table 5-76 Server Header Details

Attribute Name	Description	Details
Enabled	Indicates if the server header feature is enabled or disabled. If this value is set to true, NRF adds the server header in the error responses. If this value is set to false, NRF does not add the server header in the error responses.	**Data Type**: Boolean **Constraints**: true, false **Default Value**: false **Presence**: M
Configuration	Indicates the configuration type of the network function, such as NF Type and NF Instance Id. For more information about these attributes, see Table 5-77.	**Data Type**: Array[Table 5-77] **Constraints**: NA **Default Value**: NA **Presence**: M

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-76 (Cont.) Server Header Details

Attribute Name	Description	Details
Error Code Series Id	Indicates the error list IDs. The `serverheaderdetails.errorCodeSeriesId` attribute is compared with the `errorCodeSeriesList.Id`. If the `serverheaderDetails.errorCodeSeriesId` matches with the `errorCodeSeriesList.Id`, NRF adds the server header in the responses if the error code generated by the error response matches with the error codes defined in the error code series (`errorCodeSeriesList.errorCodeSeries.errorCodes`). For more information about `errorCodeSeriesList`, see Error Code Series.	**Data Type**: String **Constraints**: NA **Default Value**: NA **Presence**: M

Table 5-77 Server Header Details Configuration

Attribute Name	Description	Details
NF Type	Indicates the nfType of the Network Function. For NRF use cases, the value is always NRF. **Note:** These values are pre-populated.	**Data Type**: String **Constraints**: NA **Default Value**: NRF **Presence**:M
NF Instance Id	Indicates the nfInstanceId of NRF. **Note:** These values are pre-populated.	**Data Type**: String **Constraints**: NA **DefaultValue**: NA **Presence**: M

> **Note**
>
> In case any one or both the values of nfType and nfInstanceId attributes are empty, NRF does not add the server header in the error responses.

4. Use the toggle option to **Enabled** to turn on the feature.
5. Click **Save** on the **Edit Server Header Details** page to save the details.
6. Click **Cancel** to discard your progress and go back to **Server Header Details** page.

5.3.17.11 Congestion Level Configuration

This configuration is used to configure the congestion level of the pods.

Perform the following configurations:

1. From the left navigation menu, navigate to **Ingress Gateway** option to configure the Ingress Gateway APIs.
2. Click the **Congestion Level Configuration** option under Ingress Gateway.

Copyright © 2019, 2025, Oracle and/or its affiliates.

The **Congestion Level Configuration** page is displayed.

3. Click **Edit** to edit the configuration.
   The **Congestion Level Configuration** page is displayed.
4. Configure the fields in the **Edit Congestion Level Configuration** page as mentioned in the below table.

Table 5-78 Congestion Level Configuration

Field Name	Description	Details
Refresh Interval (seconds)	This attribute indicates the refresh interval for the scheduler to calculate congestion level in milliseconds. CPU levels are stable when tested with values more than 500 ms as a refresh interval. Any value less than this, CPU spikes are observed for a shorter period of time, and denied requests are processed based on the action defined for that congestion level.	**Data Type**: Integer **Constraints**: NA **Default Value**: 500
Levels	This attribute indicates an array of different levels of congestion.	**Data Type**: Object **Constraints**: NA **Default Value**: NA

5. Click **Add** to add the configuration levels.
6. Configure the fields in the **Add Levels** page as mentioned in the below table.

Table 5-79 Congestion Levels

Attribute Name	Description	Details
Level Name	Indicates the name of the congestion level. **Note**: This value should be unique.	**Data Type**: String **Constraints**: NA **Default Value**: NA
Level Value	Indicates numerical values for each congestion level. The value ranges from 1-10. The values should be defined from 1 to 10 in increasing order. The default level of the system is 0. This is the state of the system when the CPU is less than the onset of L1 defined in the configuration.	**Data Type**: Integer **Constraints**: 1 -10 **Default Value**: 0
Resources	Indicates the list of resources.	**Data Type**: Object **Constraints**: NA **Default Value**: NA

7. Click **Add** to add the resource details.
8. Configure the fields in the **Add Resources** page as mentioned in the below table.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-80 Resources

Attribute Name	Description	Details
Name	Indicates the resource names. The possible value is CPU.	**Data Type**: String **Constraints**: NA **Default Value**: CPU
Onset Threshold	Indicates the onset threshold. **Note:** Onset should be more than the abatement of the level. Denoted by percentage.	**Data Type**: Integer **Constraints**: NA **Default Value**: NA
Abatement Threshold	Indicates the abatement threshold. Denoted by percentage. **Note:** • Abatement should be less than the onset of the same level. • Abatement should be more than the onset of the previous level.	**Data Type**: Integer **Constraints**: NA **Default Value**: NA

9. Click **Save** on the **Add Resources** page to save the details. Click **Cancel** to discard your progress and go back to **Add Levels** page.
10. Click **Save** on the **Add Levels** page to save the details. Click **Cancel** to discard your progress and go back to **Edit Congestion Level Configuration** page.
11. Click **Save** on the **Congestion Level Configuration** page to save the details.

5.3.17.12 Pod Protection By Rate Limiting

This configuration is used to enable and identify the pods that should be protected by rate limiting feature.

Perform the following configurations:

1. From the left navigation menu, navigate to **Ingress Gateway** option to configure the Ingress Gateway APIs.
2. Click the **Pod Protection By Rate Limiting** option under Ingress Gateway.
   The **Pod Protection By Rate Limiting** page is displayed.
3. Click **Edit** to edit the configuration.
   The **Edit Pod Protection By Rate Limiting** page is displayed.
4. Configure the fields in the **Edit Pod Protection By Rate Limiting** page as mentioned in the below table.

Table 5-81 Pod Protection By Rate Limiting

Field Name	Description	Details
Enabled	The attribute indicates if the Ingress Gateway pod protection using rate limiting feature is enabled or not.	**Data Type**: Boolean **Constraints**: true, false **Default Value**: NA
Fill Rate	This attribute indicates the number of requests to be processed by an Ingress Gateway pod in one second (1000 ms).	**Data Type:** Integer **Constraints**: NA **Default Value**: 2500

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-81 (Cont.) Pod Protection By Rate Limiting

Field Name	Description	Details
Routes	This attribute indicates the route for rate limiting.	**Data Type**: Array (object of Routes) **Constraints**: NA **Default Value**: See Routes for more details
Error Code Profile	This attribute indicates error profile to be sent for the rejected requests.	**Data Type**: String **Constraints**: NA **Default Value**: ERR_POD_PROTECTION_RATE_LIMIT
Denied Request Actions	This attribute indicates the request actions to be performed for the requests which exceeds the fill rate.	**Data Type**: Object **Constraints**: NA **Default Value**: See Denied Request Actions Table.
Priority Header Name	Determines the priority of the request using the configured header. Indicates the name of the header. **Note:** This is for future release and not to be used in this release 25.1.200.	**Data Type**: String **Constraints**: NA **Default Value**: 3gpp-sbi-message-priority
Default Priority	This attribute indicates the default priority in the absence of a priority header in the request. **Note:** This is for future release and not to be used in this release 25.1.200.	**Data Type**: Integer **Constraints**: NA **Default Value**: 24

5. Click **Add** in the **Routes** section to configure different routes for Pod protection.
6. The **Add Routes** page is displayed.
7. Configure the fields in the **Add Routes** page as mentioned in the below table.

Table 5-82 Routes

Field Name	Description	Details
Id	Indicates the route Id that processes the request. The number of route Ids can be more than 1. **Note**: Each route should have a unique route Id.	**Data Type**: Integer **Constraints**: NA **Default Value**: NA
Path	Indicates the path to be matched to apply fill rate on the route.	**Data Type**: String **Constraints**: NA **Default Value**: NA
Methods	Indicates the Http method of the route for which the rate limiting is defined. All valid Http methods are allowed, for example: • POST • PUT • PATCH • DELETE • GET	**Data Type**: Array **Constraints**: • POST • PUT • PATCH • DELETE • GET **Default Value**: NA

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-82 (Cont.) Routes

Field Name	Description	Details
Percentage	This attribute indicates the percentage of requests for a route. If fillRate is 1000 and routes id<id value> percentage is 40%, the route will process 400 requests in 1000 ms. The minimum value is 0.1. **Note:** The total percentage across all routes should not exceed 100.	**Data Type**: Integer **Constraints**: NA **Default Value**: NA

8. Click **Save** on the **Add Routes** page to save the details. Click **Cancel** to discard your progress and go back to **Edit Pod Protection By Rate Limiting** page.
9. Click **Add** in the **Denied Request Actions** section to configure the actions for rejected requests.
10. The **Add Denied Request Actions** page is displayed.
11. Configure the fields in the **Add Denied Request Actions** page as mentioned in the below table.

Table 5-83 Denied Request Actions

Field Name	Description	Details
Id	Indicates the unique Id of the request.	**Data Type**: Integer **Constraints**: NA **Default Value**: NA
Congestion Level	Indicates the congestion level of the pods. This value depends on the **Level Value** field in the congestion level configuration or the default value 0. For more information about congestion level configuration, see Congestion Level Configuration.	**Data Type**: Integer **Constraints**: 1-10 **Default Value**: NA
Action	Rejects or processes a denied request based on the value of congestionLevel attribute in the deniedRequestActions object. The possible values are REJECT and CONTINUE.	**Data Type**: String **Constraints**: REJECT, CONTINUE **Default Value**: NA
Error Code Profile	This attribute indicates error profile to be sent for the rejected requests.	**Data Type**: String **Constraints**: NA **Default Value**: NA

12. Click **Save** on the **Add Denied Request Actions** page to save the details. Click **Cancel** to discard your progress and go back to **Edit Pod Protection By Rate Limiting** page.
13. Click **Save** on the **Edit Pod Protection By Rate Limiting** page to save the details. Click **Cancel** to discard your progress and go back to **Pod Protection By Rate Limiting** page.

5.3.18 Egress Gateway

This section describes the Egress Gateway configurations.

Copyright © 2019, 2025, Oracle and/or its affiliates.

5.3.18.1 Peer Configuration

This configuration is used to add or update the list of peers wherein each peer consists of ID, host, port or virtualHost, and apiPrefix. The ID of each peer is mapped to Peer Identifier in Peer Set Configuration. The default value is null.

Perform the following configurations to configure the peers:

1. From the left navigation menu, navigate to **Egress Gateway** option to configure the Egress Gateway APIs.
2. Click the **Peer Configuration** option under **Egress Gateway** to configure peers.

   The **Peer Configuration** page is displayed.
3. Click **Add** to add the peer configuration

   The **Create Peer Configuration** page is displayed.
4. Configure the fields in the **Create Peer Configuration** page as mentioned in Table 5-84.
5. Click **Save** on the **Create Peer Configuration** page to save the peer configuration. Click **Cancel** to discard your progress and go back to Peer Configuration page.

> **Note**
>
> - Use the **Edit** icon available in the next column of the specific entry to update the Peer Configuration information.
> - Use **Refresh** icon to refresh the list of peers configured.

Table 5-84 Peer Configuration

Field Name	Description	Details
ID	Enter an unique peer identifier value.	**DataType**: string **Constraints**: NA **Default Value**: NA
Host	Host details of a local peer. It can be IPv4, IPv6 and FQDN details.	**DataType**: string **Constraints**: NA **Default Value**: NA
Port	Port details of a local host peer.	**DataType**: string **Constraints**: NA **Default Value**: NA
Virtual Host	Host details of a remote peer. This FQDN is sent to Alternate Route Service for DNS SRV resolution.	**DataType**: string **Constraints**: NA **Default Value**: NA
API Prefix	API prefix details of a peer. **Note**: The recommended value is **/** .	**DataType**: string **Constraints**: NA **Default Value**: NA
healthApiPath	Include the SCP API details.	**DataType**: string **Constraints**: NA **Default Value**: /{scpApiRoot}/{apiVersion}/status

Copyright © 2019, 2025, Oracle and/or its affiliates.

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.18.2 Peer Set Configuration

This configuration is used to add or update the list of peer sets wherein each peer set consists of **ID** and list of HTTP/HTTPS instances. Each instance consists of priority and peer identifier that is mapped to id in peerconfiguration resource. The **ID** of each peer set is mapped to **Peer Set Identifier** in **Routes Configuration** resource. The default value is null.

Perform the following configurations to configure the peer set:

1. From the left navigation menu, navigate to **Egress Gateway** option to configure the Egress Gateway APIs.
2. Click the **Peer Set Configuration** option under **Egress Gateway** to configure peers.

   The **Peer Set Configuration** page is displayed.
3. Click **Add** to add the peer set configuration.

   The **Create Peer Set Configuration** page is displayed.
4. Configure the fields in the **Create Peer Set Configuration** page as mentioned in Table 5-85.
5. Click **Add** to add HTTP Configuration.

   The **Add HTTP Configuration** page is displayed.
6. In the **Add HTTP Configuration** page, configure the fields as mentioned in Table 5-86.
7. Click **Save** to save the details. Click **Cancel** to discard your progress, close the dialog box, and go back to **Create Peer Set Configuration** page.
8. Click **Add** to add HTTPS Configuration.

   The **Add HTTPS Configuration** page is displayed.
9. In the **Add HTTPS Configuration** page, configure the fields as mentioned in Table 5-86.
10. Click **Save** to save the details. Click **Cancel** to discard your progress, close the dialog box, and go back to **Create Peer Set Configuration** page.
11. Click **Save** on the **Create Peer Set Configuration** page to save the details. Click **Cancel** to discard your progress and go back to **Peer Set Configuration** page.

> **Note**
>
> - Use the **Edit** icon available in the next column of the specific entry to update the peer set configuration information.
> - Use **Refresh** icon to refresh the list of peers configured.

Table 5-85 Peer Set Configuration

Field Name	Description	Details
ID	Enter the unique identifier for Peer Set.	**DataType**: string **Constraints**: Unique value in peer set configuration. **Default Value**: NA

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-85 (Cont.) Peer Set Configuration

Field Name	Description	Details
HTTP Configuration	Configuration for HTTP based Peers. This value will be selected, if 3GPPAPIRootScheme value is http.	**DataType**: array (Table 5-86) **Constraints**: NA **Default Value**: NA
HTTPS Configuration	Configuration for HTTPs based Peers. This value will be selected, if 3GPPAPIRootScheme value is https.	**DataType**: array (Table 5-86) **Constraints**: NA **Default Value**: NA

Table 5-86 Peer Identifier Configuration

Field Name	Description	Details
Priority	Priority of peer to be used in a peer set.	**DataType**: integer **Constraints**: Priority must be unique. **Default Value**: NA
Peer Identifier	Peer identifier is the value of peer configured during PeerConfiguration.	**DataType**: string **Constraints**: NA **Default Value**: NA

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.18.3 Peer Monitoring Configuration

This configuration is used to retrieve the peer monitoring configuration.

Perform the following configurations to retrieve peer configuration.

1. From the left navigation menu, navigate to **Egress Gateway** option to configure the Egress Gateway APIs.
2. Click the **Peer Monitoring Configuration** option under **Egress Gateway**.

   The **Peer Monitoring Configuration** page is displayed with default values.
3. Click **Edit** to update the peer monitoring configuration.

   The **Edit Peer Monitoring Configuration** page is displayed.
4. Modify the fields in the **Edit Peer Monitoring Configuration** page as mentioned in Table 5-87.
5. Click **Save** on the **Edit Peer Monitoring Configuration** page to save the details. Click **Cancel** to discard your progress and go back to **Peer Monitoring Configuration** page.

> **Note**
>
> - Use the **Edit** icon available in the next column of the specific entry to update the peer monitoring configuration information.
> - Use **Refresh** icon to refresh the peers configured.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-87 Peer Monitoring Configuration

Field Name	Description	Details
Enabled	Indicates the attribute to enable or disable monitoring at a global level.	**DataType**: Boolean **Constraints**:true or false **Default Value**: false
Timeout	Indicates the flag to configure the duration of time after which calls to the SCP health API is timed out.	**DataType**: long **Constraints**:NA **Recommended Range:** 300 milliseconds to 10000 milliseconds **Default Value**: 1000 milliseconds
Frequency	Indicates the frequency or interval at which Egress Gateway microservice initiates health check calls toward SCP.	**DataType**: long **Constraints**:NA **Recommended Range:** 300 milliseconds to 10000 milliseconds **Default Value**: 2000 milliseconds
Failure Threshold	Indicates the number of consecutive failure responses after which a healthy SCP can be marked as unhealthy. Health API call to given SCP fails consecutively to these many attempts before it is marked as Unavailable from Available.	**Data Type**: Integer **Constraints**: NA **Recommended Range**: 1 to 5 **Default Value**: 3
Success Threshold	Indicates the number of successful responses after which an unhealthy SCP can be marked as healthy. Health API call to given SCP will succeed consecutively to these many attempts before it is marked as Available from Unavailable.	**Data Type**: Integer **Constraints**: NA **Recommended Range**: 1 to 5 **Default Value**: 4

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.18.4 Routes Configuration

This configuration is used to fetch and update the list of routes configuration.

1. From the left navigation menu, navigate to **Egress Gateway** option to configure the Egress Gateway APIs.
2. Click the **Routes Configuration** option under **Egress Gateway** to configure routes.

   The **Routes Configuration** page is displayed.
3. Click **Add** to add the peer configuration.

   The **Create Routes Configuration** page is displayed.
4. Configure the fields in the **Create Routes Configuration** page as mentioned in Table 5-88.

Copyright © 2019, 2025, Oracle and/or its affiliates.

5. Configure the fields in the **metadata** section as mentioned in Table 5-88.
6. Click **Add** in the right side column to add predicates.

   The **Add predicates** page is displayed.
7. Configure the fields in the **Add predicates** page as mentioned in Table 5-88.
8. Click **Save** to save predicates configuration. Click **Cancel** to discard your progress, close the window, and go back to **Create Routes Configuration** page.
9. Configure the fields in the **Filters** section as mentioned in the table.
10. Click **Add** in the right side column to add filters.

    The **Add Filters** window is displayed.
11. Configure the fields in the **Add Filters** window as mentioned in Table 5-89.
12. Click **Add** at the bottom of the window to save filters configuration. Click **Remove** to remove filter configuration.
13. Click **Save** in **Add Filters** window to save the details. Click **Cancel** to discard your progress, close the window, and go back to **Create Routes Configuration** page.
14. Click **Save** on the **Create Routes Configuration** page to save the details. Click **Cancel** to discard your progress and go back to **Routes Configuration** page.

> **Note**
>
> - Use the **Edit** icon available in the next column of the specific entry to update the routes configuration.
> - Use **Refresh** icon to refresh the routes configuration.

Table 5-88 Routes Configuration

Field Name	Description	Details
ID	Indicates the unique route configuration identifier	**DataType**: string **Constraints**: Unique value of route **Default Value**: NA
URI	Provide any dummy URL, or leave the existing URL with the existing value.	**DataType**: string **Constraints**: NA **Default Value**: NA
Order	Provide the order of the execution of this route. **Note:** The value of the order attribute must be unique for each routing configuration.	**DataType**: integer **Constraints**: NA **Default Value**: NA
httpRuriOnly	This flag indicates the scheme of the outgoing request from OCNRF. If the value is set to true, the scheme of RURI is changed to http. If the value is set to false, no change occurs to the scheme. **Note**: In case of non-ASM configuration and `3GPPAPIRootScheme` in Roaming Options is set to https, set the value as false.	**DataType**: boolean (true,false) **Constraints**: NA **Default Value**: NA

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-88 (Cont.) Routes Configuration

Field Name	Description	Details
httpsTargetOnly	For NRF, the value of this flag must always be set to true. **Note**: This is a read-only attribute.	**DataType**: boolean (true, false) **Constraints**: NA **Default Value**: NA
sbiRoutingEnabled	Enables or disables SBI routing true: SbiRouting functionality is enabled false: SbiRouting functionality is disabled	**DataType**: boolean (true, false) **Constraints**: NA **Default Value**: false
Predicates	Header predicate details for matching target PLMN mapped to this SBIRoute rule. **Note**: The predicates can be combined in a single configuration as shown in the Body, or only the required configuration can be retained for processing the message. Sample value:- "predicates": [{ "args": { "header": "OC-MCCMNC", "regexp": "310014" }, "name": "Header" }] **Note**: "header": "OC-MCCMNC" must not be changed. Only "regexp": "310014" can be modified. regexp consists of MCC and MNC values. In this example, value of MCC and MNC is 310 and 014. Multiple values can be provided for regexp as shown below: "predicates": [{ "args": { "header": "OC-MCCMNC", "regexp": "310014" }, "name": "Header" }, { "args": { "header": "OC-MCCMNC", "regexp": "315012" }, "name": "Header" }]	**DataType**: Predicate structure. See description for more details. **Constraints**: NA **Default Value**: NA

Table 5-89 Filters Configuration

Field Name	Description	Details
Name	Provide filtername as "SBIRoutingFilter"	**DataType**: boolean (true, false) **Constraints**: NA **Default Value**: NA
PeerSetIdentifier	This flag maps to `id` of peerSetConfiguration.	**DataType**: String **Constraints**: NA **Default Value**: NA
CustomPeerSelectorEnabled	This flag allows the user to send request to a particular instance directly when enabled according to "ocalternaterouteattempt" header. **Note**: When **Use OAuth Token** is set to **true** in SLF Options, this parameter must also be set to **true**.	**DataType**: boolean (true, false) **Constraints**: NA **Default Value**: NA

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

5.3.18.5 SBI Error Action Sets

This configuration is used to list or update the error action set configuration at Egress Gateway.

Perform the following configurations:

1. From the left navigation menu, navigate to **Egress Gateway** option to configure the Egress Gateway APIs.
2. Click **SBI Error Action Sets** option under **Egress Gateway** to configure action sets.

   The **SBI Error Action Sets** page is displayed.
3. Click **Add** to add an error action set configuration.

   The **Create SBI Error Action Sets** page is displayed.
4. Configure the fields in the **Create SBI Error Action Sets** page as mentioned in Table 5-90.
5. Configure the fields under the Black List section as mentioned in Table 5-91.
6. Click **Save** on the **Create SBI Error Action Sets** page to save the details. Click **Cancel** to discard your progress and go back to **SBI Error Action Sets** page.

> **Note**
>
> - Use the **Edit** icon available in the next column of the specific entry to update the action sets information.
> - Use **Refresh** icon to refresh the list of action sets.

Table 5-90 SBI Error Action Sets

Field Name	Description	Details
ID	Indicate the unique Id for SBI routing error action set.	**DataType**: string **Constraints**: **Default Value**: NA
Action	Action that needs to be taken when specific criteria set is matched.	**DataType**: string **Constraints**: reroute, retry **Default Value**: reroute
Attempts	Maximum number of retries to either same or different peer in case of error or failures from backend.	**DataType**: string **Constraints**: NA **Default Value**: 3

Table 5-91 Blacklist

Field Name	Description	Details
Enabled	This flag enables the peer blacklist feature using the server headers received in the response.	**DataType**: boolean **Constraints**: true, false **Default Value**: false
Duration	The duration for which the peer is blacklisted and no traffic is routed to that peer for this period.	**DataType**: integer **Constraints**: NA **Default Value**: 60000

Copyright © 2019, 2025, Oracle and/or its affiliates.

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.18.6 SBI Error Criteria Sets

This configuration is used to list or update SBI error criteria sets configuration at Egress Gateway. By default, this configuration is disabled.

Perform the following configurations:

1. From the left navigation menu, navigate to **Egress Gateway** option to configure the Egress Gateway APIs.
2. Click **SBI Error Criteria Sets** option under **Egress Gateway** to configure action sets.

   The **SBI Error Criteria Sets** page is displayed.
3. Click **Add** to add an error criteria set configuration.

   The **Create SBI Error Criteria Sets** page is displayed.
4. Configure the fields in the **Create SBI Error Criteria Sets** page as mentioned in Table 5-92.
5. Click **Add** to add HTTP status details under the **Statuses** section.

   The **Add Statuses** page is displayed.
6. Configure the following under **Add Statuses** section as mentioned in Table 5-93.
7. Click **Save** on the **Add Statuses** page to save the details. Click **Cancel** to discard your progress and go back to **Create SBI Error Criteria Sets** page.
8. Click **Save** on the **Create SBI Error Criteria Sets** page to save the details. Click **Cancel** to discard your progress and go back to **SBI Error Criteria Sets** page.

> **Note**
>
> - Use the **Edit** icon available in the next column of the specific entry to update the criteria sets information.
> - Use **Refresh** icon to refresh the list of criteria sets.

Table 5-92 SBI Error Criteria Sets

Field Name	Description	Details
ID	Unique id for a sbiRoutingErrorCriteriaSet	**DataType**: string **Constraints**: Unique value of route **Default Value**: NA
Method	Methods for which reroute or retry is triggered.	**DataType**: string **Constraints**: GET, POST, PUT, PATCH, DELETE **Default Value**: NA
Exceptions	Specific exceptions for which reroute or retry is triggered. **Note**: exceptions and response cannot be configured under same criteria Id.	**DataType**: string **Constraints**: NA **Default Value**: java.util.concurrent.TimeoutException

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-93 Statuses

Field Name	Description	Details
Status Series	HTTP Status Series for which reroute or retry is triggered, when the error response is received from downstream.	**DataType**: string **Constraints**: 4xx, 5xx **Default Value**: NA
Status	Specific HTTP Statuses that belongs to above mentioned status series for which reroute or retry is triggered. To enable retry or reroute for all the HTTP status belonging to a status series, configure this as **-1**.	**DataType**: string **Constraints**: 401, 404 or -1 **Default Value**: NA

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.18.7 Pod Protection By Rate Limiting

This configuration is used to enable and identify the pods that should be protected by rate limiting feature.

Perform the following configurations:

1. From the left navigation menu, navigate to **Egress Gateway** option to configure the Egress Gateway APIs.
2. Click the **Pod Protection By Rate Limiting** option under Egress Gateway.
   The **Pod Protection By Rate Limiting** page is displayed.
3. Click **Edit** to edit the configuration.
   The **Edit Pod Protection By Rate Limiting** page is displayed.
4. Configure the fields in the **Edit Pod Protection By Rate Limiting** page as mentioned in the below table.

Table 5-94 Pod Protection By Rate Limiting

Field Name	Description	Details
Enabled	The attribute indicates if the Egress Gateway pod protection using rate limiting feature is enabled or not.	**Data Type**: Boolean **Constraints**: true, false **Default Value**: NA
Fill Rate	This attribute indicates the number of requests to be processed by an Egress Gateway pod in one second (1000 ms).	**Data Type**: Integer **Constraints**: NA **Default Value**: 2500
Routes	This attribute indicates the route for rate limiting.	**Data Type**: Array (object of Table 5-95) **Constraints**: NA **Default Value**: See Table 5-95.
Error Code Profile	This attribute indicates error profile to be sent for the rejected requests.	**Data Type**: String **Constraints**: NA **Default Value**: ERR_POD_PROTECTION_RATE_LIMIT

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-94 (Cont.) Pod Protection By Rate Limiting

Field Name	Description	Details
Denied Request Actions	This attribute indicates the request actions to be performed for the requests which exceeds the fill rate.	**Data Type**: Array of Object **Constraints**: NA **Default Value**: See Table 5-96.
Priority Header Name	Determines the priority of the request using the configured header. Indicates the name of the header. **Note**: This is for future release and not to be used in this release 25.1.201.	**Data Type**: String **Constraints**: NA **Default Value**: 3gpp-sbi-message-priority
Default Priority	This attribute indicates the default priority in the absence of a priority header in the request. **Note:** This is for future release and not to be used in this release 25.1.201.	**Data Type**: Integer **Constraints**: NA **Default Value**: 24

5. Click **Add** in the Routes section to configure different routes for Pod protection. The **Add Routes** page is displayed.
6. Configure the fields in the **Add Routes** page as mentioned in the below table.

Table 5-95 Routes

Field Name	Description	Details
Id	Indicates the route Id that processes the request. The number of route Ids can be more than 1. **Note**: Each route should have a unique route Id.	**Data Type**: Integer **Constraints**: NA **Default Value**: NA
Path	Indicates the path to be matched to apply fill rate on the route.	**Data Type**: String **Constraints**: NA **Default Value**: NA
Methods	Indicates the Http method of the route for which the rate limiting is defined. All valid Http methods are allowed, for example: • POST • PUT • PATCH • DELETE • GET	**Data Type**: Array **Constraints**: • POST • PUT • PATCH • DELETE • GET **Default Value**: NA
Percentage	This attribute indicates the percentage of requests for a route. If fillRate is 1000 and routes id<id value> percentage is 40%, the route will process 400 requests in 1000 ms.	**Data Type**: Integer **Constraints**: • The minimum value is 0.1. • The total percentage across all routes should not exceed 100. **Default Value**: NA
Default Priority	This attribute indicates the default priority configured for a specific route.	**Data Type**: Integer **Constraints**: NA **Default Value**: NA

7. Click **Save** on the **Add Routes** page to save the details. Click **Cancel** to discard your progress and go back to **Edit Pod Protection By Rate Limiting** page.

Copyright © 2019, 2025, Oracle and/or its affiliates.

8. Click **Add** in the **Denied Request Actions** section to configure the actions for rejected requests.
   The **Add Denied Request Actions** page is displayed.
9. Configure the fields in the **Add Denied Request Actions** page as mentioned in the below table.

Table 5-96 Denied Request Actions

Field Name	Description	Details
Id	Indicates the unique Id of the request.	**Data Type**: Integer **Constraints**: NA **Default Value**: NA
Congestion Level	Indicates the congestion level of the pods. This value depends on the Level Value field in the congestion level configuration or the default value 0. For more information about congestion level configuration, see Congestion Level Configuration.	**Data Type**: Integer **Constraints**: 1-10 **Default Value**: NA
Action	Rejects or processes a denied request based on the value of congestionLevel attribute in the deniedRequestActions object. The possible values are REJECT and CONTINUE.	**Data Type**: String **Constraints**: REJECT, CONTINUE **Default Value**: NA
Error Code Profile	Indicates error profile to be sent for the rejected requests.	**Data Type**: String **Constraints**: NA **Default Value**: NA
Priority	Indicates the priority of the requests which should be processed for the CONTINUE action.	**Data Type**: Integer **Constraints**: NA **Default Value**: NA

10. Click **Save** on the **Add Denied Request Actions** page to save the details. Click **Cancel** to discard your progress and go back to **Edit Pod Protection By Rate Limiting** page.
11. Click **Save** on the **Edit Pod Protection By Rate Limiting** page to save the details. Click **Cancel** to discard your progress and go back to Pod Protection By Rate Limiting page.

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.18.8 Error Code Profile

This configuration is used to update the error code profiles that is used in Pod Protection by Rate Limiting feature for populating details in error responses when a request is discarded.

Perform the following configurations:

1. From the left navigation menu, navigate to **Egress Gateway** option to configure the Egress Gateway APIs.
2. Click **Error Code Profiles** option under Egress Gateway.
   The **Error Code Profiles** page is displayed with default values for few fields.
3. Click **Add** to add the profiles.
   The **Create Error Code Profiles** page is displayed.
4. Configure the fields in the **Create Error Code Profiles** page as mentioned in Table 5-97.

Copyright © 2019, 2025, Oracle and/or its affiliates.

5. Click **Save** on the **Create Error Code Profiles** page to save the details. Click **Cancel** to discard your progress and go back to **Error Code Profiles** page.

- Use the **Edit** icon available in the next column of the specific entry to update the route error profile information.
- Use **Refresh** icon to refresh the route error profile information.

Table 5-97 Error Code Profile Configuration

Field Name	Description	Details
Name	Indicate the error name.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: M
Error Code	Error Code field in an errorScenario determines the HttpStatusCode that needs to be populated in ProblemDetails (HttpStatus field) response from Egress Gateway when the exception occurred at Egress Gateway matches the configured errorScenario's exceptionType field.	**Data Type**: integer **Constraints**: NA **Default Value**: NA **Presence**: M
Error Cause	Error Cause field in an errorScenario determines the error cause that needs to be populated in ProblemDetails (Cause field) response from Egress Gateway when the exception occurred at Egress Gateway matches the configured errorScenario's exceptionType parameter.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: O
Error Title	Error Title field in an errorScenario determines the title that needs to be populated in ProblemDetails (Title field) response from Egress Gateway when the exception occurred at Egress Gateway matches the configured errorScenario's exceptionType parameter.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: O
Redirect URL	Redirect URL field in an errorScenario determines the redirection URL, this value is populated in LOCATION header while sending response from Egress Gateway. The header is populated only when the exception occurred at Egress Gateway matches the configured errorScenario's exceptionType parameter, the errorCode configured for the particular errorScenario lies in 3xx error series and the redirectUrl field for the particular errorScenario is configured appropriately.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: O
Retry After	Retry After field in an errorScenario determines the value in seconds or particular date after which the service should be retried, this value is populated in Retry-After header while sending response from Egress Gateway. The header is populated only when the exception occurred at Egress Gateway matches the configured errorScenario's exceptionType parameter, the errorCode configured for the particular errorScenario lies in 3xx error series and the retry-after field for the particular errorScenario is configured appropriately in seconds.	**Data Type**: integer **Constraints**: NA **Default Value**: NA **Presence**: O

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-97 (Cont.) Error Code Profile Configuration

Field Name	Description	Details
Error Description	Error Description field in an errorScenario determines the description that needs to be populated in ProblemDetails (Detail field) response from Egress Gateway when the exception occurred at Egress Gateway matches the configured errorScenario's exceptionType field.	**Data Type**: string **Constraints**: NA **Default Value**: NA **Presence**: O

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

5.3.18.9 Congestion Level Configuration

This configuration is used to configure the congestion level of the pods.

Perform the following configurations:

1. From the left navigation menu, navigate to **Egress Gateway** option to configure the Egress Gateway APIs.
2. Click the **Congestion Level Configuration** option under Egress Gateway.
   The **Congestion Level Configuration** page is displayed.
3. Click **Edit** to edit the configuration.
   The **Edit Congestion Level Configuration** page is displayed.
4. Configure the fields in the **Edit Congestion Level Configuration** page as mentioned in the below table.
   Table 5-78

Table 5-98 Congestion Level Configuration

Field Name	Description	Details
Refresh Interval (seconds)	This attribute indicates the refresh interval for the scheduler to calculate congestion level in milliseconds. CPU levels are stable when tested with values more than 500 ms as a refresh interval. Any value less than this, CPU spikes are observed for a shorter period of time, and denied requests are processed based on the action defined for that congestion level.	**Data Type**: Integer **Constraints**: NA **Default Value**: 500
Levels	This attribute indicates an array of different levels of congestion.	**Data Type**: Object **Constraints**: NA **Default Value**: NA

5. Click **Add** to add the configuration levels.
6. Configure the fields in the **Add Levels** page as mentioned in the below table.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-99 Congestion Levels

Field Name	Description	Details
Level Name	Indicates the name of the congestion level. **Note**: This value should be unique.	**Data Type**: String **Constraints**: NA **Default Value**: NA
Level Value	Indicates numerical values for each congestion level. The value ranges from 1-10. The values should be defined from 1 to 10 in increasing order. The default level of the system is 0. This is the state of the system when the CPU is less than the onset of L1 defined in the configuration.	**Data Type**: Integer **Constraints**: 1 -10 **Default Value**: 0
Resources	Indicates the list of resources.	**Data Type**: Object **Constraints**: NA **Default Value**: NA

7. Click **Add** to add the resource details.
8. Configure the fields in the **Add Resources** page as mentioned in the below table.

Table 5-100 Resources

Field Name	Description	Details
Name	Indicates the resource names. The possible value is CPU.	**Data Type**: String **Constraints**: NA **Default Value**: NA
Onset Threshold	Indicates the onset threshold. **Note**: Onset should be more than the abatement of the level. Denoted by percentage.	**Data Type**: Integer **Constraints**: NA **Default Value**: NA
Abatement Threshold	Indicates the abatement threshold. Denoted by percentage. **Note:** • Abatement should be less than the onset of the same level. • Abatement should be more than the onset of the previous level.	**Data Type**: Integer **Constraints**: NA **Default Value**: NA

9. Click **Save** on the **Add Resources** page to save the details. Click **Cancel** to discard your progress and go back to **Add Levels** page.
10. Click **Save** on the **Add Levels** page to save the details. Click **Cancel** to discard your progress and go back to **Edit Congestion Level Configuration** page.
11. Click **Save** on the **Congestion Level Configuration** page to save the details.

5.3.19 Alternate Route Service

This section describes the Alternate Route Service configurations.

Copyright © 2019, 2025, Oracle and/or its affiliates.

5.3.19.1 Upstream Dns Config

This configuration is used to update the Upstream Dns Config that is used in Overload Control, Controlled Shutdown feature for populating details in error responses when a request is discarded.

Perform the following configurations:

1. From the left navigation menu, navigate to **Alternate Route Service** option to configure the Alternate Route Service APIs.
2. Click **Upstream Dns Config** option under **Alternate Route Service**.
   The **Upstream Dns Config** page is displayed.
3. Click **Edit** to configure the upstream Dns.
   The **Edit Upstream Dns Config** page is displayed.
4. Configure the fields in the **Edit Upstream Dns Config** page as mentioned in Table 5-101.
5. Click **Save** on the **Edit Upstream Dns Config** page to save the details. Click **Cancel** to discard your progress and go back to **Upstream Dns Config** page.

Table 5-101 Upstream Dns Config

Field Name	Description	Details
Enabled	Indicates if the Alternate Route can communicate to upstream DNS server for dynamic DNS updates for the types A/AAAA/ NAPTR.	**DataType**: boolean **Constraints**: NA **Default Value**: false **Presence**: M
Host	Indicates the host IP address of a upstream dns server.	**DataType**: string **Constraints**: NA **Default Value**: NA **Presence**: M
Port	Indicates the port of a upstream dns server.	**DataType**: string **Constraints**: NA **Default Value**: NA **Presence**: O
Zone	Indicates the zone of a upstream dns server.	**DataType**: string **Constraints**: NA **Default Value**: NA **Presence**: O
TsigKeyNamespace	Indicates the namespace where the secret is created.	**DataType**: string **Constraints**: NA **Default Value**: NA **Presence**: M
TsigKeySecretName	Indicates the secret name configuration.	**DataType**: string **Constraints**: NA **Default Value**: NA **Presence**: O

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 5-101 (Cont.) Upstream Dns Config

Field Name	Description	Details
UpstreamDNSTimeout	Indicates the timeout set for a upstream dns server. **Unit**: Milliseconds	**DataType**: integer **Constraints**: NA **Default Value**: NA **Presence**: M
WatchSecretTimeout	Indicates the watch event timeout (in second). The value should be always 2000 or 3000 ms less than the **fixedTsigKeyMonitorDelay**. **Unit**: Milliseconds	**DataType**: integer **Constraints**: NA **Default Value**: NA **Presence**: M
FixedTsigKeyMonitorDelay	Indicates the monitoring interval for secret to be added or updated. **Unit**: Milliseconds	**DataType**: integer **Constraints**: NA **Default Value**: NA **Presence**: M

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

6
NRF Metrics, KPIs, and Alerts

6.1 NRF Metrics

This section includes information about metrics for Oracle Communications Cloud Native Core, Network Repository Function.

> **Note**
>
> - Sample NRF dashboard for Grafana is delivered to the customer through NRF CSAR package. Metrics and functions used to achieve KPI are covered in NRF CSAR package. For more information on how to download the NRF CSAR package from MOS, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
> - The metric scraping time varies as per the environment in which it is deployed.

The name of the metrics may contain suffix such as total, seconds, max and so on. It gets added by the micrometer registry if it is not present in the metrics name. The metric name has the following format for suffix: `<Basename of the metric>_<Suffix>`

Table 6-1 Metrics type and Suffix

Metric Type	Suffix	Description
Counter	_total	Represents the total number of occurrences of an event or traffic, such as measuring the total amount of traffic received and transmitted by NRF, and so on.
Gauge	NA	Represents a single numerical value that changes randomly. This metric type is used to measure various parameters, such as NRF load values, memory usage, and so on.
Histogram	_max, _bucket, _count, or _sum	Represents the sample observations like request durations or response sizes and counts them in configurable buckets. To measure the latency values, _seconds is used.

Dimensions Legend for the Metrics

The following table includes the details about the metrics dimensions:

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-2 Dimensions Legend

Dimension	Description
Action	Used in Gateway metrics. Indicates the type of action taken for the requests. (rejects or processes a denied request based on the value of congestion level attribute in the denied request actions object. **Sample Values**: REJECT, CONTINUE
ActionId	Used in Gateway metrics. Indicates the unique Id of the request. **Sample Values**: 1, 2
Allowed	Used in Gateway metrics. Indicates if the requests are accepted or rejected. If the request is below fillRate, the value is set as true. If the request is above the fillRate, the value is set as false. **Sample Values**: true, false
application	Indicates the application name. **Sample Values**: ocnrf
authority	Used in Gateway metrics. Indicates the destination address.
BackendSvc	Used in Gateway metrics. Indicates the address of destination.
BackendSvcAddressType	Used in Gateway metrics. Indicates the IP type (IPv4/IPv6) of the destination from the Egress Gateway.
Bucket	Indicates how many profiles are returned in the response of Discovery request. Range is not configurable. Possible values are 0-10, +Inf. According to NF profiles returned, corresponding bucket will be incremented by one. For example, if 2 profiles are returned, then bucket 2 will be incremented by one. Profiles getting returned more than 10 will fall in +Inf bucket.
CertificateName	Indicates the ccaHeaderValidation file name configured in Helm.
ClientCertIdentity	Used in Gateway metrics. Indicates the certificate identity of the client, SAN=127.0.0.1,localhost CN=localhost, N/A if data is not available.
ClientNfInstanceId	Indicates the client NF Instance Id for NF Access Token.
client_type	Used in Gateway metrics. Indicates the type of jetty client being used at Ingress Gateway. **Sample Values:** h2, h2c
CongestionLevel	Used in Gateway metrics. Indicates the congestion level of the pods. **Sample Values:** 0, 1, 2, 3
connectionId	Indicates the connectionId of the http2 connection.
ConsumerInstanceId	Indicates the consumer NF's NfInstanceId configured in Sub in JWT token.
ConsumerNFInstanceId	Indicates the consumer NF Instance Id for NF Access Token
ConsumerNFType	Indicates request is for which consumer NF type.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-2 (Cont.) Dimensions Legend

Dimension	Description
content_available	Used in Gateway metrics. Indicates if the incoming request to Ingress Gateway has a request body present in it. **Sample Values:** true, false
DataType	Indicates the type of data. **Sample Values**: NfProfiles, NfSubscriptions
dbOperation	Indicates the database operation type such as Create, update, delete, and find.
DestinationHost	Used in Gateway metrics. Indicates the destination IP address or FQDN of the host.
destinationHostAddressType	Used in Gateway metrics. Indicates the destination IP type (IPv4 or IPv6) from Egress Gateway.
Direction	Used in Gateway metrics. Indicates the direction of the request or response. In case of Ingress Gateway, the value is ingress. In case of Egress Gateway, the values are egress, egressOut.
DiscardAction	Used in Gateway metrics. Indicates the type of pod discard in overload filter. **Sample Values:** PercentageBased, PriorityBased
dnsResolvedType	Used in Gateway metrics. Indicates the actual DNS resolved IP type (IPv4 or IPv6) of the destination.
egressRoutingMode	Used in Gateway metrics. Indicates the value of the egressRoutingMode configured in Egress Gateway. **Sample Values**: IPv4, IPv6, IPv4_IPv6, IPv6_IPv4, None
ErrorProfile	Used in Gateway metrics. Indicates a valid error code for rejected requests based on the action. **Sample Values**: "ERR_POD_PROTECTION_RATE_LIMIT"
error_reason	Indicates the reason for failure response received. If message is sent in the response, then it is filled with the message otherwise exception class is filled. In case of successful response it is filled with "no-error". **Sample Values**: • "no_error" (In case successful response is received) • "java.nio.channels.ClosedChannelException" • "unable to find valid certification path to requested target" • "SSL handshake failed due to invalid SNI"
ErrorOriginator	Used in Gateway metrics. Indicates the source of error like ServiceProducer, Nrf, IngressGW, None.
ErrorReason	Indicates the CCA header validation failed message like: aud claim validation failed, sub claim validation failed, JWT Signature does not match, X5C certificate has expired.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-2 (Cont.) Dimensions Legend

Dimension	Description
event	Used in Gateway metrics. Indicates the lifecycle event processed during the jetty request processing with the back-end services. **Sample Values:** onQueued, onBegin, onHeaders, onCommit, onContent, onSuccess, onFailure
ExceptionAttribute	Indicates the NFDiscover search query attribute due to which NFDiscover service operation is not rejected even SLF specific mandatory attributes are missing from search query.
FailureReason	Indicates the failure reason for Roaming Support. **Sample Values**: • PeerGeneratedError: Error response received from NRF Peer node. • EgressGatewayGeneratedError: Error response received from NRF Egress Gateway. • InternalError: Roaming request not sent to hNRF because failure occurred at hNRF due to internal error.
ForwardingBarringReason	Indicates the forwarding barring reason for NF Access Token and NFDiscover service operations. **Sample Values**: • forwardingFeatureDisabled: Forwarding feature is disabled. • forwardingRuleNotMatched: Forwarding Rule is not matching with the configuration.
GroupId	Indicates the GroupId returned by SLF/UDR corresponding to SubscriptionId.
HeartbeatTimer	Indicates the heartbeatTimer of the NfProfile. The value is considered in seconds.
Host	Used in Gateway metrics. Indicates the gateway port (IP or FQDN). Indicates the remote address of client connected to the Ingress Gateway.
HttpStatusCode	Indicates the HTTP Status Code. **Sample Values** 429, 500 **Note**: If this error code is generated by Egress Gateway microservice, then the status code is updated as Egw-<response code>. **Sample Value**: Egw-429
HttpVersion	Indicates the Http protocol version. **Sample Values**: • HTTP/1.1 • HTTP/2.0
Id	Used in Gateway metrics. Corresponds to the id defined in the routes of the Pod Protection configuration. **Sample Values**: 1, 2, 3, default_route
InstanceIdentifier	Used in Gateway metrics. Indicates the pod prefixes configured in helm when there are multiple instances in same deployment- Prefix configured in helm otherwise UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-2 (Cont.) Dimensions Legend

Dimension	Description
isCurrentKeyId	Indicates if a specific metric is for current key id in Access Token Configuration. **Sample Values**: true, false
isLeaderPod	Used in Perf-Info. Indicates if the pod calculating the threshold level is a leader pod or not. **Sample Values**: true, false
jetty_request_timeout	Used in Gateway metrics. Indicates if a request at Egress Gateway failed due to request timeout at jetty level. **Sample Values:** true, false
KeyId	Indicates if the Key Id from Access Token Configuration is used to sign the Access Token.
KeyType	Indicates the Key type of Access Token Configuration (private key or certificate).
level	Indicates the type of log level. **Sample Values:** debug, error, info, fatal, warn, trace
LoadLevel	Used in Gateway metrics. Indicates the load level of backend microservices which are configured in the "routes" of the Ingress Gateway microservice. **Sample Values:** L1, L2
Method/method	Indicates the HTTP method name. **Sample Values**: • PUT • GET
NegotiatedTLSVersion	This denotes the TLS version used for communication between the server and the client. **Sample Values**: TLSv1.2, TLSv1.3.
NewStatus	Indicates the new NfStatus of the NfProfile or NfService. **Sample Values**: • REGISTERED • SUSPENDED • UNDISCOVERABLE • DEREGISTERED
NfFqdn	Indicates the FQDN of consumer NF. The nfFqdn dimension will contain the FQDN as present in the nfProfile. When service mesh is enabled, this dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.
NfInstanceId	Indicates the Unique identity of the NF Instance sending request to NRF.
NfServiceFqdn	Indicates the FQDN of the NfService.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-2 (Cont.) Dimensions Legend

Dimension	Description
NFServiceType	Used in Gateway metrics. Indicates the NFType extracted from URI. Path is /**nxxx-yyy**/v**z**/..... Where **nxxx-yyy** is NFServiceType UNKNOWN if unable to extract NFServiceType from the path Example: nnrf-nfm/v1/nf-instances
NfSetFqdn	Indicates the domain name or FQDN for NAPTR record. It is populated using amfSetFqdn.
NfType	Indicates the types of Network Functions (NF).
NFType	Used in Gateway metrics. Indicates the NFType extracted from URI. Path is /n**xxx-yyy**/v**z**/..... Where **xxx will be changed to (Upper Case)** is NFType UNKNOWN if unable to extract NFType from the path Example: nnrf-nfm/v1/nf-instances
Node	Indicates the name of the Kubernetes worker node on which microservice is running.
NotificationEventType	Indicates the subscription request is for which event types. **Sample Values**: • NF_REGISTERED • NF_DEREGISTERED • NF_PROFILE_CHANGED
NotificationHostPort	Indicates the port of notification callback server.
NRFRole	Indicates the NRF role whether it is hNRF or vNRF. **Sample Values**: hNRF, vNRF
NrfFqdn	Indicates the FQDN of NRF.
nrfInstanceId	Indicates the Nrf Instance Id of NRF.
NrfLevel	Indicates the NRF Deployment Name by which NRF can be identified, it will be NRF Instance Id passed through Helm.
Nsi	Indicates the Network slices are identified by Network Slice Instances (NSIs). It is in the string format.
NumberOfRetriesAttempted	Indicates the number of retries attempted to notification callback server.
oAuthHeaderPresent	Used in Gateway metrics. Indicates if the oAuth header is present or not. **Sample Values:** true, false
oAuthTokenPresent	Used in Gateway metrics. Indicates if the oAuth token is present or not. **Sample Values:** true, false
OperationType	Indicates the Operation type of the request sent towards DNS server. **Sample Values:** • Create: NF create request towards DNS server • Update: NF update request towards DNS Server • Delete: NF delete request towards DNS Server

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-2 (Cont.) Dimensions Legend

Dimension	Description
OperationType	Indicates if the NFSubscribe Service operation to specify if the request is to create or update the subscription. **Sample Values**: • Create • Update
peer	Indicates the SCP peer.
peerset	Indicates the SCP peerset.
PreviousSLFFqdn	Indicates the target SLF FQDN where the previous request was sent to and failed.
PreviousStatus	Indicates the previous NfStatus of the NfProfile or NfService. **Sample Values**: • REGISTERED • SUSPENDED • UNDISCOVERABLE • DEREGISTERED
Priority	Used in Gateway metrics. Indicates the priority of the request. **Sample Values**: 24
ProducerInstanceId	Indicates the NrfInstanceId.
quantile	Indicates the latency values with ranges as 10ms, 20ms, 40ms, 80ms, 100ms, 200ms, 500ms, 1000ms, and 5000ms.
QueryType	Indicates the type of the query sent by the NF. **Sample Values**: • NfProfiles • NfSubscriptions
receivedAddressType	Used in Gateway metrics. Indicates the IP type (IPv4/IPv6) of the remote client connected to the Ingress Gateway.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-2 (Cont.) Dimensions Legend

Dimension	Description
RejectionReason	Indicates the rejection reason for NF Access Token, Roaming Support, Subscription limit breached. **Sample Values**: • ClientNotAuthorized: Number of access token request for which client authorized failed • ProducerWithRequestedScopeNotFound: Number of access token not granted because of no producer instance registered for service/s in the scope • ProducerWithRequestedNfInstanceIdNotFound: Number of access token not granted because of no producer instance registered for No producer instance is registered at all for provided target Instance Id in request. • InconsistentScope: Number of access token not granted because services in the scope belong to different NF types. • ProducerNFTypeMismatch: Number of access token not granted because producer NF type in profile is not matching with the access token request. • InternalError: Number of access token not granted because failure at NRF due to internal error. • ConsumerNfTypeNotAllowed: Number of access token not granted because the consumer NFType is not allowed to access the requested NF. • ConsumerPlmnNotAllowed: Number of access token not granted because the consumer NF PLMN is not allowed to access the requested NF. • SecretNotAccessible: Number of access token not granted because the secret for current key id is not accessible. • InvalidFileData: Number of access token not granted because the current key id file data is invalid. • NamespaceNotAccessible: Number of access token not granted because the namespace for current key id is not accessible. • FileNotFound: Number of access token not granted because the file not found in secrets. • CurrentKeyIdNotConfigured: Number of access token not granted because the current key id is not configured. • ExpiredCertificate: Number of access token not granted because the NRF certificate is expired. • BadRequest: Number of access token not granted because the Request is incorrect. • NFFqdnAuthenticationFailed: Number of access token not granted because of NF Fqdn Authentication Failure • ConfigurationMissing: Number of access token not granted because of configuration missing • ConsumerDetailsNotMatched: Number of access token not granted because consumer details in profile is not matching with the access token request. • InvalidProducerPlmn: Number of access token not granted because targetPlmn in profile is not matching with the access token request. • ProducerNFProfileNotFound: Number of access token not granted because there are no NFs found for the requested profile

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-2 (Cont.) Dimensions Legend

Dimension	Description
	• AuthScreeningFailed: Number of access token not granted because the consumer NF is not authorized to access the requested NF or its services. • InternalError: Request not forwarded because failure occurred at NRF due to internal error. • NRFCommunicationFailure: Request not forwarded because NRF is not able to connect with another NRF. • ErrorFromNRF: Error response received from another NRF. • NRFForwardingConfigurationMissing: Request not forwarded due to missing forwarding configuration. • LoopDetected: Received request message is already processed by NRF during forwarding mechanism. • *NotApplicable is applicable for 2xx Status code. • MandatoryAttributesMissing: Roaming request is rejected due to mandatory attribute missing in the request. • TargetMatchesNRFPLMN: Request intended for roaming but target PLMN matches with the vNRF PLMN. • TargetNotMatchesNRFPLMN: Request intended for roaming hNRF but target PLMN not matches with the hNRF PLMN. • NotificationURINonPLMNFormat: Notification URI in NFStatusSubscribe request is not in 3GPP defined PLMN format for roaming. • RoamingDisabledAttributesPresent: Roaming is disabled but roaming specific parameters are present. • RequesterNFTypeNotAllowed: Requester NF Type is not allowed to access the NF Producer Type. • RequesterPLMNNotAllowed: Requester PLMN Type is not allowed to access the NF Producer Type. • RequesterSNSSAINotAllowed: Requester SNSSAI Type is not allowed to access the NF Producer Type. • PeerGeneratedError: Error response received from NRF Peer node. • EgressGatewayGeneratedError: Error response received from NRF Egress Gateway. • InternalError: Roaming request not sent to hNRF because failure occurred at hNRF due to internal error. • EgressGatewayOverloaded: Error response received from NRF Egress Gateway when it is overloaded.
RemoteNrfInstanceId	Indicates the Nrf Instance Id of the remote NRF.
RemoteSiteName	Indicates the site name of remote NRF.
Replacement	Indicates the value as populated using 3GPP AMF name. **Sample Values:** • 0 - Healthy • 1 - Failure - Any Failure which is not covered below • 2 - Not Found in DNS server • 3 - Mismatch with DNS Server and NRF data
RequesterNfType	Indicates the NF type which originating the request. This value comes from UserAgent header. For NFDiscover Service operation it is taken from Search Query. In case, there is no header or value, this value will be UNKNOWN in the metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-2 (Cont.) Dimensions Legend

Dimension	Description
RequesterPLMNId	Indicates the PLMN ID which originating the request.
RequestType	Indicates the type of the service request sent by the NF. **Sample Values**: • RemoteAndLocalSetProfiles - is pegged for the requests received from discovery and remote cache data service. • ProfilesByQueryParam - is pegged for the requests received from the nfregistration service for NfProfileRetrieval and NfListRetrieval, and from nfaccesstoken service for AccessToken requests. • RemoteAndLocalSetSubscriptions - is pegged for requests querying for subscription data.
ResponseReason	Indicates the response reason in response sent back to NF. **Sample Values**: • ResponseReason =SLFCommunicationFailure = NRF unable to communication with SLF • ResponseReason =MandatoryParamsMissing = Mandatory Parameter (Subscriber data) for performing SLF is missing in the incoming discovery request • ResponseReason =SLFSubscriberNotProvisioned = Subscriber not provisioned • ResponseReason =ErrorFromSLF = Error response received from SLF (non 2xx, excluding 404) • ResponseReason =InternalError = NRF internal error • ResponseReason =SuccessFromSLF = Successful SLF lookup • ResponseReason = SkipSlfAttributeUsedFromSearchQuery = No SLF performed due to SkipSLF attribute configuration • ResponseReason = ExceptionAttributePresentInSearchQuery = No SLF performed due to Exception attribute configuration
route_id	Used in Gateway metrics. Indicates a unique Id to identify a specific route. All the requests always get mapped to a unique route. **Sample Values**: egress_scp_proxy1, default_route
RouteLevelColor	Used in Gateway metrics. Indicates Green color for the requests accepted at route level policer and Red for requests rejected at route level policer. **Sample Values**: RED, GREEN
Route_path	Used in Gateway metrics. Indicates the Path predicate or Header predicate that matched the current request.
Scheme	Inidicates the HTTP protocol scheme. **Sample Values**: • HTTP • HTTPS • UNKNOWN
Scope	Indicates the scope as received in the AccessToken Request **Sample Values**: NfDiscover, NFProfileRetrieval
SecretName	Indicates the k8SecretName configured in Helm.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-2 (Cont.) Dimensions Legend

Dimension	Description
serialNumber	Indicates the type of the certificate. **Sample Values**: serialNumber=4661 is used for RSA and serialNumber =4662 is used for ECDSA
service	Used in Perf-Info. Indicates the NRF microservice type. **Sample Values**: ocnrf-nfaccesstoken, ocnrf-nfdiscovery
ServiceInstanceId	Indicates the unique ID of the service instance within a given NF Instance.
ServiceName	Indicates the service instance name. **Sample Values**: nudm-sdm
ServiceOperation	Indicates the service operations as defined in 3GPP specification for NRF. **Sample Values** • NfProfileRetrieval • NfListRetrieval • NfAccessToken • NfDiscover • NfStatusSubscribe • NfStatusUnsubscribe • NfStatusNotify • nrfCacheDataService
siteName	Indicates the site name of NRF.
SkipSLFLookupParameter	Indicates the NFDiscover search query attribute due to which SLF lookup is skipped. **Sample Values**: group-id-list
SkipSLFLookupValue	Indicates the configured value for which SLF lookup is skipped, if `enableValueBasedSkipSLFLookup` is set to true. If the value of `enableValueBasedSkipSLFLookup` is set as false, the value of SkipSLFLookupValue dimension is set as NotApplicable.
slfFqdn	Indicates the FQDN of the SLF.
sni_status	Indicates if the SNI feature is enabled or disabled when an invalid SNI (direct and indirect routing) is sent. **Sample Values**: • true • false
sNssai	Indicates the Network slices are identified by Single Network Slice Selection Assistance Information (SNSSAI). It is in string format as outlined in 3GPP TS 29.571 section 5.4.4.2.
SourceNrfSetId	Indicates the set ID of the source NRF.
Status	Indicates the HTTP Status Code in response. When used in Gateway metrics it indicates the status of the pod. **Sample Values**: DISCARDED
StatusCode	Indicates the HTTP Status code.
SubscriberPLMNId	Indicates request is for which subscriber PLMN ID.
SubscriptionIdType	Indicates the Subscription Id type for which SLF query is received.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-2 (Cont.) Dimensions Legend

Dimension	Description
subscriptionLimitFeatureStatus	Indicates whether the Subscription Limit feature is enabled or disabled. **Sample Values**: • enabled • disabled
SubscriptionStatus	Indicates the Status of subscription. **Sample Values**: • SUBSCRIBED • SUSPENDED • UNSUBSCRIBED
tableName	Indicates the NRF Table Name
TargetNfInstanceId	Indicates the target NF Instance Id for NF Access Token.
TargetNfType	Indicates request is for which target NF type.
TargetNrfFqdn	Indicates the FQDN of the NRF to which the request is forwarded to.
TargetNrfSetId	Indicating the set ID of the target NRF.
TargetPLMNId	Indicates request is for which target PLMN ID.
TargetSLFFqdn	Indicates the target SLF FQDN where the current SLF request is sent.
TLSFqdn	Indicates the Fully Qualified Domain Name (FQDN) received in TLS Certificate.
TriggerMode	Indicates the trigger mode. **Sample Values** • Automatic: Triggers generated by 3GPP service operations or NRF Audit. • Manual: Triggers generated by the operator initiated retrigger API.
type	This dimension is used in Gateway services and in Perf-Info. • Gateway services: Indicates whether it is a request towards Data Director or an acknowledgement from Data Director. **Sample Values**: – req: Indicates request is sent from Ingress Gateway Microservice towards Data Director. – ack: Indicates Data Director has sent an acknowledgement for the received request. This will be only pegged if ack is enabled in NRF Message Feed feature. • Perf-Info: Indicates the resource type. **Sample Value**: svc_failure_count, svc_pending_count, cpu
Updatetype (Partial/Complete)	Indicates if the NF Update is with PUT (Complete) or PATCH (Partial) methods. **Sample Values**: • PUT • PATCH
Uri	Indicates the URI defined to identify the service operation at Ingress Gateway.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-2 (Cont.) Dimensions Legend

Dimension	Description
vendor	Indicates the Vendor name of NRF. **Sample Values**:Oracle
vfqdn	Indicates the FQDN of the peer.
XfccHeaderPresent	Used in Gateway metrics. Indicates if the XFCC header is present or not. **Sample Values:** true, false

6.1.1 NRF Gateways Metrics

This section provides details about the Gateway metrics of NRF.

6.1.1.1 Ingress Gateway Metrics

Table 6-3 oc_ingressgateway_msgcopy_requests_total

Field	Details
Description	This is incremented whenever request message is sent or acknowledged from Kafka.
Type	Counter
Dimensions	type

Table 6-4 oc_ingressgateway_msgcopy_responses_total

Field	Details
Description	This is incremented whenever response message is sent or acknowledged from Kafka.
Type	Counter
Dimensions	type

Table 6-5 oc_ingressgateway_dd_unreachable

Field	Details
Description	This indicates whether DD (kafka) is reachable or not 0 - reachable, 1 - unreachable
Type	Gauge

Table 6-6 oc_ingressgateway_jetty_request_stat_metrics_total

Field	Details
Description	This metric is pegged for every event occurred when a request is sent to Ingress Gateway.
Type	Counter
Dimensions	event, client_type, InstanceIdentifier, DestinationHost

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-7 oc_ingressgateway_jetty_response_stat_metrics_total

Field	Details
Description	This metric is pegged for every event occurred when a response is received by Ingress Gateway.
Type	Counter
Dimensions	event, client_type, InstanceIdentifier, DestinationHost

Table 6-8 oc_ingressgateway_traffic_rejection_mode

Field	Details
Description	Gauge metric that pegs if the traffic is being rejected or not. The possible values are 0 and 1 (where "1" stands for rejected).
Type	Gauge

Table 6-9 oc_ingressgateway_http_requests_total

Field	Details
Description	Total number of ingress requests received at NRF.
Type	Counter
Dimensions	ClientCertIdentity, NFType, NFServiceType, Host, DestinationHost, oAuthTokenPresent, HttpVersion, Scheme, Route_path, InstanceIdentifier, consumerNfType, consumerInstanceId, ConsumerFqdn

Table 6-10 oc_ingressgateway_http_responses_total

Field	Details
Description	Total number of responses for the following service operations at NRF: • NfRegister • NfUpdate (complete/partial replacement) • NfDelete • NfProfileRetrieval • NfListRetrieval • NFStatusSubscribe • NFStatusUnSubscribe • NfDiscover • NfAccessToken
Type	Counter
Dimensions	Status, Method, Route_path, NFType, NFServiceType, Host, DestinationHost, HttpVersion, Scheme, InstanceIdentifier, ClientCertIdentity, ErrorOriginator, jetty_request_timeout, error-reason

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-11 oc_ingressgateway_http_responses_total (4xx responses)

Field	Details
Description	Total number of 4xx responses for the following service operations at NRF. • NfRegister • NfUpdate (complete/partial replacement) • NfDelete • NfProfileRetrieval • NfListRetrieval • NFStatusSubscribe • NFStatusUnSubscribe • NfDiscover • NfAccessToken
Type	Counter
Dimensions	Status, Method, Route_path, NFType, NFServiceType, Host, DestinationHost, HttpVersion, Scheme, InstanceIdentifier, ClientCertIdentity, ErrorOriginator, jetty_request_timeout, error-reason

Table 6-12 oc_ingressgateway_http_responses_total (5xx responses)

Field	Details
Description	Total number of 5xx responses for the following service operations at NRF. • NfRegister • NfUpdate (complete/partial replacement) • NfDelete • NfProfileRetrieval • NfListRetrieval • NFStatusSubscribe • NFStatusUnSubscribe • NfDiscover • NfAccessToken
Type	Counter
Dimensions	Status, Method, Route_path, NFType, NFServiceType, Host, DestinationHost, HttpVersion, Scheme, InstanceIdentifier, ClientCertIdentity, ErrorOriginator, jetty_request_timeout, error-reason

Table 6-13 oc_ingressgateway_request_latency_seconds

Field	Details
Description	Time (in microseconds) to process an Ingress Gateway request. It is measured from when the request is received to when the response is sent
Type	Histogram
Dimensions	quantile, InstanceIdentifier, Route_path, Method

Table 6-14 oc_ingressgateway_connection_failure_total

Field	Details
Description	Total number of failures at Ingress Gateway while establishing a connection with NRF.
Type	Counter

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-14 (Cont.) oc_ingressgateway_connection_failure_total

Field	Details
Dimensions	Direction, InstanceIdentifier, ErrorOriginator

Table 6-15 oc_ingressgateway_request_processing_latency_seconds

Field	Details
Description	Metric to capture the amount of time taken for processing of the request only within Ingress Gateway
Type	Histogram
Dimensions	quantile, InstanceIdentifier, Route_path, Method

Table 6-16 oc_ingressgateway_request_content_metrics_total

Field	Details
Description	This metric is pegged by default filter RequestContentMetrics. It pegs whether a request has the request body or not.
Type	Counter
Dimensions	method, content_available, InstanceIdentifier

Table 6-17 oc_ingressgateway_xfcc_header_validate_total

Field	Details
Description	This metric is pegged when xfccHeaderValidation is enabled in XfccHeaderValidationFilter. This metric along with the specified dimension captures the successful/ un-successful validation of XFCC header in the incoming request.
Type	Counter
Dimensions	Route_path, Status, Cause, CertsCompared, InstanceIdentifier, ErrorOriginator, ServiceType, ErrorDetail

Table 6-18 oc_ingressgateway_incoming_connections

Field	Details
Description	Gauge metric that will peg active incoming connections from client to Ingress Gateway.
Type	Gauge
Dimensions	Direction, Host, InstanceIdentifier

Table 6-19 oc_ingressgateway_outgoing_connections

Field	Details
Description	Gauge metric that will peg active outgoing connections from Ingress Gateway to destination.
Type	Gauge
Dimensions	Direction, Host, InstanceIdentifier

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-20 oc_ingressgateway_count_invalid_route

Field	Details
Description	Counter metric that is pegged for every invalid route.
Type	Counter
Dimensions	oAuthHeaderPresent, XfccHeaderPresent

Table 6-21 oc_ingressgateway_request_latency_seconds_bucket

Field	Details
Description	This metric is pegged when a request comes to Ingress gateway. It shows latency consumed by ingress to process the request.
Type	Histogram
Dimensions	Method, Route_path, InstanceIdentifier

Table 6-22 oc_ingressgateway_server_latency_seconds_bucket

Field	Details
Description	This metric is pegged when a request comes to Ingress gateway. It shows latency consumed by the backend to process the request.
Type	Histogram
Dimensions	Method, Route_path, InstanceIdentifier

Table 6-23 oc_ingressgateway_jetty_request_stat_metrics_total

Field	Details
Description	This metric is pegged for every outgoing request from Ingress gateway to the backend at jetty level
Type	Counter
Dimensions	DestinationHost, InstanceIdentifier, event, client_type

Table 6-24 oc_ingressgateway_cca_header_request_total

Field	Details
Description	This metric is pegged when CCA validation is enabled and CCA header is present in the request.
Type	Counter
Dimensions	ConsumerInstanceId, ProducerInstanceId

Table 6-25 oc_ingressgateway_cca_header_response_total

Field	Details
Description	This metric is pegged when an error response is sent due to a validation error of the CCA header.
Type	Counter
Dimensions	ConsumerInstanceId, ProducerInstanceId, StatusCode, ErrorReason

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-26 oc_ingressgateway_cca_certificate_info

Field	Details
Description	Gauge metric used to peg the expiry date of the certificate. This metric is used further for raising alarms.
Type	Counter
Dimensions	CertificateName, SecretName, SerialNumber

Table 6-27 oc_ingressgateway_pod_congestion_state

Field	Details
Description	The metric is used to track the overall congestion state per Ingress Gateway pod. The values of the metric is as follow: • -1: Feature is Disabled • 0: Normal • 1: DangerOfCongestion • 2: Congested **Note**: Prometheus automatically injects pod name with label name "kubernetes_pod_name" to metric. This information is used for alert.
Type	Gauge
Dimensions	InstanceIdentifier

Table 6-28 oc_ingressgateway_pod_resource_stress

Field	Details
Description	This metric pegs the CPU usage and the pending message count, based on which Ingress Gateway pod is calculating its congestion state.
Type	Gauge
Dimensions	InstanceIdentifier, type Possible values for type are CPU, MEMORY, PENDING_REQUEST. **Note**: The type "MEMORY" is not used to determine the pod congestion state in NRF Ingress Gateway.

Table 6-29 oc_ingressgateway_pod_resource_state

Field	Details
Description	This metric pegs the congestion state per resource type which is calculated based on the resource usage and configured threshold. The values of the metric is as follow: • 0: Normal • 1: DoC • 2: Congested **Note**: Prometheus automatically injects pod name with label name "kubernetes_pod_name" to metric. This information is used for alert.
Type	Gauge
Dimensions	InstanceIdentifier, type Possible values for type are CPU, MEMORY, PENDING_REQUEST. **Note**: The type "MEMORY" is not used to determine the pod congestion state in NRF Ingress Gateway.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-30 oc_ingressgateway_incoming_pod_connections_rejected_total

Field	Details
Description	This metric pegs the number of connections dropped at Congested or DoC state. **Note**: Prometheus automatically injects pod name with label name "kubernetes_pod_name" to metric. This information is used for alert.
Type	Counter
Dimensions	InstanceIdentifier, Direction

Table 6-31 oc_ingressgateway_incoming_tls_connections

Field	Details
Description	Number of TLS connections received on the Ingress Gateway and their negotiated TLS versions. The versions can be TLSv1.3 or TLSv1.2.
Type	Gauge
Dimensions	NegotiatedTLSVersion, Host, Direction, InstanceIdentifier

Table 6-32 oc_ingressgateway_incoming_ip_type

Field	Details
Description	This is incremented when the IP type of the active incoming connections from the client to Ingress Gateway.
Type	Gauge
Dimensions	host, receivedAddressType

Table 6-33 oc_ingressgateway_outgoing_ip_type

Field	Details
Description	This is incremented when the IP type of the active outgoing connections from Ingress Gateway to the backend services.
Type	Gauge
Dimensions	BackendSvc, BackendSvcAddressType

Table 6-34 oc_ingressgateway_http_request_ratelimit_values_total

Field	Details
Description	This metric tracks the total count of requests handled by the rate limiting filter, categorized by whether each request was allowed or denied. If there was no route configuration for the dropped message, the id attribute is set to "NA".
Type	Counter
Dimension	Id, Allowed, RouteLevelColor

Table 6-35 oc_ingressgateway_http_request_ratelimit_reject_chain_duration_histogram_seconds

Field	Details
Description	This metric records the reject chain duration.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-35 (Cont.)
oc_ingressgateway_http_request_ratelimit_reject_chain_duration_histogram_seconds

Field	Details
Type	Histogram
Dimension	NA

Table 6-36 oc_ingressgateway_http_request_ratelimit_reject_chain_length_histogram

Field	Details
Description	This metric records the number of sequential requests rejected due to rate limiting. This metric identifies periods of high traffic volume.
Type	Histogram
Dimension	NA

Table 6-37 oc_ingressgateway_http_request_ratelimit_denied_count_total

Field	Details
Description	This metric records the details of the denied request. If there is no denied request action configuration, the ActionId attribute is set to "default".
Type	Counter
Dimension	Action, ActionId, CongestionLevel, Priority, ErrorProfile

Table 6-38 oc_ingressgateway_congestion_cpu_state

Field	Details
Description	This metric records the CPU levels at the congestion state.
Type	Gauge
Dimension	NA

Table 6-39 oc_ingressgateway_congestion_system_state

Field	Details
Description	Records the congestion level of the pod based on all the resources. The highest congestion level recorded for all resources in that interval is the system congestion level. However, currently, the congestion level is calculated based on CPU usage. Therefore, it records the system level based on CPU.
Type	Gauge
Dimension	NA

Table 6-40 oc_ingressgateway_system_state_duration_percentage

Field	Details
Description	This metric records the percentage of time the system was in a particular congestion level.
Type	Gauge
Dimension	level

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-41 oc_ingressgateway_congestion_level_total

Field	Details
Description	This metric counts the number of times congestion levels, such as Normal, Danger Of Congestion, Congestion, and so on, were identified as defined in Ingress Gateway Pod Protection Using Rate Limiting. This congestion level value varies between 0-10.
Type	Counter
Dimension	NA

Table 6-42 oc_ingressgateway_congestion_level_bucket_total

Field	Details
Description	This metric identifies the number of times the congestion level was observed. **Note:** This metric also records any increase in CPU, which sometimes is not recorded in the `oc_ingressgateway_congestion_cpu_state` metric. Refer to both the metrics to obtain the congestion level of the pod.
Type	Histogram
Dimension	NA

Table 6-43 oc_ingressgateway_congestion_cpu_percentage_bucket

Field	Details
Description	This metric records the CPU utilization percentages, tracking the distribution of values across predefined ranges. For example, 0 to 10 %, 11 to 20 %, 21 to 30 %, 31 to 40 %, 41 to 50 %, 51 to 60 %, 61 to 70 %, 71 to 80 %, 81 to 90 %, and 91 to 100%.
Type	Histogram
Dimension	NA

Table 6-44 oc_ingressgateway_route_overloadcontrol_discard_total

Field	Details
Description	This metric is pegged when the incoming request is discarded by the Overload load filter on percentage basis. **Note**: The LoadLevel label identifies the exact overload threshold at which each request is dropped.
Type	Counter
Dimension	DiscardAction, ErrorOriginator, InstanceIdentifier, LoadLevel, Method, Scheme, ServiceName, Status

6.1.1.2 Egress Gateway Metrics

Table 6-45 oc_egressgateway_msgcopy_requests_total

Field	Details
Description	This is incremented whenever request message is sent or acknowledged from Kafka.
Type	Counter

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-45 (Cont.) oc_egressgateway_msgcopy_requests_total

Field	Details
Dimensions	type

Table 6-46 oc_egressgateway_msgcopy_responses_total

Field	Details
Description	This is incremented whenever response message is sent or acknowledged from Kafka.
Type	Counter
Dimensions	type

Table 6-47 oc_egressgateway_dd_unreachable

Field	Details
Description	This indicates whether DD (kafka) is reachable or not 0 - reachable, 1 - unreachable
Type	Gauge
Dimensions	NA

Table 6-48 oc_egressgateway_http_requests_total

Field	Details
Description	Metric to capture the request count reaches the Egress Gateway from the application microservice and pegs with Direction as egress. Also, when the request goes out of Egress Gateway and pegs with Direction as egressOut.
Type	Counter
Dimensions	Method, NFType, NFServiceType, Host, HttpVersion, Scheme, Proxy, InstanceIdentifier, Direction,DestinationHost

Table 6-49 oc_egressgateway_http_responses_total

Field	Details
Description	Metric to capture when Egress Gateway sends response back to backend NF microservice and pegs with Direction as egress. Also, when the response is received Egress gateway and pegs with Direction as egressOut
Type	Counter
Dimensions	Status, Method, NFType, NFServiceType, Host, HttpVersion, Scheme, InstanceIdentifier, Direction, BlacklistedFqdn, DestinationHost, ErrorOriginator, jetty_request_timeout, error-reason

Table 6-50 oc_egressgateway_connection_failure_total

Field	Details
Description	This metric will be pegged by jetty client when the destination is not reachable by Egress Gateway. Here, the destination is producer NF.
Type	Counter

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-50 (Cont.) oc_egressgateway_connection_failure_total

Field	Details
Dimensions	Host, Port, InstanceIdentifier, Direction, error_reason, ErrorOriginator

Table 6-51 oc_egressgateway_request_processing_latency_seconds

Field	Details
Description	Metric captures the amount of time taken for processing the request only within Egress Gateway
Type	Histogram
Dimensions	quantile, InstanceIdentifier, Route_path, Method

Table 6-52 oc_egressgateway_request_latency_seconds

Field	Details
Description	This metric is pegged in the last custom filter of the Egress Gateway while the response is being sent back to the consumer NF. This metric tracks the amount of time taken for processing the request. It starts as soon the request reaches the first custom filter of the application and lasts till the response is sent back to the consumer NF from the last custom filter of the application.
Type	Histogram
Dimensions	quantile, InstanceIdentifier, Route_path, Method

Table 6-53 oc_egressgateway_request_content_metrics_total

Field	Details
Description	This metric is pegged by default filter RequestContentMetrics. It pegs whether request has request body or not.
Type	Counter
Dimensions	method, content_available, InstanceIdentifier

Table 6-54 oc_egressgateway_incoming_connections

Field	Details
Description	Gauge metric that will peg active incoming connections from client to Egress Gateway.
Type	Gauge
Dimensions	Direction, Host, InstanceIdentifier

Table 6-55 oc_egressgateway_outgoing_connections

Field	Details
Description	Gauge metric that will peg active outgoing connections from Egress Gateway to destination.
Type	Gauge
Dimensions	Direction, Host, InstanceIdentifier

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-56 oc_egressgateway_server_latency_seconds

Field	Details
Description	This metric is pegged in Jetty response listener that captures the amount of time taken for processing of the request by jetty client
Type	Histogram
Dimensions	quantile, InstanceIdentifier, Host, Method

Table 6-57 oc_egressgateway_jetty_request_stat_metrics_total

Field	Details
Description	This metric is pegged for every outgoing request from Egress gateway to the backend at jetty level
Type	Counter
Dimensions	DestinationHost, InstanceIdentifier, event

Table 6-58 oc_egressgateway_jetty_response_stat_metrics_total

Field	Details
Description	This metric is pegged for every event occurred when a response is received by Egress Gateway.
Type	Counter
Dimensions	event, client_type, InstanceIdentifier, DestinationHost

Table 6-59 oc_egressgateway_sni_error_total

Field	Details
Description	Indicates the count of TLS connection failure due to SNI validation failure.
Type	Counter
Dimensions	error_reason, sni_status

Table 6-60 oc_egressgateway_peer_health_status

Field	Details
Description	It defines Egress Gateway peer health status. • This metric is set to 1, if a peer is unhealthy. • This metric is reset to 0, when it becomes healthy again. • This metric is set to -1, if peer is removed from peerconfiguration.
Type	Gauge
Dimensions	peer, vfqdn

Table 6-61 oc_egressgateway_peer_health_ping_request_total

Field	Details
Description	This metric is incremented every time a health ping is sent toward a peer.
Type	Counter

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-61 (Cont.) oc_egressgateway_peer_health_ping_request_total

Field	Details
Dimensions	peer, vfqdn **Note**: If virtual host is configured then vfqdn is pegged along with peer otherwise only peer will be pegged.

Table 6-62 oc_egressgateway_peer_health_ping_response_total

Field	Details
Description	This metric is incremented every time a health ping response is received from a peer irrespective of success or failure.
Type	Counter
Dimensions	peer, vfqdn, status, cause **Note**: If virtual host is configured then vfqdn is pegged along with peer otherwise only peer will be pegged.

Table 6-63 oc_egressgateway_peer_health_status_transitions_total

Field	Details
Description	This metric is incremented every time a peer is transitioned from Available to Unavailable or from Unavailable to Available.
Type	Counter
Dimensions	peer, vfqdn, from, to

Table 6-64 oc_egressgateway_peer_count

Field	Details
Description	This metric is incremented every time for the peer count.
Type	Gauge
Dimensions	peerset

Table 6-65 oc_egressgateway_peer_available_count

Field	Details
Description	This metric is incremented every time for the available peer count.
Type	Gauge
Dimensions	peerset

Table 6-66 oc_egressgateway_outgoing_tls_connections

Field	Details
Description	Number of TLS connections received on the Egress Gateway and their negotiated TLS versions. The versions can be TLSv1.3 or TLSv1.2.
Type	Gauge
Dimensions	NegotiatedTLSVersion, Host, Direction, InstanceIdentifier

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-67 oc_egressgateway_incoming_ip_type

Field	Details
Description	This is incremented when the IP type of the active incoming connections from the NRF microservices to the Egress Gateway.
Type	Gauge
Dimensions	host, receivedAddressType

Table 6-68 oc_egressgateway_outgoing_ip_type

Field	Details
Description	This is incremented when the IP type of the active outgoing connections from Egress Gateway to the destination.
Type	Gauge
Dimensions	destinationHost, destinationHostAddressType

Table 6-69 oc_egressgateway_dualstack_ip_rejected_total

Field	Details
Description	This is incremented by counting the total IP rejections which are caused by a IP mismatch between the IP type configured in the egressRoutingMode and the IP type returned by DNS resolution.
Type	Gauge
Dimensions	authority, egressRoutingMode, dnsResolvedType

Table 6-70 oc_egressgateway_podlevel_throttling_allowed_total

Field	Details
Description	Indicates the total number of requests processed by Egress Gateway microservice when the Egress Gateway pod throttling feature is enabled. **Note**: The metric will not be pegged if the Egress Gateway pod throttling feature is disabled.
Type	Counter
Dimensions	route_id

Table 6-71 oc_egressgateway_http_request_ratelimit_values_total

Field	Details
Description	This metric tracks the total number of requests handled by the rate limiting filter, categorized by whether each request was allowed or denied. If there was no route configuration for the dropped message, the id attribute is set to "NA".
Type	Counter
Dimensions	Id, Allowed, RouteLevelColor

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-72 oc_egressgateway_http_request_ratelimit_reject_chain_duration_histogram_seconds_bucket

Field	Details
Description	Records the reject chain duration.
Type	Histogram
Dimensions	NA

Table 6-73 oc_egressgateway_http_request_ratelimit_reject_chain_length_histogram_bucket

Field	Details
Description	Records the number of sequential requests rejected due to rate limiting. This metric identifies periods of high traffic volume.
Type	Histogram
Dimensions	NA

Table 6-74 oc_egressgateway_http_request_ratelimit_denied_count_total

Field	Details
Description	Records the details of the total denied requests. If there is no denied request action configuration, the ActionId attribute is set to "default".
Type	Counter
Dimensions	Action, ActionId, CongestionLevel, Priority, ErrorProfile

Table 6-75 oc_egressgateway_congestion_cpu_state

Field	Details
Description	Records the CPU levels at the congestion state.
Type	Gauge
Dimensions	NA

Table 6-76 oc_egressgateway_congestion_system_state

Field	Details
Description	Records the congestion level of the pod based on all the resources. The highest congestion level recorded for all resources in that interval is the system congestion level. However, currently, the congestion level is calculated based on CPU usage. Therefore, it records the system level based on CPU.
Type	Gauge
Dimensions	NA

Table 6-77 oc_egressgateway_system_state_duration_percentage

Field	Details
Description	Records the percentage of time the system was in a particular congestion level.
Type	Gauge
Dimensions	level

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-78 oc_egressgateway_congestion_level_total

Field	Details
Description	Counts the number of times congestion levels, such as Normal, Danger Of Congestion, Congestion, and so on, were identified as defined in Egress Gateway Pod Protection Using Rate Limiting.
Type	Counter
Dimensions	NA

Table 6-79 oc_egressgateway_congestion_level_bucket_total

Field	Details
Description	Identifies the number of times the congestion level was observed. **Note:** This metric also records any increase in CPU, which sometimes is not recorded in the `oc_egressgateway_congestion_cpu_state` metric. Refer to both the metrics to obtain the congestion level of the pod. 0,1,2,3...10 based on configured levels.
Type	Histogram
Dimensions	NA

Table 6-80 oc_egressgateway_congestion_cpu_percentage_bucket

Field	Details
Description	Records the CPU utilization percentages, tracking the distribution of values across predefined ranges. For example, 0 to 10 %, 11 to 20 %, 21 to 30 %, 31 to 40 %, 41 to 50 %, 51 to 60 %, 61 to 70 %, 71 to 80 %, 81 to 90 %, and 91 to 100%.
Type	Histogram
Dimensions	NA

6.1.1.3 Common Gateway Metrics

Table 6-81 oc_configserver_reachability

Field	Details
Description	Indicates whether microservices that uses a common config client to interact with the common config server is able to establish connection and fetch configuration from the common config server. • The value is set to 1 if the microservices are able to connect and read the value from the common config server. • The value is set to 0 if the microservices receive errors while connecting with the common config server.
Type	Gauge
Dimensions	NA

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-82 oc_configclient_request_total

Field	Details
Description	This metric is pegged whenever the configuration client is polling for configuration update from common configuration server.
Type	Counter
Dimensions	releaseVersion, configVersion

Table 6-83 oc_configclient_response_total

Field	Details
Description	This metric is pegged whenever config client receives response from common configuration server.
Type	Counter
Dimensions	releaseVersion, configVersion, updated

Table 6-84 oc_oauth_nrf_request_total

Field	Details
Description	This metric is pegged in the OAuth client implementation if the request is sent to NRF for requesting the OAuth token. OAuth client implementation is used in Egress Gateway.
Type	Counter
Dimensions	ConsumerNFInstanceId, ConsumerNFType, TargetNFType, TargetNFInstanceId, scope, NrfFqdn

Table 6-85 oc_oauth_nrf_response_success_total

Field	Details
Description	This metric is pegged in the OAuth client implementation if an OAuth token is successfully received from the NRF. OAuth client implementation is used in Egress Gateway.
Type	Counter
Dimensions	ConsumerNFInstanceId, ConsumerNFType, TargetNFType, TargetNFInstanceId, scope, NrfFqdn,StatusCode

Table 6-86 oc_oauth_nrf_response_failure_total

Field	Details
Description	This metric is pegged in the OAuthClientFilter in Egress Gateway whenever GetAccessTokenFailedException is caught.
Type	Counter
Dimensions	ConsumerNFInstanceId, ConsumerNFType, TargetNFType, TargetNFInstanceId, scope, NrfFqdn, ErrorOriginator, StatusCode

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-87 oc_oauth_request_failed_internal_total

Field	Details
Description	This metric is pegged in the OAuthClientFilter in Egress Gateway whenever InternalServerErrorException is caught.
Type	Counter
Dimensions	ConsumerNFInstanceId, ConsumerNFType, ErrorOriginator, NrfFqdn, StatusCode, TargetNFType, TargetNFInstanceId, scope

Table 6-88 oc_oauth_token_cache_total

Field	Details
Description	This metric is pegged in the OAuth Client Implementation if the OAuth token is found in the cache.
Type	Counter
Dimensions	ConsumerNFInstanceId, ConsumerNFType, TargetNFType, TargetNFInstanceId, scope, NrfFqdn

Table 6-89 oc_oauth_request_invalid_total

Field	Details
Description	This metric is pegged in the OAuthClientFilter in Egress Gateway whenever a BadAccessTokenRequestException/JsonProcessingException is caught.
Type	Counter
Dimensions	ConsumerNFInstanceId, ConsumerNFType, TargetNFType, TargetNFInstanceId, scope, StatusCode, ErrorOriginator, NrfFqdn

Table 6-90 security_cert_x509_expiration_seconds

Field	Details
Description	Indicates the time to certificate expiry in epoch seconds.
Type	Histogram
Dimensions	serialNumber

6.1.2 Platform Metrics

This section provides details about the platform metrics used in NRF.

Table 6-91 log4j2_events_total

Field	Details
Description	This metric tracks the total number of logging events generated by a microservice at each specified log level. It helps monitor the volume of logs such as DEBUG, INFO, WARN, and ERROR over time.
Type	Counter
Service Operation	NFRegister, NFUpdate, NFDeregister, NFStatusSubscribe, NFStatusNotify, NFStatusUnSubscribe, NFListRetrieval, NFProfileRetrieval, NFDiscover, Access Token Request

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-91 (Cont.) log4j2_events_total

Field	Details
Dimensions	level

6.1.3 Configuring NRF Metrics Dashboard in OCI

This section describes the steps to upload the `ocnrf_oci_metric_dashboard_25.1.201.json` file on OCI Logging Analytics Dashboard. As OCI doesn't support Grafana, OCI uses the Logging Analytics Dashboard Service for visualizing the metrics and logs.

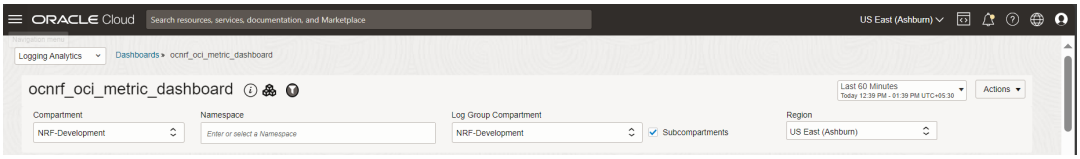
The steps are:

1. Log in to OCI Console.

   > **Note**
   >
   > For more details about logging in to the OCI, refer to Signing In to the OCI Console.

2. Open the navigation menu and click **Observability & Management**.
3. Under **Logging Analytics**, click **Dashboards**. The **Dashboards** page appears.
4. Choose the **Compartment** on the left pane.
5. Click **Import dashboards**.
6. User can select and upload the `ocnrf_oci_metric_dashboard_25.1.201.json` file. ##METRIC_NAMESPACE in json file must be customized to metrics namespace that the user provided while deploying OCI adaptor.
7. **Import dashboard** page appears. Click **Import** button on the page.
8. Click filter option in the ocnrf_oci_metric_dashboard page and select the **Compartment**, **Namespace**, **Log Group Compartment**, and **Region** from the drop-down list.
   User can view the imported dashboard and can view the metrics in the dashboard.

Figure 6-1 ocnrf_oci_metric_dashboard



For more details, see *Oracle Communications Cloud Native Core, OCI Adaptor Deployment Guide*.

6.1.4 NRF NF Metrics

This section provides details about the NRF metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-92 ocnrf_nfRegister_rx_requests_total

Field	Details
Description	Number of Registration Requests received.
Type	Counter
Dimensions	NrfLevel, NfInstanceId, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-93 ocnrf_nfRegister_tx_responses_total

Field	Details
Description	Number of Registration Responses sent.
Type	Counter
Dimensions	NrfLevel, NfInstanceId, RequesterNfType, HttpStatusCode, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-94 ocnrf_nfRegister_rx_requests_success_perService_total

Field	Details
Description	Number of Registrations received and processed successfully per Service.
Type	Counter
Dimensions	NrfLevel, NfInstanceId, serviceName, serviceInstanceId, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.
Recommended legend to see dimension level data (as applicable)	NfRegistrations Per Service [serviceName :- {{ serviceName }}, nfInstanceId :- {{NfInstanceId}}]

Table 6-95 ocnrf_nfUpdate_rx_requests_total

Field	Details
Description	Number of Update Requests received.
Type	Counter
Dimensions	NrfLevel, NfInstanceId, RequesterNfType, Updatetype (Partial/Complete), NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-96 ocnrf_nfUpdate_tx_responses_total

Field	Details
Description	Number of Update Responses sent.
Type	Counter

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-96 (Cont.) ocnrf_nfUpdate_tx_responses_total

Field	Details
Dimensions	NrfLevel, NfInstanceId, RequesterNfType, Updatetype (Partial/Complete), HttpStatusCode, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-97 ocnrf_nfUpdate_rx_requests_success_perService_total

Field	Details
Description	Number of NfUpdates received and processed successfully per Service.
Type	Counter
Dimensions	NrfLevel, Updatetype (Partial/Complete), NfInstanceId, ServiceName, ServiceInstanceId, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.
Recommended legend to see dimension level data (as applicable)	NFUpdates Per Service [serviceName :- {{ serviceName }}, serviceInstanceId:- {{ServiceInstanceId}}]

Table 6-98 ocnrf_nfHeartbeat_rx_requests_total

Field	Details
Description	Number of Heartbeat Requests received.
Type	Counter
Dimensions	NrfLevel, NfInstanceId, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-99 ocnrf_nfHeartBeat_tx_responses_total

Field	Details
Description	Number of Heartbeat Responses sent
Type	Counter
Dimensions	Nrflevel, NfInstanceId, RequesterNfType, HttpStatusCode, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-100 ocnrf_nfDeregister_rx_requests_total

Field	Details
Description	Number of deregistration requests received.
Type	Counter

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-100 (Cont.) ocnrf_nfDeregister_rx_requests_total

Field	Details
Dimensions	NrfLevel, NfInstanceId, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-101 ocnrf_nfDeregister_tx_responses_total

Field	Details
Description	Number of deregistration responses sent.
Type	Counter
Dimensions	NrfLevel, NfInstanceId, RequesterNfType, HttpStatusCode, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-102 ocnrf_nfDeregister_rx_requests_success_perService_total

Field	Details
Description	Number of deregistration requests received and process successfully per service.
Type	Counter
Dimensions	NrfLevel, serviceName, serviceInstanceId, NfInstanceId, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.
Recommended legend to see dimension level data (as applicable)	NFDeregistration Per Service [serviceName :- {{ serviceName }}, serviceInstanceId:- {{ServiceInstanceId}}]

Table 6-103 ocnrf_nfListRetrieval_rx_requests_total

Field	Details
Description	Number of NFListRetrieval requests received.
Type	Counter
Dimensions	NrfLevel, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-104 ocnrf_nfListRetrieval_tx_responses_total

Field	Details
Description	Number of NFListRetrieval responses sent.
Type	Counter

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-104 (Cont.) ocnrf_nfListRetrieval_tx_responses_total

Field	Details
Dimensions	NrfLevel, RequesterNfType, HttpStatusCode, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-105 ocnrf_nfProfileRetrieval_rx_requests_total

Field	Details
Description	Number of NFProfileRetrieval requests received.
Type	Counter
Dimensions	NrfLevel, NfInstanceId, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-106 ocnrf_nfProfileRetrieval_tx_responses_total

Field	Details
Description	Number of NFProfileRetrieval responses sent.
Type	Counter
Dimensions	NrfLevel, NfInstanceId, HttpStatusCode, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-107 ocnrf_heartbeat_missed_total

Field	Details
Description	Number of heartbeats missed.
Type	Counter
Dimensions	NrfLevel, NfType, NfInstanceId, NfFqdn **Note**: The NfFqdn dimension contain the FQDN as present in the NfProfile.

Table 6-108 ocnrf_nfStatusSubscribe_rx_requests_total

Field	Details
Description	Number of NStatusSubscribe requests received.
Type	Counter
Dimensions	NrfLevel, RequesterNfType, OperationType, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-109 ocnrf_nfStatusSubscribe_tx_responses_total

Field	Details
Description	Number of NfStatusSubscribe responses sent. In case of subscription limit feature, NRF includes and populates new measurements for number of new subscription requests or subscription renewal rejected (due to limit being exceeded) per second.
Type	Counter
Dimensions	NrfLevel, RequesterNfType, HttpStatusCode, OperationType, NfFqdn, RejectionReason **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-110 ocnrf_nfset_active_subscriptions

Field	Details
Description	Number of global subscription count status. For georedundancy, it reflects the total subscriptions in quad or triplet set.
Type	Gauge
Dimensions	NrfLevel

Table 6-111 ocnrf_nfset_limit_level

Field	Details
Description	Indicates the global subscription limit threshold level.
Type	Gauge
Dimensions	NrfLevel

Table 6-112 ocnrf_subscription_migration_status

Field	Details
Description	Indicates the status of the migration of records in the NfSubscriptions table. Value 1 indicates that the migration is in Progress. Value 0 indicates the migration is Complete.
Type	Gauge
Dimensions	subscriptionLimitFeatureStatus

Table 6-113 ocnrf_nfStatusUnsubscribe_rx_requests_total

Field	Details
Description	Number of NfStatusUnsubscribe requests received.
Type	Counter
Dimensions	NrfLevel, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-114 ocnrf_nfStatusUnsubscribe_tx_responses_total

Field	Details
Description	Number of NfStatusUnsubscribe responses sent.
Type	Counter
Dimensions	NrfLevel, RequesterNfType, HttpStatusCode, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-115 ocnrf_nfStatusNotify_tx_requests_total

Field	Details
Description	Number of NfStatusNotify requests sent.
Type	Counter
Dimensions	NrfLevel, NotificationEventType, TargetNfType, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-116 ocnrf_nfStatusNotify_rx_responses_total

Field	Details
Description	Number of NfStatusNotify responses received.
Type	Counter
Dimensions	NrfLevel, NotificationEventType, NotificationHostPort, NumberOfRetriesAttempted, TargetNfType, HttpStatusCode, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-117 ocnrf_nfStatusNotify_requests_failed_total

Field	Details
Description	Number of NfStatusNotify requests failed to sent out.
Type	Counter
Dimensions	NrfLevel, NotificationEventType, TargetNfType, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-118 ocnrf_nfDiscover_rx_requests_total

Field	Details
Description	Number of NfDiscover Requests received.
Type	Counter
Dimensions	NrfLevel, TargetNfType, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.
Recommended legend to see dimension level data (as applicable)	NfDiscover Req [TargetNf :- {{ TargetNfType }}, RequesterNfType :- {{RequesterNfType}}]

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-119 ocnrf_nfDiscover_tx_responses_total

Field	Details
Description	Number of NfDiscover responses sent.
Type	Counter
Dimensions	NrfLevel, TargetNfType, RequesterNfType, HttpStatusCode, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-120 ocnrf_nfDiscover_rx_requests_success_perService_total

Field	Details
Description	Number of NfDiscover requests received and processed successfully per Service.
Type	Counter
Dimensions	NrfLevel, RequesterNfType, serviceName, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.
Recommended legend to see dimension level data (as applicable)	NFDiscover Per Service [serviceName :- {{ serviceName }}]

Table 6-121 ocnrf_nfDiscover_profiles_discovered_total

Field	Details
Description	Number of Profiles returned in discovery response. Depending on bucket size and corresponding value will tell how many profiles are returned in discovery response.
Type	Counter
Dimensions	NrfLevel, TargetNfType, Bucket, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.
Recommended legend to see dimension level data (as applicable)	Discovered profiles [TargetNfType :- {{TargetNfType}}, Bucket :- {{ Bucket }}]

Table 6-122 ocnrf_active_registrations_count

Field	Details
Description	Number of active NFs that are registered at the local site at any point of time.
Type	Counter
Dimensions	RequesterNfType, NrfLevel
Recommended legend to see dimension level data (as applicable)	Active Registrations [NfType-{{ NfType }}, NrfLevel-{{ NrfLevel }}]

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-123 ocnrf_message_processing_time_seconds

Field	Details
Description	Time taken by NRF specific microservice to process the service operation (NfRegister/ NfUpdate-Partial/NfUpdate-Complete/NfDeregister/NfProfileRetrieval/NfListRetrieval/ NfHeartBeat/NfDiscover/NFStatusSubscribe/NFStatusUnSubscribe/NfAccessToken) **Note**: Latency calculated by this metric does not include time taken by NRF API gateway.
Type	Histogram
Dimensions	NrfLevel, NfFqdn, RequesterNfType, ServiceOperation **Note**: The NfFqdnss dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.
Recommended legend to see dimension level data (as applicable)	Avg NRF Latency {{ ServiceOperation }} {{ RequesterNfType }}

Table 6-124 ocnrf_dbmetrics_total

Field	Details
Description	Database operation count corresponding to every service operation.
Type	Counter
Dimensions	method, dbOperation, NrfLevel, HttpStatusCode

Table 6-125 ocnrf_dbmetrics_round_trip_time_seconds

Field	Details
Description	Time (in microseconds) taken by database operation corresponding to every service operation (NfRegister/NfUpdate/NfDeregister/NfProfileRetrieval/NfListRetrieval/NfHeartBeat/ NfDiscover/NfSubscribe/NfUnsubscribe/NfStatusNotify/NfAccessToken/NrfConfiguration/ InternalConfigurationCache)
Type	Histogram
Dimensions	method, dbOperation, serviceOperation, tableName: (NRF Table Names), NrfLevel, HttpStatusCode

Table 6-126 ocnrf_nf_suspended_total

Field	Details
Description	The number of profiles that have been marked as SUSPENDED when a profile has missed nfHeartBeatMissAllowed.
Type	Counter
Dimensions	NrfLevel, NfInstanceId,NfStatus, HeartbeatTimer, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-127 ocnrf_nfSubscriptions_expired_total

Field	Details
Description	The number of NfSubscriptions whose validityTime has expired.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-127 (Cont.) ocnrf_nfSubscriptions_expired_total

Field	Details
Type	Counter
Dimensions	NrfLevel

Table 6-128 ocnrf_nfDiscover_emptyList_total

Field	Details
Description	This metric is pegged: • When feature flag is ***DISABLED*** and discovery request processing resulted in EMPTY LIST. • When feature flag is ***ENABLED*** and discovery request processing resulted in all matching NFs in SUSPENDED state.
Type	Counter
Dimensions	NfType, FeatureStatus

Table 6-129 ocnrf_operational_state

Field	Details
Description	This metric is pegged when the operational state of NRF switches between NORMAL and COMPLETE_SHUTDOWN. • 0 - Indicates the operational state is NORMAL. • 1 - Indicates the operational state is COMPLETE_SHUTDOWN. The metric is pegged in the NrfAuditor service, if any one of the NrfAuditor pod shows the operational state as COMPLETE_SHUTDOWN, then the state is considered as COMPLETE_SHUTDOWN.
Type	Gauge
Dimensions	NrfLevel

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-130 ocnrf_audit_status

Field	Details
Description	This metric indicates whether the NRF Audit is in-progress or in paused state. The status of the audit procedures at . • 0 - Indicates the NRF audit is in-progress. • 1 - Indicates the NRF audit is paused. The metric is pegged in the NrfAuditor service, if any one of the NrfAuditor pod shows the operational state as COMPLETE_SHUTDOWN, then the state is considered as COMPLETE_SHUTDOWN. **Note:** NrfAuditor will continue to remain in the paused state for some time, even after `OcnrfOperationalStateCompleteShutdown` alarm is cleared. For more information, see "From CONTROLLED_SHUTDOWN to NORMAL" in Controlled Shutdown of NRF section.
Type	Gauge
Dimensions	NrfLevel

Table 6-131 ocnrf_nfDiscover_limiting_profile_count_for_nfSet_total

Field	Details
Description	This metric will be pegged when the "Limiting the Number of Producers Based on NF Set Ids and Extended Preferred Locality" features are enabled and the number of NF profiles sent in the discovery response is less than the `profileCountInDiscoveryResponse` configuration.
Type	Counter
Dimensions	TargetNfType, RequesterNfType, NrfLevel

Table 6-132 ocnrf_nfDiscover_limiting_profiles_not_applied_for_nfSet_total

Field	Details
Description	This metric will be pegged when "Limiting the Number of Producers Based on NF Set Ids" feature could not be applied due to the unavailability of `nfSetIdList` in any one of the NF profiles selected with the first matching location(s) from `targetPreferredLocations` configuration in `extendedPreferredLocality`.
Type	Counter
Dimensions	TargetNfType, RequesterNfType, NrfLevel

Table 6-133 ocnrf_nfDiscover_tx_response_size_bytes

Field	Details
Description	The size of the NfDiscover response. The size is pegged in bytes and is published in buckets of 1KB, 3KB, 7KB, 10KB, 15KB, 20KB, 30KB, 40KB, 50KB, 60KB, 70KB, and 80KB.
Type	Histogram
Dimensions	TargetNfType

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-134 ocnrf_nf_profile_size

Field	Details
Description	The size of the registered NfProfiles. **Note**:The size of the metric is calculated as per the payload received during registration or update of the profile. To calculate the size of the payload, save the payload in a text file without spaces, and verify the size of the file using Linux tools like wc.
Type	Gauge
Dimensions	NfInstanceId, NfType

Table 6-135 ocnrf_nfProfile_size_limit_breached

Field	Details
Description	Indicates the number of NF profiles that get rejected due to profile size limit during registration or update service operations.
Type	Counter
Dimensions	NfInstanceId, NfType, ServiceOperation

6.1.5 NF Screening Metrics

This section provides details about the NF screening metrics.

Table 6-136 ocnrf_nfScreening_nfFqdn_requestFailed_total

Field	Details
Description	The total number of requests for which screening failed against NF FQDN screening list.
Type	Counter
Service Operation	NFRegister, NFUpdate
Dimensions	NrfLevel, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-137 ocnrf_nfScreening_nfFqdn_requestRejected_total

Field	Details
Description	The total number of requests rejected for which screening failed against NF FQDN screening list.
Type	Counter
Service Operation	NFRegister, NFUpdate
Dimensions	NrfLevel, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-138 ocnrf_nfScreening_nfIpEndPoint_requestFailed_total

Field	Details
Description	The total number of requests for which screening failed against NF IP endpoint screening list.
Type	Counter

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-138 (Cont.) ocnrf_nfScreening_nfIpEndPoint_requestFailed_total

Field	Details
Service Operation	NFRegister, NFUpdate
Dimensions	NrfLevel, RequesterNfType, NfFqdn

Table 6-139 ocnrf_nfScreening_nfIpEndPoint_requestRejected_total

Field	Details
Description	The total number of requests rejected because screening failed against NF IP endpoint screening list.
Type	Counter
Service Operation	NFRegister, NFUpdate
Dimensions	NrfLevel, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-140 ocnrf_nfScreening_callbackUri_requestFailed_total

Field	Details
Description	The total number of requests for which screening failed against Callback URI screening list.
Type	Counter
Service Operation	NFRegister, NFUpdate, NFStatusSubscribe
Dimensions	NrfLevel, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-141 ocnrf_nfScreening_callbackUri_requestRejected_total

Field	Details
Description	The total number of requests rejected because screening failed against Callback URI screening list.
Type	Counter
Service Operation	NFRegister, NFUpdate, NFStatusSubscribe
Dimensions	NrfLevel, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-142 ocnrf_nfScreening_plmnId_requestFailed_total

Field	Details
Description	The total number of requests for which screening failed against PLMN id screening list.
Type	Counter
Service Operation	NFRegister, NFUpdate
Dimensions	NrfLevel, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-143 ocnrf_nfScreening_nfFqdn_requestRejected_total

Field	Details
Description	The total number of requests rejected because screening failed against PLMN id screening list.
Type	Counter
Service Operation	NFRegister, NFUpdate
Dimensions	NrfLevel, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-144 ocnrf_nfScreening_nfTypeRegister_requestFailed_total

Field	Details
Description	The total number of NFRegister requests rejected as NF type was not allowed to register with NRF.
Type	Counter
Service Operation	NFRegister
Dimensions	NrfLevel, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-145 ocnrf_nfScreening_nfTypeRegister_requestRejected_total

Field	Details
Description	The total number of NFRegister requests for which screening failed against NF type screening list.
Type	Counter
Service Operation	NFRegister
Dimensions	NrfLevel, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-146 ocnrf_nfScreening_notApplied_InternalError_total

Field	Details
Description	The total number of times screening not applied due to internal error.
Type	Counter
Service Operation	NFRegister, NFUpdate, NFStatusSubscribe
Dimensions	NrfLevel, RequesterNfType, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

6.1.6 NF Access token Metrics

This section provides details about the NF access token metrics.

Table 6-147 ocnrf_accessToken_rx_requests_total

Field	Details
Description	The total number of access token requests received.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-147 (Cont.) ocnrf_accessToken_rx_requests_total

Field	Details
Type	Counter
Service Operation	AccessToken
Dimensions	TargetNfType, ClientNfType, TargetNfInstanceId, ClientNfInstanceId, ServiceName, NrfLevel, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-148 ocnrf_accessToken_tx_responses_total

Field	Details
Description	The total number of access token responses sent.
Type	Counter
Service Operation	AccessToken
Dimensions	TargetNfType, RequesterNfType, TargetNfInstanceId, ClientNfInstanceId, Scope, NrfLevel, HttpStatusCode, NfFqdn, KeyId, RejectionReason **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

6.1.7 NRF Configuration Metrics

This section provides details about the NRF configuration metrics.

Table 6-149 ocnrf_oauth_keyData_healthStatus

Field	Details
Description	Oauth Token Signing keys health status Value 0 - Healthy Value 1 - Unhealthy
Type	Gauge
Service Operation	Configuration
Dimensions	KeyId, KeyType, isCurrentKeyId, NrfLevel

Table 6-150 ocnrf_oauth_currentKeyId_configuredStatus

Field	Details
Description	Oauth Current Key Id Configuration Status Value 0 - Healthy Value 1 - Unhealthy
Type	Gauge
Service Operation	Configuration
Dimensions	NrfLevel

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-151 ocnrf_oauth_keyData_expiryStatus

Field	Details
Description	Oauth Token Signing keys Expiry Status (Value is expiry time in epoch time)
Type	Gauge
Service Operation	Configuration
Dimensions	KeyId, isCurrentKeyId, NrfLevel

6.1.8 NRF SLF Metrics

This section provides details about the NRF SLF metrics.

Table 6-152 ocnrf_nfDiscover_forSLF_rx_requests_total

Field	Details
Description	The total number of NF Discover request received for SLF.
Type	Counter
Service Operation	NFDiscover
Dimensions	TargetNfType, NrfLevel, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-153 ocnrf_nfDiscover_forSLF_tx_responses_total

Field	Details
Metric Name	ocnrf_nfDiscover_forSLF_tx_responses_total
Description	The total number of NF Discover responses sent for SLF.
Type	Counter
Service Operation	NFDiscover
Dimensions	TargetNfType, NrfLevel, HttpStatusCode, ResponseReason, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-154 ocnrf_SLF_tx_requests_total

Field	Details
Description	The total number of SLF query request sent.
Type	Counter
Service Operation	NFDiscover
Dimensions	TargetNfType, NrfLevel, SubscriptionIdType, NfFqdn, slfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-155 ocnrf_SLF_rx_responses_total

Field	Details
Description	The total number of SLF query response received.
Type	Counter
Service Operation	NFDiscover
Dimensions	TargetNfType, NrfLevel, SubscriptionIdType, HttpStatusCode, GroupId, NfFqdn, slfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-156 ocnrf_nfDiscover_SLFlookup_skipped_total

Field	Details
Description	The total number of NF Discover requests received for SLF but SLF lookup is skipped.
Type	Counter
Service Operation	NFDiscover
Dimensions	TargetNfType, NrfLevel, NfFqdn, SkipSLFLookupParameter, SkipSLFLookupValue **Note**: • The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN. • If the value of `enableValueBasedSkipSLFLookup` is set as false, the value of SkipSLFLookupValue dimension is set as NotApplicable.

Table 6-157 ocnrf_nfDiscover_continue_mandatoryAttributes_missing_total

Field	Details
Description	The total number of NF Discover requests processed while SLF is enabled but SLF mandatory attributes are missing in Search query.
Type	Counter
Service Operation	NFDiscover
Dimensions	TargetNfType, NrfLevel, NfFqdn, ExceptionAttribute **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-158 ocnrf_max_slf_attempts_exhausted_total

Field	Details
Description	The maximum number of SLF attempts to the hosts present in the SLF configuration. This metric is pegged when either of the following reasons: • when `maxslfattempts` is configured to a value > 0 and the `maxslfattempts` is exhausted. • when `maxslfattempts` is configured to a value =0, and the number of SLF attempts have exceeded `maximumHopCount`. • when the available SLF hosts are exhausted.
Type	Counter
Service Operation	NFDiscover

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-158 (Cont.) ocnrf_max_slf_attempts_exhausted_total

Field	Details
Dimensions	TargetNfType, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-159 ocnrf_slf_jetty_latency_seconds

Field	Details
Description	Time (in microseconds) after sending query to SLF and getting response from SLF.
Type	Histogram
Service Operation	NFDiscover
Dimensions	Host, InstanceIdentifier, Method, NrfLevel, ServiceOperation

Table 6-160 ocnrf_SLF_reroutes_total

Field	Details
Description	The total number of SLF query which is rerouted to alternate SLFs.
Type	Counter
Service Operation	NFDiscover
Dimensions	RequesterNfType, TargetNfType, TargetSLFFqdn, and PreviousSLFFqdn

6.1.9 NRF Forwarding Metrics

This section provides details about the NRF forwarding metrics.

Table 6-161 ocnrf_forward_accessToken_tx_requests_total

Field	Details
Description	The total number of Access Token Request forwarded to Primary or Secondary NRF.
Type	Counter
Service Operation	AccessToken
Dimensions	TargetNfType, RequesterNfType, TargetNfInstanceId, ClientNfInstanceId, Scope, NrfLevel, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-162 ocnrf_forward_accessToken_rx_responses_total

Field	Details
Description	The total number of Access Token Responses for request forwarded to Primary/Secondary NRF
Type	Counter
Service Operation	AccessToken

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-162 (Cont.) ocnrf_forward_accessToken_rx_responses_total

Field	Details
Dimensions	TargetNfType,RequesterNfType, TargetNfInstanceId, ClientNfInstanceId, Scope, NrfLevel, HttpStatusCode, RejectionReason, NfFqdn **Note** The NfFqdn dimension indicates the FQDN of the consumer NF. This dimension will contain the FQDN if the service mesh sends the FQDN of the consumer NF in the XFCC header, else the dimension contains the value as UNKNOWN. Rejection reason with meaning • InternalError: Request not forwarded because failure occurred at NRF due to internal error. • NRFCommunicationFailure: Request not forwarded because NRF is not able to connect with another NRF. • ErrorFromNRF: Error response received from another NRF. • NRFForwardingConfigurationMissing: Request not forwarded due to missing forwarding configuration. • LoopDetected: Received request message is already processed by NRF during forwarding mechanism. • *NotApplicable is applicable for 2xx Status code

Table 6-163 ocnrf_forward_nfProfileRetrieval_tx_requests_total

Field	Details
Description	The total number of Profile Retrieval Request forwarded to Primary/Secondary NRF.
Type	Counter
Service Operation	NFProfileRetrieval
Dimensions	NrfLevel, NfInstanceId, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-164 ocnrf_forward_nfProfileRetrieval_rx_responses_total

Field	Details
Description	The total number of Profile Retrieval Responses for Request forwarded to Primary/ Secondary NRF.
Type	Counter
Service Operation	NFProfileRetrieval
Dimensions	NrfLevel, NfInstanceId, HttpStatusCode, RejectionReason, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-165 ocnrf_forward_nfStatusSubscribe_tx_requests_total

Field	Details
Description	The total number of Status Subscribe Request forwarded to Primary/Secondary NRF.
Type	Counter
Service Operation	NFStatusSubscribe, NFStatusUnsubscribe
Dimensions	NrfLevel, RequesterNfType, OperationType, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-166 ocnrf_forward_nfStatusSubscribe_rx_responses_total

Field	Details
Description	The total number of Responses for Status Subscribe Request forwarded to Primary/ Secondary NRF.
Type	Counter
Service Operation	NFStatusSubscribe, NFStatusUnsubscribe
Dimensions	NrfLevel, RequesterNfType, HttpStatusCode, OperationType, RejectionReason, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-167 ocnrf_forward_nfDiscover_tx_requests_total

Field	Details
Description	The total number of NF Discovery Request forwarded to Primary/Secondary NRF.
Type	Counter
Service Operation	NFDiscover
Dimensions	NrfLevel, TargetNfType, RequesterNfType, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-168 ocnrf_forward_nfDiscover_rx_responses_total

Field	Details
Description	The total number of Responses for NF Discovery Request forwarded to Primary/Secondary NRF.
Type	Counter
Service Operation	NFDiscover
Dimensions	NrfLevel, TargetNfType, RequesterNfType, HttpStatusCode, RejectionReason, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-169 ocnrf_forward_round_trip_time_seconds

Field	Details
Description	Time taken by NRF specific microservice to forward the message to other Primary or Secondary NRF with the service operation: NFProfileRetrieval/NFStatusSubscribe/ NfStatusUnsubscribe/AccessToken
Type	Histogram
Service Operation	NFStatusSubscribe, NFStatusUnsubscribe, NFProfileRetrieval, AccessToken
Dimensions	NrfLevel, RequesterNfType, serviceOperation, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-170 ocnrf_forward_nfDiscover_barred_total

Field	Details
Description	The total number of discovery requests barred from getting forwarded to another NRF.
Type	Counter
Service Operation	NFDiscover
Dimensions	NrfLevel, TargetNfType, RequesterNfType, NfFqdn, ForwardingBarringReason **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-171 ocnrf_forward_accessToken_barred_total

Field	Details
Description	The total number of access token requests barred from getting forwarded to another NRF.
Type	Counter
Service Operation	AccessToken
Dimensions	TargetNfType, RequesterNfType, TargetNfInstanceId, ClientNfInstanceId, Scope, NrfLevel, NfFqdn, ForwardingBarringReason **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-172 ocnrf_forward_nfStatusSubscribe_barred_total

Field	Details
Description	The total number of status subscribe requests barred from getting forwarded to another NRF.
Type	Counter
Service Operation	NFStatusSubscribe, NFStatusUnsubscribe
Dimensions	NrfLevel, RequesterNfType, OperationType, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-173 ocnrf_forward_profileRetrieval_barred_total

Field	Details
Description	The total number of profile retrieval requests barred from getting forwarded to another NRF.
Type	Counter
Service Operation	NFProfileRetrieval
Dimensions	NrfLevel, NfInstanceId, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-174 ocnrf_forward_jetty_latency_seconds

Field	Details
Description	Time (in microseconds) after sending query to SLF and getting response from SLF.
Type	Histogram
Service Operation	NFDiscover
Dimensions	Host, InstanceIdentifier, Method, NrfLevel, ServiceOperation

6.1.10 Roaming Support Metrics

This section provides details about the NRF roaming metrics.

Table 6-175 ocnrf_roaming_nfStatusSubscribe_rx_requests_total

Field	Details
Description	Number of inter-PLMN subscription creation requests received by NRF from NF Consumers (in case of vNRF) and hSEPP (in case of hNRF).
Type	Counter
Service Operation	NFStatusSubscribe
Dimensions	NrfLevel, RequesterPLMNId, TargetPLMNId, NfFqdn, NRFRole, OperationType **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-176 ocnrf_roaming_nfStatusSubscribe_tx_responses_total

Field	Details
Description	Number of inter-PLMN subscription creation responses sent by NRF.
Type	Counter
Service Operation	NFStatusSubscribe
Dimensions	NrfLevel, RequesterPLMNId, TargetPLMNId, NfFqdn, NRFRole, OperationType, HttpStatusCode, RejectionReason **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-177 ocnrf_roaming_nfStatusSubscribe_tx_requests_total

Field	Details
Description	Number of inter-PLMN subscription creation requests sent by vNRF towards hNRF.
Type	Counter
Service Operation	NFStatusSubscribe
Dimensions	NrfLevel, RequesterPLMNId, TargetPLMNId, NfFqdn, OperationType **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-178 ocnrf_roaming_nfStatusSubscribe_rx_responses_total

Field	Details
Description	Number of inter-PLMN subscription creation response received by vNRF from hNRF for routed or forwarded requests.
Type	Counter
Service Operation	NFStatusSubscribe
Dimensions	NrfLevel, RequesterPLMNId, TargetPLMNId, NfFqdn, OperationType, HttpStatusCode, FailureReason **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-179 ocnrf_roaming_nfStatusSubscribe_rx_requests_total

Field	Details
Description	Number of inter-PLMN subscription update requests received by NRF from NF Consumers (in case of vNRF) and hSEPP (in case of hNRF).
Type	Counter
Service Operation	NFStatusSubscribe - Update
Dimensions	NrfLevel, NfFqdn, NRFRole, OperationType **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-180 ocnrf_roaming_nfStatusSubscribe_tx_responses_total

Field	Details
Description	Number of inter-PLMN subscription update responses sent by NRF.
Type	Counter
Service Operation	NFStatusSubscribe - Update
Dimensions	NrfLevel, NfFqdn, NRFRole, OperationType, HttpStatusCode, RejectionReason **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-181 ocnrf_roaming_nfStatusSubscribe_tx_requests_total

Field	Details
Description	Number of inter-PLMN subscription update requests routed or forwarded by vNRF towards hNRF.
Type	Counter
Service Operation	NFStatusSubscribe - Update
Dimensions	NrfLevel, NfFqdn, OperationType **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-182 ocnrf_roaming_nfStatusSubscribe_rx_responses_total

Field	Details
Description	Number of inter-PLMN subscription update response received by vNRF from hNRF for routed/forwarded requests.
Type	Counter
Service Operation	NFStatusSubscribe - Update
Dimensions	NrfLevel, NfFqdn, OperationType, HttpStatusCode, FailureReason **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-183 ocnrf_roaming_nfStatusUnSubscribe_rx_requests_total

Field	Details
Description	Number of inter-PLMN subscription unsubscribe requests received by NRF from NF Consumers (in case of vNRF) and hSEPP (in case of hNRF).
Type	Counter
Service Operation	NFStatusUnSubscribe
Dimensions	NrfLevel, NfFqdn, NRFRole **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-184 ocnrf_roaming_nfStatusUnSubscribe_tx_responses_total

Field	Details
Description	Number of inter-PLMN subscription unsubscribe responses sent by NRF.
Type	Counter
Service Operation	NF Status UnSubscribe
Dimensions	NrfLevel, NfFqdn, NRFRole, HttpStatusCode, RejectionReason **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-185 ocnrf_roaming_nfStatusUnSubscribe_tx_requests_total

Field	Details
Description	Number of inter-PLMN subscription unsubscribe requests routed or forwarded by vNRF towards hNRF.
Type	Counter
Service Operation	NFStatusUnSubscribe
Dimensions	NrfLevel, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-186 ocnrf_roaming_nfStatusUnSubscribe_rx_responses_total

Field	Details
Description	Number of inter-PLMN subscription unsubscribe response received by vNRF from hNRF for routed or forwarded requests.
Type	Counter
Service Operation	NFStatusUnSubscribe
Dimensions	NrfLevel, NfFqdn, HttpStatusCode, FailureReason **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-187 ocnrf_roaming_nfDiscover_rx_requests_total

Field	Details
Description	Number of inter-PLMN NFDiscover requests received by NRF from NF Consumers (in case of vNRF) and hSEPP (in case of hNRF).
Type	Counter
Service Operation	NFDiscover
Dimensions	NrfLevel, RequesterPLMNId, TargetPLMNId, NfFqdn, NRFRole **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-188 ocnrf_roaming_nfDiscover_tx_responses_total

Field	Details
Description	Number of inter-PLMN NFDiscover response sent by NRF.
Type	Counter
Service Operation	NFDiscover
Dimensions	NrfLevel, RequesterPLMNId, TargetPLMNId, NfFqdn, NRFRole, HttpStatusCode, RejectionReason **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-189 ocnrf_roaming_nfDiscover_tx_requests_total

Field	Details
Description	Number of inter-PLMN NF Discover request routed or forwarded by vNRF towards hNRF
Type	Counter
Service Operation	NFDiscover
Dimensions	NrfLevel, RequesterPLMNId, TargetPLMNId, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-190 ocnrf_roaming_nfDiscover_rx_responses_total

Field	Details
Description	Number of inter-PLMN NFDiscover response received by vNRF from hNRF for routed or forwarded requests.
Type	Counter
Service Operation	NFDiscover
Dimensions	NrfLevel, RequesterPLMNId, TargetPLMNId, NfFqdn, HttpStatusCode, FailureReason **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-191 ocnrf_roaming_accessToken_rx_requests_total

Field	Details
Description	Number of inter-PLMN Access Token requests received by NRF from NF Consumers (in case of vNRF) and hSEPP (in case of hNRF).
Type	Counter
Service Operation	AccessToken
Dimensions	NrfLevel, RequesterPLMNId, TargetPLMNId, NfFqdn, NRFRole **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-192 ocnrf_roaming_accessToken_tx_responses_total

Field	Details
Description	Number of inter-PLMN Access Token response sent by NRF
Type	Counter
Service Operation	AccessToken
Dimensions	NrfLevel, RequesterPLMNId, TargetPLMNId, NfFqdn, NRFRole, HttpStatusCode, RejectionReason **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-193 ocnrf_roaming_accessToken_tx_requests_total

Field	Details
Description	Number of inter-PLMN Access Token request routed or forwarded by vNRF towards hNRF.
Type	Counter
Service Operation	AccessToken
Dimensions	NrfLevel, RequesterPLMNId, TargetPLMNId, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-194 ocnrf_roaming_accessToken_rx_responses_total

Field	Details
Description	Number of inter-PLMN Access Token response received by vNRF from hNRF for routed or forwarded requests.
Type	Counter
Service Operation	AccessToken
Dimensions	NrfLevel, RequesterPLMNId, TargetPLMNId, NfFqdn, HttpStatusCode, FailureReason **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-195 ocnrf_roaming_nfStatusNotify_tx_requests_total

Field	Details
Description	Number of NFStatusNotify requests sent for inter_PLMN cases
Type	Counter
Service Operation	NFNotify
Dimensions	NrfLevel, NotificationEventType, TargetNfType, NfFqdn, SubscriptionId, SubscriberPLMNId **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-196 ocnrf_roaming_nfStatusNotify_rx_responses_total

Field	Details
Description	Number of NFStatusNotify responses received for inter_PLMN cases.
Type	Counter
Service Operation	NFNotify
Dimensions	NrfLevel, NotificationEventType, TargetNfType, NfFqdn, HttpStatusCode, SubscriptionId, SubscriberPLMNId **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-197 ocnrf_roaming_round_trip_time_seconds

Field	Details
Description	Average Latency between messages sent by visited NRF towards home NRF and after receiving response from home NRF for NRF Roaming messages.
Type	Counter
Service Operation	NFStatusSubscribe, NFStatusUnSubscribe, NfAccessToken
Dimensions	NrfLevel, RequesterPLMNId, TargetPLMNId, ServiceOperation, NfFqdn **Note**: The NfFQDN dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-198 ocnrf_roaming_jetty_latency_seconds

Field	Details
Description	Time (in microseconds) after sending query to SLF and getting response from SLF.
Type	Histogram
Service Operation	NFDiscover
Dimensions	Host, InstanceIdentifier, Method, NrfLevel, ServiceOperation

6.1.11 Georedundancy Metrics

This section provides details about the georedundancy metrics.

Table 6-199 ocnrf_dbreplication_status

Field	Details
Description	The current status of the replication channel between the given site and the remote site(s). This metric is pegged only if the georedundancy feature is enabled. Value 0 - UP Value 1 - DOWN
Type	Gauge
Service Operation	NA
Dimensions	NrfLevel, RemoteSiteName, RemoteNrfInstanceId

Table 6-200 ocnrf_dbreplication_down_time_seconds

Field	Details
Description	Time taken for the replication status to change from "INACTIVE" to "ACTIVE". This metric is pegged only if the georedundancy feature is enabled.
Type	Histogram
Service Operation	NA
Dimensions	NrfLevel, DbReplicationDownStartTime, DbReplicationDownEndTime, RemoteSiteName, RemoteNrfInstanceId

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-201 ocnrf_nf_switch_over_total

Field	Details
Description	The number of NFInstances that got switched over from the mated site.
Type	Counter
Service Operation	NfRegister, NfUpdate,NfDeregister, NfHeartbeat
Dimensions	NrfLevel, NfInstanceId, RemoteNrfInstanceId, Service_Operation,OperationType, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-202 ocnrf_nfSubscriptions_switch_over_total

Field	Details
Description	The number of NfSubscriptions that got switched over from the mated site.
Type	Counter
Service Operation	NfStatusSubscribe, NfStatusUnsubscribe, NrfAuditor
Dimensions	NrfLevel, RemoteNrfInstanceId, serviceOperation, OperationType

Table 6-203 ocnrf_stale_nf_deleted_total

Field	Details
Description	The number of NfInstances that get deleted by the NrfAuditor when it detects a record to be stale.
Type	Counter
Service Operation	NA
Dimensions	NrfLevel, NfInstanceId, NfStatus, NfFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-204 ocnrf_stale_nfSubscriptions_deleted_total

Field	Details
Description	The number of NfSubscriptions that get deleted by the NrfAuditor when it detects a record to be stale.
Type	Counter
Service Operation	NA
Dimensions	NrfLevel, SubscriptionStatus

Table 6-205 ocnrf_reported_dbreplication_status

Field	Details
Description	The current status of the replication channel between the given site and a remote site(s) as reported by the DB Monitor Service. This metric shall be pegged only if the georedundancy feature is enabled. Value 0 - UP Value 1 - DOWN
Type	Gauge
Service Operation	NA
Dimensions	NrfLevel, nrfInstanceId,SiteName

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-206 ocnrf_replication_status_check_total

Field	Details
Description	If the metric value remains constant over a period of time, an alert is raised.
Type	Counter
Service Operation	NA
Dimensions	NrfLevel

Table 6-207 ocnrf_replication_status_check_total

Field	Details
Description	This metric measures whether replication status monitoring thread is running in the following NRF microservices: • nfsubscription • nrfconfiguration • nfdiscovery • nfregistration • nfaccesstoken • nrfartisan This metric is pegged before the replication status monitoring performs the replication check as per the interval configured in `monitorDBReplicationStatusInterval` parameter under geoRedundancyOptions.
Type	Counter
Service Operation	NA
Dimensions	NrfLevel

6.1.12 NF Authentication Metrics

This section provides details about the network function authentication metrics.

Table 6-208 ocnrf_nf_authentication_failure_total

Field	Details
Description	The total number of request for which FQDN based authentication failed at OCNRF.
Type	Counter
Service Operation	NFAccessToken/NFRegistration/NFSubscription/NFDiscovery/NfListRetrieval/ NfProfileRetrieval For NfListRetrieval and NfProfileRetrieval serviceOperations NfFqdn is filled as NotApplicable. If OC-XFCC-DNS header is not received at NRF microservice then TLSFqdn is filled as "UNKNOWN".
Dimensions	NrfLevel, Method, ServiceOperation, NfFqdn, TLSFqdn **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

6.1.13 NF Instance and NF Services Status Change Metrics

This section provides details about the NF Instance and NF Services Status Change metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-209 ocnrf_nfInstance_status_change_total

Field	Details
Description	This metric indicates the total number of times the NfStatus changed for a given NfInstance. For each NfStatus change this metric will be pegged with a combination of NfInstanceId, PreviousStatus, and NewStatus (with other dimensions). For example, NfInstanceId-1 changed its NfStatus from SUSPENDED to REGISTERED, then the metric will be pegged with a combination of NfInstanceId-1, PreviousStatus as SUSPENDED, and NewStatus as REGISTERED. Similarly, NfInstanceId-1 changed its NfStatus from REGISTERED to UNDISCOVERABLE, then the metric will be pegged with a combination of NfInstanceId-1, PreviousStatus as REGISTERED, and NewStatus as UNDISCOVERABLE.
Type	Counter
Service Operation	NfRegister, NfUpdate, NfDeregister
Dimensions	NrfLevel, NfInstanceId, NfProfileFqdn, NfType, PreviousStatus, NewStatus

Table 6-210 ocnrf_nfService_status_change_total

Field	Details
Description	This metric indicates the total number of times the NfServiceStatus changed for a given NfServiceInstance of a NfProfile. For each NfServiceStatus change this metric will be pegged with a combination of NfInstanceId, NfServiceInstanceId, PreviousStatus, and NewStatus (with other dimensions). For example, NfInstanceId-1's NfServiceInstanceId-1 changed its NfServiceStatus from SUSPENDED to REGISTERED, then the metric will be pegged with a combination of NfInstanceId-1, NfServiceInstanceId-1, PreviousStatus as SUSPENDED, and NewStatus as REGISTERED. Similarly, NfInstanceId-1's NfServiceInstanceId-1 changed its NfServiceStatus from REGISTERED to UNDISCOVERABLE, then the metric will be pegged with a combination of NfInstanceId-1, NfServiceInstanceId-1, PreviousStatus as REGISTERED, and NewStatus as UNDISCOVERABLE.
Type	Counter
Service Operation	NfRegister, NfUpdate, NfDeregister
Dimensions	NrfLevel, NfInstanceId, NfServiceName, NfServiceInstanceId, NfProfileFqdn, NfServiceFqdn, PreviousStatus, NewStatus

6.1.14 DNS NAPTR Update Metrics

This section provides details about the DNS NAPTR update metrics.

NRF Microservices Metrics

Table 6-211 ocnrf_dns_naptr_tx_requests_total

Field	Details
Description	Number of DNS record update requests sent. **Note**: The value of OperationType dimension is Update.
Type	Counter
Service Operation	NFRegister, NFUpdate, NFDeregister, NRF Audit

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-211 (Cont.) ocnrf_dns_naptr_tx_requests_total

Field	Details
Dimensions	NrfLevel, NfInstanceId, NfFqdn, NfSetFqdn, Replacement, OperationType, TriggerMode Possible Values: • OperationType: – Update: NF update request towards DNS Server – Delete: NF delete request towards DNS Server • TriggerMode: – Automatic: Triggers generated by 3GPP service operations or NRF Audit. – Manual: Triggers generated by the operator initiated retrigger API. **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-212 ocnrf_dns_naptr_rx_responses_total

Field	Details
Description	Number of DNS record update responses received.
Type	Counter
Service Operation	NFRegister, NFUpdate, NFDeregister, NRF Audit
Dimensions	NrfLevel, NfInstanceId, NfFqdn, NfSetFqdn, Replacement, OperationType, HttpStatusCode, FailureReason, TriggerMode **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-213 ocnrf_dns_naptr_tx_requests_total

Field	Details
Description	Number of DNS record delete requests sent. **Note**: The value of OperationType dimension is Delete.
Type	Counter
Service Operation	NFRegister, NFUpdate, NFDeregister, NRF Audit
Dimensions	NrfLevel, NfInstanceId, NfFqdn, NfSetFqdn, Replacement, OperationType, HttpStatusCode, FailureReason, TriggerMode **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-214 ocnrf_dns_naptr_rx_responses_total

Field	Details
Description	Number of DNS record delete responses received. **Note**: The value of OperationType dimension is Delete.
Type	Counter
Service Operation	None
Dimensions	NrfLevel, NfInstanceId, NfFqdn, NfSetFqdn, Replacement, OperationType, HttpStatusCode, FailureReason, TriggerMode **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-215 ocnrf_dns_naptr_audit_tx_requests_total

Field	Details
Description	Number of DNS record audit requests sent. **Note**: The value of OperationType dimension is Get.
Type	Counter
Service Operation	None
Dimensions	NrfLevel, NFInstanceId, NfFqdn, NfSetFqdn, Replacement, OperationType **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-216 ocnrf_dns_naptr_audit_rx_responses_total

Field	Details
Description	Number of DNS record audit responses received. **Note**: The value of OperationType dimension is Get.
Type	Counter
Service Operation	None
Dimensions	NrfLevel, NFInstanceId, NfFqdn, NfSetFqdn, Replacement, OperationType, HttpStatusCode, FailureReason **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-217 ocnrf_dns_naptr_failure_rx_responses

Field	Details
Metric Name	ocnrf_dns_naptr_failure_rx_responses
Description	Number of DNS record failed responses received.
Type	Gauge
Service Operation	None
Dimensions	NrfLevel, NFInstanceId, NfFqdn, NfSetFqdn, Replacement **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-218 ocnrf_dns_naptr_round_trip_time_seconds

Field	Details
Description	Average Latency between messages sent by NRF towards DNS and after receiving response from DNS.
Type	Histogram
Service Operation	NFRegister, NFUpdate, NFDeregister, NRFAudit, DNS NAPTR Audit, Manual
Dimensions	NrfLevel, NFInstanceId, NfFqdn, NfSetFqdn, Replacement, OperationType, TriggerMode **Note**: The NfFqdn dimension contains the FQDN as present in the NfProfile.

Table 6-219 ocnrf_dns_naptr_nfRegistration_tx_requests_total

Field	Details
Description	Number of DNS record trigger sent by NFRegistration microservice to Artisan microservice.
Type	Counter

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-219 (Cont.) ocnrf_dns_naptr_nfRegistration_tx_requests_total

Field	Details
Service Operation	NFRegister, NFUpdate, NFDeregister
Dimensions	NfInstanceId, TriggerType, NrfLevel

Table 6-220 ocnrf_dns_naptr_nfRegistration_rx_responses_total

Field	Details
Description	Number of DNS record trigger response received by NFRegistration microservice from Artisan microservice.
Type	Counter
Service Operation	NFRegister, NFUpdate, NFDeregister
Dimensions	NfInstanceId, TriggerType, NrfLevel

Table 6-221 ocnrf_dns_naptr_nrfAuditor_tx_requests_total

Field	Details
Description	Number of DNS record trigger sent by NRFAuditor microservice to Artisan microservice.
Type	Counter
Service Operation	NRFAuditor
Dimensions	NfInstanceId, TriggerType, NrfLevel

Table 6-222 ocnrf_dns_naptr_nrfAuditor_rx_responses_total

Field	Details
Description	Number of DNS record trigger response received by NRFAuditor microservice from Artisan microservice.
Type	Counter
Service Operation	NRFAuditor
Dimensions	NfInstanceId, NrfLevel

Table 6-223 ocnrf_dns_naptr_trigger_rx_requests_total

Field	Details
Description	Number of DNS record trigger request received by Artisan microservice.
Type	Counter
Service Operation	None
Dimensions	NfInstanceId, NrfLevel

Table 6-224 ocnrf_dns_naptr_trigger_tx_responses_total

Field	Details
Description	Number of DNS record trigger responses sent by Artisan microservice.
Type	Counter
Service Operation	None

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-224 (Cont.) ocnrf_dns_naptr_trigger_tx_responses_total

Field	Details
Dimensions	NfInstanceId, NrfLevel

NRF Alternate Route Service Metrics

Table 6-225 oc_alternate_route_upstream_dns_request_timeout_total

Field	Details
Description	Number of DNS record requests timeout total.
Type	Counter
Service Operation	Service towards DNS server
Dimensions	amfFqdn, host, port, replacement

6.1.15 Pod Protection Metrics

This section provides details about the Pod Protection metrics.

Table 6-226 ocnrf_pod_congestion_state

Field	Details
Description	The metric is used to track the overall congestion state per nfSubscription pod. The values of the metric is as follow: • -1: NotApplicable • 0: Normal • 1: DangerOfCongestion • 2: Congested **Note**: If the pod protection feature is disabled, this metric will not be available or will contain value as -1.
Type	Gauge
Dimensions	NrfLevel

Table 6-227 ocnrf_pod_cpu_congestion_state

Field	Details
Description	The metric is used to track the congestion state based on the CPU usage of the subscription pod. This metric is valid only when the nfSubscription pod protection feature is enabled. The values of the metric is as follow: • -1: NotApplicable • 0: Normal • 1: DangerOfCongestion • 2: Congested **Note**: If the pod protection feature is disabled, this metric will not be available or will contain value as -1.
Type	Gauge
Dimensions	NrfLevel

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-228 ocnrf_pod_pending_message_count_congestion_state

Field	Details
Description	The metric is used to track the congestion state based on the message pending count of the subscription pod. This metric is valid only when the nfSubscription pod protection feature is enabled. The values of the metric is as follow: • -1: NotApplicable • 0: Normal • 1: DangerOfCongestion • 2: Congested **Note**: If the pod protection feature is disabled, this metric will not be available or will contain value as -1.
Type	Gauge
Dimensions	NrfLevel

Table 6-229 ocnrf_incoming_connections

Field	Details
Metric Name	ocnrf_incoming_connections
Description	The metric is used to track the number of incoming connections. This metric is applicable only for nfsubscription service.
Type	Gauge
Dimensions	NrfLevel

Table 6-230 ocnrf_max_concurrent_streams

Field	Details
Description	The metric is used to track the maximum concurrent streams per connection. This metric is applicable only for nfsubscription service.
Type	Gauge
Dimensions	NrfLevel , connectionId

Table 6-231 ocnrf_pod_cpu_usage

Field	Details
Description	The CPU usage of the pod. **Note**: If the pod protection feature is disabled, this metric will not be available or will contain value as 0.
Type	Gauge
Dimensions	NrfLevel

Table 6-232 ocnrf_pod_pending_message_count

Field	Details
Description	The Pending Message Count value of the pod. **Note**: If the pod protection feature is disabled, this metric will not be available or will contain value as 0.
Type	Gauge

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-232 (Cont.) ocnrf_pod_pending_message_count

Field	Details
Dimensions	NrfLevel

Table 6-233 ocnrf_pod_incoming_connection_rejected_total

Field	Details
Description	The number of connections that have been rejected in the DoC or Congested state. This metric is applicable only for nfsubscription service.
Type	Counter
Dimensions	NrfLevel

Table 6-234 ocnrf_nfNotification_trigger_total

Field	Details
Description	The number of times a notification trigger is sent to nfSubscription service from the nfregistration and nrfauditor services.
Type	Counter
Dimensions	NotificationEventType, errorReason: The reason if nfNotification trigger failed from nfregistration or nrfauditor service pods to nfsubscription service pods.

6.1.16 Network Slice Specific Metrics

This section describes the Network Slice Specific metrics that are pegged based on Network Slice Instances (NSIs) and Single Network Slice Selection Assistance Information (SNSSAI).

Table 6-235 ocnrf_nfDiscover_rx_requests_perSnssai_total

Field	Details
Description	The total number of NFDiscover requests received at a slice that is identified by specific SNSSAI. This metric is pegged for each `snssai` present in the search query parameters. The search query attributes are `snssais` and `plmn-specific-snssai-list`. **Note**: • `snssais` of NfService and NfServiceList are not considered for pegging. • The SNSSAI belongs to the Producer NF that is discovered. • The metric will not get pegged if any of the search query parameters are invalid.
Type	Counter
Service Operation	NFDiscover
Dimensions	sNssai

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-236 ocnrf_nfDiscover_tx_success_response_perSnssai_total

Field	Details
Description	The total number of success discovery responses with one or more matching NfProfiles sent per slice that is identified by SNSSAI. The metric is pegged for each unique `snssai` present in the profiles of discovery response. The search query attributes are `snssais` and `plmn-specific-snssai-list`. The NfProfile attributes are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: • The SNSSAI belongs to the Producer NF that is discovered. • Only `snssais` of NfProfile and xxxInfo are considered for pegging. • Response metric may not get pegged for all SNSSAI that is being received in the request message, if there is no matching profiles sent in discovery response for some SNSSAIs.
Type	Counter
Service Operation	NFDiscover
Dimensions	sNssai

Table 6-237 ocnrf_nfDiscover_tx_empty_response_perSnssai_total

Field	Details
Description	The total number of discovery responses sent with no matching NfProfiles for a slice identified by SNSSAI. This metric is pegged for each `snssai` present in the search query parameters. The search query attributes are `snssais` and `plmn-specific-snssai-list`. **Note**: • The SNSSAI belongs to the Producer NF that is discovered. • In this case, response metric will get pegged for all SNSSAI that is being received in the request message.
Type	Counter
Service Operation	NFDiscover
Dimensions	sNssai

Table 6-238 ocnrf_nfDiscover_tx_failure_response_perSnssai_total

Field	Details
Description	The total number of failure discovery responses sent per slice that is identified by specific SNSSAI. This metric is pegged for each `snssai` present in the search query parameters. The search query attributes are `snssais` and `plmn-specific-snssai-list`. **Note**: • The SNSSAI belongs to the Producer NF that is discovered. • In this case, response metric will get pegged for all SNSSAI that is being received in the request message. • The metric will not get pegged if any of the search query parameters are invalid.
Type	Counter
Service Operation	NFDiscover
Dimensions	sNssai, HttpStatusCode

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-239 ocnrf_nfDiscover_rx_requests_perNsi_total

Field	Details
Description	The total number of NFDiscover requests received at a slice that is identified by NSI. This metric is pegged for each `nsi` present in the search query parameter. The search query attribute is `nsi-list`. **Note**: • The NSI belongs to the Producer NF that is discovered. • The metric will not get pegged if any of the search query parameters are invalid.
Type	Counter
Service Operation	NFDiscover
Dimensions	Nsi

Table 6-240 ocnrf_nfDiscover_tx_success_response_perNsi_total

Field	Details
Description	The total number of success discovery responses with one or more matching NfProfiles sent per slice that is identified by specific NSI. The metric is pegged for each `nsi` in the intersection of `nsi-list` search query parameter and `nsiList` attribute in the NfProfile present in the discovery response. **Note**: • The NSI belongs to the Producer NF that is discovered. • The metric will not get pegged if any of the search query parameters are invalid. • Response metric may not get pegged for all NSI that is being received in the request message, if there is no matching profiles sent in discovery response for some NSIs.
Type	Counter
Service Operation	NFDiscover
Dimensions	Nsi

Table 6-241 ocnrf_nfDiscover_tx_empty_response_perNsi_total

Field	Details
Description	The total number of discovery response with no matching NfProfiles for a slice identified by NSI. The metric is pegged for each `nsi` present in the search query parameter. The search query attribute is `nsi-list`. **Note**: • The NSI belongs to the Producer NF that is discovered. • The metric will not get pegged if any of the search query parameters are invalid. • In this case, response metric will get pegged for all SNSSAI that is being received in the request message.
Type	Counter
Service Operation	NFDiscover
Dimensions	Nsi

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-242 ocnrf_nfDiscover_tx_failure_response_perNsi_total

Field	Details
Description	The total number of failure discovery response for a slice identified by NSI. The metric is pegged for each `nsi` present in the search query parameter. The search query attribute is `nsi-list`. **Note**: • The NSI belongs to the Producer NF that is discovered. • The metric will not get pegged if any of the search query parameters are invalid. • In this case, response metric will get pegged for all SNSSAI that is being received in the request message.
Type	Counter
Service Operation	NFDiscover
Dimensions	Nsi, HttpStatusCode

Table 6-243 ocnrf_nfDiscover_tx_forwarded_requests_perSnssai_total

Field	Details
Description	The total number of discovery requests forwarded to another NRF for a slice identified by SNSSAI. The metric is pegged for each `snssai` present in the search query parameters. The search query attributes are `snssais` and `plmn-specific-snssai-list`. **Note**: • Only `snssais` of NfProfile and xxxInfo are considered for pegging. • The SNSSAI belongs to the Producer NF that is discovered.
Type	Counter
Service Operation	NFDiscover
Dimensions	sNssai, TargetNrfFqdn, NfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-244 ocnrf_nfDiscover_rx_success_forwarded_responses_perSnssai_total

Field	Details
Description	The total number of success responses with one or more matching NfProfiles that is received for forwarded discovery request for a slice identified by SNSSAI. The metric is pegged for each `snssai` present in the intersection of the `snssai` present in the search query and in the profile. The search query attributes are `snssais` and `plmn-specific-snssai-list`. The NfProfile attributes are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: • The SNSSAI is associated with producer NF being discovered. • Only `snssais` of NfProfile and xxxInfo are considered for pegging. • Response metric may not get pegged for all SNSSAI that is being received in the request message, if there is no matching profiles sent in discovery response for some SNSSAIs.
Type	Counter
Service Operation	NFDiscover

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-244 (Cont.) ocnrf_nfDiscover_rx_success_forwarded_responses_perSnssai_total

Field	Details
Dimensions	sNssai, NfFqdn, TargetNrfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-245 ocnrf_nfDiscover_rx_empty_forwarded_responses_perSnssai_total

Field	Details
Description	The total number of responses that is received for forwarded discovery request with no matching NfProfiles for a slice identified by SNSSAI. The metric is pegged for each `snssai` present in the search query parameters. The search query attributes are `snssais` and `plmn-specific-snssai-list`. **Note**: • The SNSSAI belongs to the Producer NF that is discovered. • The metric will not get pegged if any of the search query parameters are invalid. • In this case, response metric will get pegged for all SNSSAI that is being received in the request message.
Type	Counter
Service Operation	NFDiscover
Dimensions	sNssai, NfFqdn, TargetNrfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-246 ocnrf_nfDiscover_rx_failure_forwarded_responses_perSnssai_total

Field	Details
Description	The total number of failure responses for forwarded discovery requests for a slice identified by SNSSAI. The metric is pegged for each `snssai` present in the search query parameters. The search query attributes are `snssais` and `plmn-specific-snssai-list`. **Note**: • The SNSSAI belongs to the Producer NF that is discovered. • The metric will not get pegged if any of the search query parameters are invalid. • In this case, response metric will get pegged for all SNSSAI that is being received in the request message.
Type	Counter
Service Operation	NFDiscover
Dimensions	sNssai, NfFqdn, HttpStatusCode, TargetNrfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-247 ocnrf_nfDiscover_tx_forwarded_requests_perNsi_total

Field	Details
Description	The total number of discovery requests forwarded to another NRF for a slice identified by NSI. The metric is pegged for each `nsi` present in the search query parameter. The search query attribute is `nsi-list`. **Note**: The NSI belongs to the Producer NF that is discovered.
Type	Counter
Service Operation	NFDiscover
Dimensions	Nsi, NfFqdn, TargetNrfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-248 ocnrf_nfDiscover_rx_success_forwarded_responses_perNsi

Field	Details
Description	The total number of success response for forwarded discovery requests for a slice identified by NSI. The NSI associated with producer NF being discovered. The metric is pegged for each `nsi` in the intersection of `nsi-list` search query parameter and `nsiList` attribute in the NfProfiles present in the discovery response. **Note**: • The NSI is associated with producer NF being discovered. • Only `nsi` of NfProfile and xxxInfo are considered for pegging. • Response metric may not get pegged for all NSI that is being received in the request message, if there is no matching profiles sent in discovery response for some NSIs.
Type	Counter
Service Operation	NFDiscover
Dimensions	Nsi, NfFqdn, TargetNrfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-249 ocnrf_nfDiscover_rx_empty_forwarded_responses_perNsi

Field	Details
Description	The total number of responses with no matching NfProfiles for forwarded discovery request for a slice identified by NSI. The NSI is associated with the producer NF being discovered. The metric is pegged for each `nsi` present in the search query parameter. The search query attribute is `nsi-list`. **Note**: • The NSI belongs to the Producer NF that is discovered. • The metric will not get pegged if any of the search query parameters are invalid. • In this case, response metric will get pegged for all NSI that is being received in the request message.
Type	Counter
Service Operation	NFDiscover
Dimensions	Nsi, NfFqdn, TargetNrfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-250 ocnrf_nfDiscover_rx_failure_forwarded_responses_perNsi_total

Field	Details
Description	The total number of failure response for forwarded discovery requests for a slice identified by NSI. The metric is pegged for each `nsi` present in the search query parameter. The search query attribute is `nsi-list`. **Note**: • The NSI belongs to the Producer NF that is discovered. • The metric will not get pegged if any of the search query parameters are invalid. • In this case, response metric will get pegged for all NSI that is being received in the request message.
Type	Counter
Service Operation	NFDiscover
Dimensions	Nsi, NfFqdn, HttpStatusCode, TargetNrfFqdn **Note**: The NfFqdn dimension indicates the consumer NF FQDN. This dimension contains the NfFqdn if the service mesh sends the consumer NF FQDN in XFCC header, else the dimension will contain the value as UNKNOWN.

Table 6-251 ocnrf_nfRegister_requests_perSnssai_total

Field	Details
Description	The total number of NFRegister requests received at a slice that is identified by specific SNSSAI. The metric is pegged for each `snssai` present in the NfProfile being registered. The attributes considered are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: • `snssais` of NfService and NfServiceList are not considered for pegging. • If the NFRegister request has an invalid body, the metric may not get pegged.
Type	Counter
Service Operation	NFRegister
Dimensions	sNssai

Table 6-252 ocnrf_nfRegister_success_responses_perSnssai_total

Field	Details
Description	The total number of successful NFRegister responses with one or more matching NfProfiles sent per slice identified by SNSSAI. The metric is pegged for each `snssai` present in the NfProfile being registered. The attributes considered are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: Only `snssais` of NfProfile and xxxInfo are considered for pegging.
Type	Counter
Service Operation	NFRegister
Dimensions	sNssai

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-253 ocnrf_nfRegister_failure_responses_perSnssai_total

Field	Details
Description	The total number of failure NFRegister responses sent per slice identified by SNSSAIs. The metric is pegged for each `snssai` present in the NfProfile being registered. The attributes considered are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: • Only `snssais` of NfProfile and xxxInfo are considered for pegging. • The metric may not get pegged if the NFRegister request has an invalid body.
Type	Counter
Service Operation	NFRegister
Dimensions	sNssai, HttpStatusCode

Table 6-254 ocnrf_nfRegister_requests_perNsi_total

Field	Details
Description	The total number of NFRegister requests received at a slice that is identified by NSI. The metric is pegged for each `nsi` present in the NfProfile being registered. The attribute considered is `nsiList`. **Note**: The metric may not get pegged if the NFRegister request has an invalid body.
Type	Counter
Service Operation	NFRegister
Dimensions	Nsi

Table 6-255 ocnrf_nfRegister_success_responses_perNsi_total

Field	Details
Description	The total number of successful NFRegister responses with one or more matching NfProfiles sent per slice that is identified by NSI. The metric is pegged for each `nsi` present in the NfProfile being registered. The attribute considered is `nsiList`.
Type	Counter
Service Operation	NFRegister
Dimensions	Nsi

Table 6-256 ocnrf_nfRegister_failure_responses_perNsi_total

Field	Details
Description	The total number of failure NFRegister responses sent per slice that is identified by NSI. The metric is pegged for each `nsi` present in the NfProfile being registered. The attribute considered is `nsiList`. **Note**: The metric may not get pegged if the NFRegister request has an invalid body.
Type	Counter
Service Operation	NFRegister
Dimensions	Nsi, HttpStatusCode

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-257 ocnrf_nfUpdate_requests_perSnssai_total

Field	Details
Description	The total number of NFUpdate requests for slices identified by SNSSAIs. The metric is pegged for Complete Update and Partial Update(PATCH) of a registered NfProfile and for each `snssais` present in the NfProfile registered. The attributes considered are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: • Only `snssais` of NfProfile and xxxInfo are considered for pegging. • If the NFUpdate request has an invalid body, the metric may not get pegged. • In case of partial update (PATCH), the metric is pegged only based on the final list of SNSSAI updated in the profile. • In case of NfUpdate (PATCH), if the profile is not found, the metric will not be pegged.
Type	Counter
Service Operation	NFUpdate
Dimensions	sNssai

Table 6-258 ocnrf_nfUpdate_success_responses_perSnssai_total

Field	Details
Description	The total number of successful NFUpdate responses with one or more matching NfProfiles sent per slice identified by SNSSAI. The metric is pegged for Complete Update and Partial Update(PATCH) of a registered NfProfile and for each `snssais` present in the NfProfile registered. The attributes considered are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: • `snssais` of NfService and NfServiceList are not considered for pegging. • If the NFUpdate request has an invalid body, the metric may not get pegged. • In case of partial update (PATCH), the metric is pegged only based on the final list of SNSSAI updated in the profile. • In case of NfUpdate (PATCH), if the profile is not found, the metric will not be pegged.
Type	Counter
Service Operation	NFUpdate
Dimensions	sNssai

Table 6-259 ocnrf_nfUpdate_failure_responses_perSnssai_total

Field	Details
Description	The total number of failure NFUpdate responses sent per slice identified by SNSSAIs. The metric is pegged for Complete Update and Partial Update(PATCH) of a registered NfProfile and for each `snssais` present in the NfProfile registered. The attributes considered are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: • Only `snssais` of NfProfile and xxxInfo are considered for pegging. • If the NFUpdate request has an invalid body, the metric may not get pegged. • In case of partial update (PATCH), the metric is pegged only based on the final list of SNSSAI updated in the profile. • In case of NfUpdate (PATCH), if the profile is not found, the metric will not be pegged.
Type	Counter
Service Operation	NFUpdate
Dimensions	sNssai, HttpStatusCode

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-260 ocnrf_nfUpdate_requests_perNsi_total

Field	Details
Description	The total number of NFUpdate requests for slices identified by NSI. The metric is pegged for Complete Update and Partial Update (PATCH) of a registered NfProfile and for each `nsi` present in the NfProfile registered. The attribute considered is `nsiList`. **Note**: • Only `nsi` of NfProfile and xxxInfo are considered for pegging. • If the NFUpdate request has an invalid body, the metric may not get pegged. • In case of partial update (PATCH), the metric is pegged only based on the updated profile. • In case of NfUpdate (PATCH), if the profile is not found, the metric will not be pegged.
Type	Counter
Service Operation	NFUpdate
Dimensions	Nsi

Table 6-261 ocnrf_nfUpdate_success_responses_perNsi_total

Field	Details
Description	The total number of successful NFUpdate responses for slices identified by NSI. The metric is pegged for Complete Update and Partial Update (PATCH) of a registered NfProfile and for each `nsi` present in the NfProfile registered. The attribute considered is `nsiList`. **Note**: • Only `nsi` of NfProfile and xxxInfo are considered for pegging. • If the NFUpdate request has an invalid body, the metric may not get pegged. • In case of partial update (PATCH), the metric is pegged only based on the final list of SNSSAI updated in the profile. • In case of NfUpdate (PATCH), if the profile is not found, the metric will not be pegged.
Type	Counter
Service Operation	NFUpdate
Dimensions	Nsi

Table 6-262 ocnrf_nfUpdate_failure_responses_perNsi_total

Field	Details
Description	The total number of failure NFUpdate responses for slices identified by NSI. The metric is pegged for Complete Update and Partial Update (PATCH) of a registered NfProfile and for each `nsi` present in the NfProfile registered. The attribute considered is `nsiList`. **Note**: • Only `nsi` of NfProfile and xxxInfo are considered for pegging. • If the NFUpdate request has an invalid body, the metric may not get pegged. • In case of partial update (PATCH), the metric is pegged only based on the final list of SNSSAI updated in the profile. • In case of NfUpdate (PATCH), if the profile is not found, the metric will not be pegged.
Type	Counter
Service Operation	NFUpdate
Dimensions	Nsi, HttpStatusCode

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-263 ocnrf_nfDeregister_requests_perSnssai_total

Field	Details
Description	The total number of NFDeregister requests received at a slice that is identified by SNSSAIs. The metric is pegged for each `snssais` present in the registered NfProfile. The attributes considered are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: • Only `snssais` of NfProfile and xxxInfo are considered for pegging. • If the profile is not found in the database or if the profile is in deregistered state, the metric will not get pegged.
Type	Counter
Service Operation	NFDeregister
Dimensions	sNssai

Table 6-264 ocnrf_nfDeregister_success_responses_perSnssai_total

Field	Details
Description	The total number of successful NFDeregister responses with one or more matching NfProfiles sent per slice that is identified by SNSSAI. The metric is pegged for each `snssais` present in the registered NfProfile. The attributes considered are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: • Only `snssais` of NfProfile and xxxInfo are considered for pegging. • If the profile is not found in the database or if the profile is in deregistered state, the metric will not get pegged.
Type	Counter
Service Operation	NFDeregister
Dimensions	sNssai

Table 6-265 ocnrf_nfDeregister_failure_responses_perSnssai_total

Field	Details
Description	The total number of failure NFDeregister responses sent per slice that is identified by SNSSAIs. The metric is pegged for each `snssais` present in the registered NfProfile. The attributes considered are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: • Only `snssais` of NfProfile and xxxInfo are considered for pegging. • If the profile is not found in the database or if the profile is in deregistered state, the metric will not get pegged.
Type	Counter
Service Operation	NFDeregister
Dimensions	sNssai, HttpStatusCode

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-266 ocnrf_nfDeregister_requests_perNsi_total

Field	Details
Description	The total number of NFDeregister requests received at a slice that is identified by NSI. The metric is pegged for each `nsi` present in the registered NfProfile. The attribute considered is `nsiList`. **Note**: • Only `nsi` of NfProfile and xxxInfo are considered for pegging. • If the profile is not found in the database or if the profile is in deregistered state, the metric will not get pegged.
Type	Counter
Service Operation	NFDeregister
Dimensions	Nsi

Table 6-267 ocnrf_nfDeregister_success_responses_perNsi_total

Field	Details
Description	The total number of successful NFDeregister responses with one or more matching NfProfiles sent per slice that is identified by NSI. The metric is pegged for each `nsi` present in the registered NfProfile. The attribute considered is `nsiList`. **Note**: • Only `nsi` of NfProfile and xxxInfo are considered for pegging. • If the profile is not found in the database or if the profile is in deregistered state, the metric will not get pegged.
Type	Counter
Service Operation	NFDeregister
Dimensions	Nsi

Table 6-268 ocnrf_nfDeregister_failure_responses_perNsi_total

Field	Details
Description	The total number of failure NFDeregister responses sent per slice that is identified by NSI. The metric is pegged for each `nsi` present in the registered NfProfile. The attribute considered is `nsiList`. **Note**: • Only `nsi` of NfProfile and xxxInfo are considered for pegging. • If the profile is not found in the database or if the profile is in deregistered state, the metric will not get pegged.
Type	Counter
Service Operation	NFDeregister
Dimensions	Nsi, HttpStatusCode

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-269 ocnrf_nfHeartBeat_requests_perSnssai_total

Field	Details
Description	The total number of NF HeartBeat requests received at a slice that is identified by SNSSAIs. The attributes considered are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: • Only `snssais` of NfProfile and xxxInfo are considered for pegging. • The metric will not get pegged if the nfHeartBeat request has an invalid body.
Type	Counter
Service Operation	NfHeartBeat
Dimensions	sNssai

Table 6-270 ocnrf_nfHeartBeat_success_responses_perSnssai_total

Field	Details
Description	The total number of success NF HeartBeat responses with one or more matching NfProfiles sent per slice that is identified by SNSSAIs. The attributes considered are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: Only `snssais` of NfProfile and xxxInfo are considered for pegging.
Type	Counter
Service Operation	NfHeartBeat
Dimensions	sNssai

Table 6-271 ocnrf_nfHeartBeat_failure_responses_perSnssai_total

Field	Details
Description	The total number of failure NF HeartBeat Update responses for slices identified by SNSSAIs. The attributes considered are `snssais`, `perPlmnSnssaiList`, `sNssaiSmfInfoList`, and `sNssaiUpfInfoList`. **Note**: • `snssais` of NfService and NfServiceList are not considered for pegging. • The metric will not get pegged if the nfHeartBeat request has an invalid body.
Type	Counter
Service Operation	NfHeartBeat
Dimensions	sNssai, HttpStatusCode

Table 6-272 ocnrf_nfHeartBeat_requests_perNsi_total

Field	Details
Description	The total number of NF HeartBeat requests received at a slice that is identified by Nsi. The attribute considered is `nsiList`. **Note**: • Only `nsi` of NfProfile and xxxInfo are considered for pegging. • The metric will not get pegged if the nfHeartBeat request has an invalid body.
Type	Counter
Service Operation	NfHeartBeat
Dimensions	Nsi

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-273 ocnrf_nfHeartBeat_success_responses_perNsi_total

Field	Details
Description	The total number of successful NF HeartBeat responses with one or more matching NfProfiles sent per slice that is identified by Nsi. The attribute considered is `nsiList`. **Note**: Only `nsi` of NfProfile and xxxInfo are considered for pegging.
Type	Counter
Service Operation	NfHeartBeat
Dimensions	Nsi

Table 6-274 ocnrf_nfHeartBeat_failure_responses_perNsi_total

Field	Details
Description	The total number of failure NF HeartBeat responses sent per slice that is identified by Nsi. The attribute considered is `nsiList`.
Type	Counter
Service Operation	NfHeartBeat
Dimensions	Nsi, HttpStatusCode

6.1.17 NRF Cache Data Metrics

Table 6-275 ocnrf_cds_rx_requests_total

Field	Details
Description	The number of requests received at Cache Data Service.
Type	Counter
Dimensions	NrfLevel, RequestType Possible values for RequestType are RemoteAndLocalSetProfiles, ProfilesByQueryParam, and RemoteAndLocalSetSubscriptions

Table 6-276 ocnrf_cds_tx_responses_total

Field	Details
Description	The number of responses sent by the Cache Data Service.
Type	Counter
Dimensions	NrfLevel, RequestType, HttpStatusCode. Possible values for RequestType are RemoteAndLocalSetProfiles, ProfilesByQueryParam, and RemoteAndLocalSetSubscriptions

Table 6-277 ocnrf_cds_round_trip_time_seconds

Field	Details
Description	The time taken to process the request at Cache Data Service in 50, 90, 95, and 99 percentiles.
Type	Histogram

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-277 (Cont.) ocnrf_cds_round_trip_time_seconds

Field	Details
Dimensions	NrfLevel, RequestType Possible values for RequestType are RemoteAndLocalSetProfiles, ProfilesByQueryParam, and RemoteAndLocalSetSubscriptions

Table 6-278 ocnrf_query_remote_cds_requests_total

Field	Details
Description	The number of requests sent to the NRF of a remote NRF set by Cache Data Service.
Type	Counter
Dimensions	NrfLevel, QueryType, RemoteNrfFqdn, TargetNrfSetId. Possible values of QueryType are NfProfiles and NfSubscriptions.

Table 6-279 ocnrf_query_remote_cds_responses_total

Field	Details
Description	The number of responses received from the NRF of a remote NRF set by Cache Data Service.
Type	Counter
Dimensions	NrfLevel, QueryType, HttpStatusCode, RemoteNrfFqdn, TargetNrfSetId. Possible values of QueryType are NfProfiles and NfSubscriptions.

Table 6-280 ocnrf_query_remote_cds_round_trip_time_seconds

Field	Details
Description	The time taken between the request sent by Cache Data Service towards remote NRF set and response received. It is measured in 50, 90, 95, and 99 percentiles.
Type	Histogram
Dimensions	NrfLevel, QueryType, HttpStatusCode, RemoteNrfFqdn, TargetNrfSetId. Possible values of QueryType are NfProfiles and NfSubscriptions.

Table 6-281 ocnrf_query_remote_cds_message_size_bytes

Field	Details
Description	The size of the response received by the Cache Data Service from the remote NRF set. The size is pegged in bytes and is published in buckets of 1kB, 5KB, 10KB, 50KB, 100KB, 500KB, and 1MB.
Type	Histogram
Dimensions	NrfLevel, QueryType, RemoteNrfFqdn, HttpStatusCode, SourceNrfSetId, TargetNrfSetId. Possible values of QueryType are NfProfiles and NfSubscriptions.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-282 ocnrf_cache_fallback_total

Field	Details
Description	The number of times the request was fulfilled from the in-memory cache of Cache Data Service due to cnDBTier query errors.
Type	Counter
Dimensions	NrfLevel, ErrorReason

Table 6-283 ocnrf_db_fallback_total

Field	Details
Description	The number of times a service request was fulfilled from the cnDBTier using local NRF set data as the query towards the Cache Data Service failed.
Type	Counter
Dimensions	NrfLevel, ServiceOperation Possible values of ServiceOperation are NfProfileRetrieval, NfListRetrieval, NfAccessToken, NfDiscover, NfStatusSubscribe, NfStatusUnsubscribe, and NfStatusNotify

Table 6-284 ocnrf_query_cds_requests_total

Field	Details
Description	The number of requests to the Cache Data Service sent by the local NRF microservices for state data. For the list of local NRF microservices, see the Enhanced NRF Set Based Deployment (NRF Growth) section.
Type	Counter
Dimensions	NrfLevel, ServiceOperation Possible values of ServiceOperation are NfProfileRetrieval, NfListRetrieval, NfAccessToken, NfDiscover, NfSubscribe, NfUnsubscribe, notifyProfileChange, notifyProfileDeregistration, notifyProfileRegistration, FindAllNfSubscription, NFInstanceFetch, CUSTOM_NRF_AUDIT_NF_COUNT, and CUSTOM_NRF_AUDIT_NF_SIZE.

Table 6-285 ocnrf_query_cds_responses_total

Field	Details
Description	The number of responses from the Cache Data Service to the local NRF microservices for state data. For the list of local NRF microservices, see the Enhanced NRF Set Based Deployment (NRF Growth) section.
Type	Counter
Dimensions	NrfLevel, HttpStatusCode, ServiceOperation Possible values of ServiceOperation are NfProfileRetrieval, NfListRetrieval, NfAccessToken, NfDiscover, NfSubscribe, NfUnsubscribe, notifyProfileChange, notifyProfileDeregistration, notifyProfileRegistration, FindAllNfSubscription, NFInstanceFetch, CUSTOM_NRF_AUDIT_NF_COUNT, and CUSTOM_NRF_AUDIT_NF_SIZE.

Table 6-286 ocnrf_query_cds_round_trip_time_seconds

Field	Details
Description	The time taken between the request sent by local NRF set microservices to Cache Data Service and response received. It is measured in 50, 95, and 99 percentiles.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-286 (Cont.) ocnrf_query_cds_round_trip_time_seconds

Field	Details
Type	Histogram
Dimensions	NrfLevel, ServiceOperation Possible values of ServiceOperation are NfProfileRetrieval, NfListRetrieval, NfAccessToken, NfDiscover, NfSubscribe, NfUnsubscribe, notifyProfileChange, notifyProfileDeregistration, notifyProfileRegistration, FindAllNfSubscription, NFInstanceFetch, CUSTOM_NRF_AUDIT_NF_COUNT, and CUSTOM_NRF_AUDIT_NF_SIZE.

Table 6-287 ocnrf_dbmetrics_total

Field	Details
Description	The number of database operation counts corresponding to every service operation.
Type	Counter
Dimensions	method, dbOperation, ServiceOperation, NrfLevel, HttpStatusCode Possible values for: • dbOperation: find • ServiceOperation: nrfCacheDataService

Table 6-288 ocnrf_nf_registered_count

Field	Details
Description	The total number of NFs registered in the segment.
Type	Gauge
Dimensions	NrfLevel, RequesterNfType

Table 6-289 ocnrf_cache_sync_count_total

Field	Details
Description	The number of in-memory cache entries synchronized. The metric is pegged at the at the Cache Data Service and Discovery Service.
Type	Counter
Dimensions	NrfLevel, NrfFqdn, SourceNrfSetId, TargetNrfSetId, DataType, CompleteSync Possible values of DataType are NfProfiles and NfSubscriptions Possible values of CompleteSync are true and false

Table 6-290 ocnrf_remote_set_unavailable_total

Field	Details
Description	All the NRFs of a remote NRF set are unavailable for synchronizing state data.
Type	Counter
Dimensions	NrfLevel,SourceNrfSetId, TargetNrfSetId

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-291 ocnrf_all_remote_sets_unavailable_total

Field	Details
Description	All the NRFs of all remote NRF sets are unavailable for synchronizing state data.
Type	Counter
Dimensions	NrfLevel,SourceNrfSetId, TargetNrfSetId

Table 6-292 ocnrf_cds_rx_remote_requests_total

Field	Details
Description	The number of requests received at Cache Data Service from remote NRFs.
Type	Counter
Dimensions	QueryType

Table 6-293 ocnrf_cds_rx_remote_responses_total

Field	Details
Description	The number of responses sent by the Cache Data Service to the remote NRF.
Type	Counter
Dimensions	QueryType

Table 6-294 ocnrf_cds_remote_rx_request_round_trip_time

Field	Details
Description	The time taken to process the request received from the Remote NRF at Cache Data Service.
Type	Histogram
Dimensions	QueryType

6.1.18 Overload Control

Table 6-295 load_level

Field	Details
Description	The metric is used to track the load level as per the services.
Type	Gauge
Dimensions	service, isLeaderPod

Table 6-296 service_resource_stress

Field	Details
Description	This metric tracks CPU, memory, failure count, and pending requests on the basis of which the overload level of a service is calculated.
Type	Gauge
Dimension	type, service, isLeaderPod

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-297 service_resource_overload_level

Field	Details
Description	This metric tracks an individual resource's overload level that is calculated based on the resource usage and configured threshold.
Type	Gauge
Dimension	type, service, isLeaderPod

6.2 NRF KPIs

This section includes information about KPIs for Oracle Communications Cloud Native Core, Network Repository Function (NRF).

> **Note**
>
> Sample NRF dashboard for Grafana is delivered to the customer through NRF CSAR package. Metrics and functions used to achieve KPI are already covered in NRF CSAR package.

6.2.1 NRF Service KPIs

This section lists the service status, Gateway, Service latency, Database latency KPIs.

Table 6-298 Registration Service Success Ratio (2xx) %

Field	Details
Description	Measures the total number of successful registration service operations at NRF. For more information about metric used, see NRF Gateways Metrics.
Expression	**Expression**: sum(increase(oc_ingressgateway_http_responses_total{Status=~\"2.*\",Route_path=~\".*nnrf-nfm/v1/nf-instances.*\",kubernetes_namespace=\"$namespace\"}[5m]))/ sum(increase(oc_ingressgateway_http_responses_total{Route_path=~\".*nnrf-nfm/v1/nf-instances.*\",kubernetes_namespace=\"$namespace\"}[5m]))*100

Table 6-299 Subscription Service Success Ratio (2xx) %

Field	Details
Description	Measures the total number of successful subscription service operations at NRF. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(increase(oc_ingressgateway_http_responses_total{Status=~\"2.*\",Route_path=~\".*nnrf-nfm/v1/nf-instances.*\",kubernetes_namespace=\"$namespace\"}[5m]))/ sum(increase(oc_ingressgateway_http_responses_total{Route_path=~\".*nnrf-nfm/v1/nf-instances.*\",kubernetes_namespace=\"$namespace\"}[5m]))*100

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-300 Discover Service Success Ratio (2xx) %

Field	Details
Description	Measures the total number of successful discover service operations at NRF. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(increase(oc_ingressgateway_http_responses_total{Status=~\"2.*\",Route_path=~\".*nnrf-disc/v1/nf-instances.*\",kubernetes_namespace=\"$namespace\"}[5m]))/ sum(increase(oc_ingressgateway_http_responses_total{Route_path=~\".*nnrf-disc/v1/nf-instances.*\",kubernetes_namespace=\"$namespace\"}[5m]))*100

Table 6-301 AccessToken Service Success Ratio (2xx) %

Field	Details
Description	Measures the total number of successful Access Token service operations at NRF. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(increase(oc_ingressgateway_http_responses_total{Status=~\"2.*\",Route_path=~\".*oauth2/token.*\",kubernetes_namespace=\"$namespace\"}[5m]))/ sum(increase(oc_ingressgateway_http_responses_total{Route_path=~\".*oauth2/token.*\",kubernetes_namespace=\"$namespace\"}[5m]))*100

Table 6-302 SLF Query Success Ratio (2xx) %

Field	Details
Description	Measures the total number of SLF query response received. For more information about metric used, see NRF SLF Metrics.
Expression	sum(increase(ocnrf_SLF_rx_responses_total{HttpStatusCode=~\"2.*\",kubernetes_namespace=\"$namespace\"}[5m]))/ sum(increase(ocnrf_SLF_rx_responses_total{kubernetes_namespace=\"$namespace\"}[5m]))*100

Table 6-303 Active Registrations

Field	Details
Description	Measures the total number of active registered NFs at any point of time to indicate that the data comes per NF type. This is based on [NfType :- {{ RequesterNfType }}]
Expression	sum(ocnrf_active_registrations_count{namespace="$namespace"}) by(RequesterNfType)

Table 6-304 Suspended NFs

Field	Details
Description	Measures the difference in number of profiles that have been marked as suspended when a profile has missed nfHeartBeatMissAllowed. This is based on [NfInstance :- {{ NfInstanceId }}]
Expression	increase(ocnrf_nf_suspended_total{namespace="$namespace"}[5m])

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-305 Total NfSubscriptions expired

Field	Details
Description	Measures the total number of NfSubscriptions whose validityTime has expired.
Expression	increase(ocnrf_nfSubscriptions_expired_total{namespace="$namespace"}[5m])

Table 6-306 NfDiscoveredProfiles

Field	Details
Description	Measures the rate of number of profiles returned in discovery response. For more information about metric used, see NRF NF Metrics.
Expression	sum(irate(ocnrf_nfDiscover_profiles_discovered_total{namespace="$namespace"}[2m])) by (Bucket,TargetNfType)

Table 6-307 NfRegister Responses

Field	Details
Description	Measures the rate of successful registration responses sent. This is based on [HttpStatusCode:- {{HttpStatusCode}}, NfFqdn-{{ NfFqdn }}] For more information about metric used, see NRF NF Metrics.
Expression	sum(irate(ocnrf_nfRegister_tx_responses_total{namespace="$namespace"}[2m])) by (HttpStatusCode, NfFqdn)

Table 6-308 NfUpdate Response (Complete)

Field	Details
Description	Measures the rate of successful updates response sent. This is based on [HttpStatusCode :- {{HttpStatusCode }}, NfFqdn :- {{ NfFqdn }}] For more information about metric used, see NRF NF Metrics.
Expression	sum(irate(ocnrf_nfUpdate_tx_responses_total{namespace="$namespace",Updatetype="Complete"}[2m])) by (HttpStatusCode, NfFqdn)

Table 6-309 NfUpdate Response (Partial)

Field	Details
Description	Measures the rate of successful updates of an NF Profile which is previously registered at NRF. This is based on [HttpStatusCode :- {{HttpStatusCode }}, NfFqdn :- {{ NfFqdn }}] For more information about metric used, see NRF NF Metrics.
Expression	sum(irate(ocnrf_nfUpdate_tx_responses_total{namespace="$namespace",Updatetype="Partial"}[2m])) by (HttpStatusCode, NfFqdn)

Table 6-310 NfDeregister Responses

Field	Details
Description	Measures the rate of successful Deregistration of NFs at NRF. This is based on [HttpStatusCode :- {{HttpStatusCode}},NfFqdn :- {{ NfFqdn }}] For more information about metric used, see NRF NF Metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-310 (Cont.) NfDeregister Responses

Field	Details
Expression	sum(irate(ocnrf_nfDeregister_tx_responses_total{namespace="$namespace"}[2m])) by (HttpStatusCode, NfFqdn)

Table 6-311 NfProfileRetrieval Response

Field	Details
Description	Measures the rate at which NfProfileRetrieval service requests that NRF rejects due to various error reasons. This KPI considers all non success error codes. This is based on [HttpStatusCode :- {{HttpStatusCode}}] For more information about metric used, see NRF NF Metrics.
Expression	sum(irate(ocnrf_nfProfileRetrieval_tx_responses_total{namespace="$namespace"} [2m])) by(HttpStatusCode)

Table 6-312 NfHeartbeat Responses

Field	Details
Description	Measures the number of Heartbeat Responses sent. This is based on [HttpStatusCode :- {{HttpStatusCode}}] For more information about metric used, see NRF NF Metrics.
Expression	sum(irate(ocnrf_nfHeartBeat_tx_responses_total{namespace="$namespace"}[2m])) by(HttpStatusCode)

Table 6-313 NFStatusSubscribe

Field	Details
Description	Measures the total number of NfStatusSubscribe responses sent. This is based on [HttpStatusCode :- {{HttpStatusCode}}] For more information about metric used, see NRF NF Metrics.
Expression	sum(irate(ocnrf_nfStatusSubscribe_tx_responses_total{namespace="$namespace",OperationType="Create"}[2m])) by (HttpStatusCode)

Table 6-314 NFStatusSubscribe (PATCH)

Field	Details
Description	Measures the total number of NfStatusSubscribe update responses sent. This is based on [HttpStatusCode :- {{HttpStatusCode}}] For more information about metric used, see NRF NF Metrics.
Expression	sum(irate(ocnrf_nfStatusSubscribe_tx_responses_total{namespace="$namespace",OperationType="Update"}[2m])) by (HttpStatusCode)

Table 6-315 NFStatusUnsubscribe [HttpStatusCode :- {{HttpStatusCode}}]

Field	Details
Description	Measures the total number of NfStatusUnsubscribe responses sent. For more information about metric used, see NRF NF Metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-315 (Cont.) NFStatusUnsubscribe [HttpStatusCode :- {{HttpStatusCode}}]

Field	Details
Expression	sum(irate(ocnrf_nfStatusUnsubscribe_tx_responses_total{namespace="$namespace"}[2m])) by (HttpStatusCode)

Table 6-316 NfStatusNotify Response

Field	Details
Description	Measures the total number of NfStatusNotify responses received. This is based on [HttpStatusCode :- {{HttpStatusCode}}] For more information about metric used, see NRF NF Metrics.
Expression	sum(irate(ocnrf_nfStatusNotify_rx_responses_total{namespace="$namespace"}[2m])) by(HttpStatusCode)

Table 6-317 NfDiscovery Responses

Field	Details
Description	Measures the total number of NfDiscover responses sent. This is based on [HttpStatusCode :- {{HttpStatusCode}}] For more information about metric used, see NRF NF Metrics.
Expression	sum(irate(ocnrf_nfDiscover_tx_responses_total{namespace="$namespace"}[2m])) by (HttpStatusCode)

Table 6-318 SLF Response

Field	Details
Description	Measures the total number of SLF query response received. This is based on [HttpStatusCode :- {{HttpStatusCode}},TargetNfType :- {{TargetNfType}}]
Expression	sum(irate(ocnrf_SLF_rx_responses_total{namespace="$namespace"}[2m])) by(HttpStatusCode,TargetNfType)

Table 6-319 Forwarded Response

Field	Details
Description	Measures the total number of forwarded query response received. This is based on [HttpStatusCode :- {{HttpStatusCode}}]
Expression	sum(irate(ocnrf_forward_nfDiscover_rx_responses_total{namespace="$namespace"}[2m])) by(HttpStatusCode)

Table 6-320 Access Token Success

Field	Details
Description	Measures the total number of successful Access Token service operations at NRF. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_ingressgateway_http_responses_total{Status="200 OK",Route_path=~".*/oauth2/token.*",Method="POST",namespace="$namespace"}[5m]))

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-321 4xx Responses (AccessToken)

Field	Details
Description	Measures the rate of successful AccessToken requests with 4xx response at NRF. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_ingressgateway_http_responses_total{Status=~"4.*",Route_path=~".*/oauth2/token*.",namespace="$namespace"}[5m]))

Table 6-322 5xx Responses (AccessToken)

Field	Details
Description	Measures the rate of successful AccessToken requests with 5xx response at NRF. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_ingressgateway_http_responses_total{Status=~"5.*",Route_path=~".*/oauth2/token.*",namespace="$namespace"}[5m]))

Table 6-323 Current KeyId Health

Field	Details
Description	Measures the health status of Oauth Token Signing keys. This is based on [KeyId : {{ KeyId }}, KeyType : {{ KeyType }}] For more information about metric used, see NRF NF Metrics.
Expression	ocnrf_oauth_keyData_healthStatus{app_kubernetes_io_name=\"nrfconfiguration\",isCurrentKeyId=\"true\",kubernetes_namespace=\"$namespace\"}

Table 6-324 NF AccessToken Failures

Field	Details
Description	Measures the rate at which NFAccesstoken service requests that NRF rejects due to various error reasons. This KPI considers all non success error codes. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_ingressgateway_http_responses_total{Status!~"2.*",Route_path=~".*oauth2/token.*",kubernetes_namespace="$namespace"}[5m]))

Table 6-325 NF Discover Profile Discovered

Field	Details
Description	Measures the number of Profiles returned in discovery response. Depending on bucket size and corresponding value will tell how many profiles are returned in discovery response. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(ocnrf_nfDiscover_profiles_discovered_total{namespace="$namespace"}[2m])) by (Bucket,TargetNfType)

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-326 Egress Gateway Response for Nrf Forward Response {{ Status }}

Field	Details
Description	Measures the rate of Nrf forward response received. This is based on {{ Status }} of Nrf Forward Response. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_egressgateway_http_responses_total{namespace="$namespace",Method="GET",NFServiceType="nnrf-disc"}[2m])) by(Status)

Table 6-327 Egress Gateway Response for SLF Response {{ Status }}

Field	Details
Description	Measures the rate of SLF forward response received. This is based on {{ Status }} of SLF response. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_egressgateway_http_responses_total{namespace="$namespace",Method="GET",NFServiceType="nudr-group-id-map"}[2m])) by(Status)

Table 6-328 Ingress Request Rate

Field	Details
Description	Measures the rate of incoming requests at NRF. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_ingressgateway_http_requests_total{container="ingressgateway",namespace="$namespace"}[2m]))

Table 6-329 Response with 2xx status code

Field	Details
Description	Measures the rate of outgoing response with 2xx status code. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_ingressgateway_http_responses_total{Status=~"2.*",container="ingressgateway",namespace="$namespace"}[5m]))

Table 6-330 Response with 4xx status code

Field	Details
Description	Measures the rate of outgoing response with 4xx status code. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_ingressgateway_http_responses_total{Status=~"4.*",container="ingressgateway",namespace="$namespace"}[5m]))

Table 6-331 Response with 5xx status code

Field	Details
Description	Measures the rate of outgoing response with 5xx status code. For more information about metric used, see NRF Gateways Metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-331 (Cont.) Response with 5xx status code

Field	Details
Expression	sum(irate(oc_ingressgateway_http_responses_total{Status=~"5.*",container="ingressgateway",namespace="$namespace"}[5m]))

Table 6-332 NRF Outgoing Response

Field	Details
Description	Measures the rate of outgoing requests at NRF. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_ingressgateway_http_responses_total{app_kubernetes_io_name=\"ingressgateway\",kubernetes_namespace=\"$namespace\"}[5m]))

Table 6-333 Egress Request Rate

Field	Details
Description	Measures the rate of outgoing requests at NRF. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_egressgateway_http_requests_total{namespace="$namespace"}[5m]))

Table 6-334 Response with 2xx status code

Field	Details
Description	Measures the rate of outgoing response with 2xx status code. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_egressgateway_http_responses_total{Status=~"2.*",namespace="$namespace",Direction="egress"}[5m]))

Table 6-335 Response with 4xx status code

Field	Details
Description	Measures the rate of outgoing response with 4xx status code. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_egressgateway_http_responses_total{Status=~"4.*",namespace="$namespace",Direction="egress"}[5m]))

Table 6-336 Response with 5xx status code

Field	Details
Description	Measures the rate of outgoing response with 5xx status code. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_egressgateway_http_responses_total{Status=~"5.*",namespace="$namespace",Direction="egress"}[5m]))

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-337 Egress Response Rate

Field	Details
Description	Measures the rate of outgoing requests at NRF. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_egressgateway_http_responses_total{namespace="$namespace",Direction="egressOut"}[5m]))

Table 6-338 Ingress Request Rate

Field	Details
Description	This metric is pegged for every event occurred when a request is sent to Ingress Gateway. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_ingressgateway_jetty_request_stat_metrics_total{event="onCommit",app_kubernetes_io_name="ingressgateway",namespace="$namespace"}[2m])) by (pod)

Table 6-339 Ingress Response Rate

Field	Details
Description	This metric is pegged for every event occurred when a response is received from Ingress Gateway. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_ingressgateway_jetty_response_stat_metrics_total{event="onComplete",app_kubernetes_io_name="ingressgateway",namespace="$namespace"}[2m])) by (pod)

Table 6-340 Total Incoming Connections at Pod

Field	Details
Description	Measures the total number of active incoming connections per Egress Gateway pod. For more information about metric used, see NRF Gateways Metrics.
Expression	sum by (pod) (oc_egressgateway_incoming_connections{namespace="$namespace"})

Table 6-341 Total Incoming Connections

Field	Details
Description	Measures the total number of active incoming connections from client to Egress Gateway. For more information about metric used, see NRF Gateways Metrics.
Expression	sum (oc_egressgateway_incoming_connections{namespace="$namespace"})

Table 6-342 Total Outgoing Connections at pod

Field	Details
Description	Measures the total number of active outgoing connections per Egress Gateway pod. For more information about metric used, see NRF Gateways Metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-342 (Cont.) Total Outgoing Connections at pod

Field	Details
Expression	sum by (pod) (oc_egressgateway_outgoing_connections{namespace="$namespace"})

Table 6-343 Total Outgoing Connections

Field	Details
Description	Measures the total number of active outgoing connections from Egress Gateway. For more information about metric used, see NRF Gateways Metrics.
Expression	sum (oc_egressgateway_outgoing_connections{namespace="$namespace"})

Table 6-344 Total Incoming Connections at Pod

Field	Details
Description	Measures the total number of active incoming connections per Ingress Gateway pod. For more information about metric used, see NRF Gateways Metrics.
Expression	sum by (pod) (oc_ingressgateway_incoming_connections{namespace="$namespace"})

Table 6-345 Total Incoming Connections

Field	Details
Description	Measures the total number of active incoming connections from client to Ingress Gateway. For more information about metric used, see NRF Gateways Metrics.
Expression	sum (oc_ingressgateway_incoming_connections{namespace="$namespace"})

Table 6-346 Total Outgoing Connections at pod

Field	Details
Description	Measures the total number of active outgoing connections per Ingress Gateway pod. For more information about metric used, see NRF Gateways Metrics.
Expression	sum by (pod) (oc_ingressgateway_outgoing_connections{namespace="$namespace"})

Table 6-347 Total Outgoing Connections

Field	Details
Description	Measures the total number of active outgoing connections from Ingress Gateway. For more information about metric used, see NRF Gateways Metrics.
Expression	sum (oc_ingressgateway_outgoing_connections{namespace="$namespace"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-348 SLF Egress Gateway Latency 99%

Field	Details
Description	Measures the number of SLF request falling in 99 percentile of the latency value at Egress Gateway.
Expression	avg(oc_egressgateway_request_latency_seconds{namespace="$namespace",Method="GET",quantile="0.99"})

Table 6-349 SLF Egress Gateway Latency 95%

Field	Details
Description	Measures the number of SLF request falling in 95 percentile of the latency value at Egress Gateway.
Expression	avg(oc_egressgateway_request_latency_seconds{namespace="$namespace",Method="GET",quantile="0.95"})

Table 6-350 SLF Egress Gateway Latency 90%

Field	Details
Description	Measures the number of SLF request falling in 90 percentile of the latency value at Egress Gateway.
Expression	avg(oc_egressgateway_request_latency_seconds{namespace="$namespace",Method="GET",quantile="0.90"})

Table 6-351 SLF Egress Gateway Latency 50%

Field	Details
Description	Measures the number of SLF request falling in 50 percentile of the latency value at Egress Gateway.
Expression	avg(oc_egressgateway_request_latency_seconds{namespace="$namespace",Method="GET",quantile="0.50"})

Table 6-352 NfAccess Token Latency 99%

Field	Details
Description	Measures the average amount of time taken for processing NfAccess token latency falling in 99 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/oauth2/token",Method="POST",quantile="0.99"})

Table 6-353 NfAccess Token Latency 95%

Field	Details
Description	Measures the average amount of time taken for processing NfAccess token latency falling in 95 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/oauth2/token",Method="POST",quantile="0.95"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-354 NfAccess Token Latency 90%

Field	Details
Description	Measures the average amount of time taken for processing NfAccess token latency falling in 90 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/oauth2/token",Method="POST",quantile="0.90"})

Table 6-355 NfAccess Token Latency 50%

Field	Details
Description	Measures the average amount of time taken for processing NfAccess token latency falling in 50 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/oauth2/token",Method="POST",quantile="0.50"})

Table 6-356 Discovery SLF Latency 99%

Field	Details
Description	Measures the number of SLF discovery request falling in 99 percentile of the latency value.
Expression	avg(ocnrf_slf_jetty_latency_seconds{namespace="$namespace",ServiceOperation="NfDiscover",quantile="0.99"})

Table 6-357 Discovery SLF Latency 90%

Field	Details
Description	Measures the number of SLF discovery request falling in 90 percentile of the latency value.
Expression	avg(ocnrf_slf_jetty_latency_seconds{namespace="$namespace",ServiceOperation="NfDiscover",quantile="0.9"})

Table 6-358 Discovery SLF Latency 50%

Field	Details
Description	Measures the number of SLF discovery request falling in 50 percentile of the latency value.
Expression	avg(ocnrf_slf_jetty_latency_seconds{namespace="$namespace",ServiceOperation="NfDiscover",quantile="0.5"})

Table 6-359 Discovery SLF Latency 95%

Field	Details
Description	Measures the number of SLF discovery request falling in 95 percentile of the latency value.
Expression	avg(ocnrf_slf_jetty_latency_seconds{namespace="$namespace",ServiceOperation="NfDiscover",quantile="0.95"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-360 Discovery Latency 99%

Field	Details
Description	Measures the number of discovery request falling in 99 percentile of the latency value.
Expression	avg(ocnrf_message_processing_time_seconds{namespace="$namespace",ServiceOperation="NfDiscover",quantile="0.99"})

Table 6-361 Discovery Latency 90%

Field	Details
Description	Measures the number of discovery request falling in 90 percentile of the latency value.
Expression	avg(ocnrf_message_processing_time_seconds{namespace="$namespace",ServiceOperation="NfDiscover",quantile="0.9"})

Table 6-362 Discovery Latency 50%

Field	Details
Description	Measures the number of discovery request falling in 50 percentile of the latency value.
Expression	avg(ocnrf_message_processing_time_seconds{namespace="$namespace",ServiceOperation="NfDiscover",quantile="0.5"})

Table 6-363 Discovery Latency 95%

Field	Details
Description	Measures the number of discovery request falling in 95 percentile of the latency value.
Expression	avg(ocnrf_message_processing_time_seconds{namespace="$namespace",ServiceOperation="NfDiscover",quantile="0.95"})

Table 6-364 Ingress Gateway Discovery Latency 99%

Field	Details
Description	Measures the number of discovery request falling in 99 percentile of the latency value at Ingress Gateway.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-disc/v1/nf-instances/**",quantile="0.99"})

Table 6-365 Ingress Gateway Discovery Latency 90%

Field	Details
Description	Measures the number of discovery request falling in 90 percentile of the latency value at Ingress Gateway.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-disc/v1/nf-instances/**",quantile="0.9"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-366 Ingress Gateway Discovery Latency 50%

Field	Details
Description	Measures the number of discovery request falling in 50 percentile of the latency value at Ingress Gateway.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-disc/v1/nf-instances/**",quantile="0.5"})

Table 6-367 Ingress Gateway Discovery Latency 95%

Field	Details
Description	Measures the number of discovery request falling in 95 percentile of the latency value at Ingress Gateway.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-disc/v1/nf-instances/**",quantile="0.95"})

Table 6-368 Ingress Gateway Backend Latency 99%

Field	Details
Description	Measures the number of request falling in 99 percentile of the latency value at Ingress Gateway backend.
Expression	avg(oc_ingressgateway_server_latency_seconds{namespace="$namespace",quantile="0.99",Method="GET"})

Table 6-369 Ingress Gateway Backend Latency 95%

Field	Details
Description	Measures the number of request falling in 95 percentile of the latency value at Ingress Gateway backend.
Expression	avg(oc_ingressgateway_server_latency_seconds{namespace="$namespace",quantile="0.95",Method="GET"})

Table 6-370 Ingress Gateway Backend Latency 90%

Field	Details
Description	Measures the number of request falling in 90 percentile of the latency value at Ingress Gateway backend.
Expression	avg(oc_ingressgateway_server_latency_seconds{namespace="$namespace",quantile="0.9",Method="GET"})

Table 6-371 Discovery Cache Sync

Field	Details
Description	Measures the rate at which in-memory cache entries in the nfdiscovery are synched with Cache Data Service.
Expression	sum(rate(ocnrf_cache_sync_count_total{namespace="$namespace",container="nfdiscovery"}[2m]))

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-372 Ingress Gateway Backend Latency 50%

Field	Details
Description	Measures the number of request falling in 50 percentile of the latency value at Ingress Gateway backend.
Expression	avg(oc_ingressgateway_server_latency_seconds{namespace="$namespace",quantile="0.5",Method="GET"})

Table 6-373 Ingress Gateway Latency 99%

Field	Details
Description	Measures the number of request falling in 99 percentile of the latency value at Ingress Gateway.
Expression	avg(oc_ingressgateway_request_latency_seconds{namespace="$namespace",quantile="0.99",Method="GET"})

Table 6-374 Ingress Gateway Latency 95%

Field	Details
Description	Measures the number of request falling in 95 percentile of the latency value at Ingress Gateway.
Expression	avg(oc_ingressgateway_request_latency_seconds{namespace="$namespace",quantile="0.95",Method="GET"})

Table 6-375 Ingress Gateway Latency 90%

Field	Details
Description	Measures the number of request falling in 90 percentile of the latency value at Ingress Gateway.
Expression	avg(oc_ingressgateway_request_latency_seconds{namespace="$namespace",quantile="0.9",Method="GET"})

Table 6-376 Ingress Gateway Latency 50%

Field	Details
Description	Measures the number of request falling in 50 percentile of the latency value at Ingress Gateway.
Expression	avg(oc_ingressgateway_request_latency_seconds{namespace="$namespace",quantile="0.5",Method="GET"})

Table 6-377 Forward Latency 99%

Field	Details
Description	Measures the number of forward request falling in 99 percentile of the latency value.
Expression	avg(ocnrf_forward_jetty_latency_seconds{namespace="$namespace",ServiceOperation="NfDiscover",quantile="0.99"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-378 Forward Latency 95%

Field	Details
Description	Measures the number of forward request falling in 95 percentile of the latency value.
Expression	avg(ocnrf_forward_jetty_latency_seconds{namespace="$namespace",ServiceOperation="NfDiscover",quantile="0.95"})

Table 6-379 Forward Latency 90%

Field	Details
Description	Measures the number of forward request falling in 90 percentile of the latency value.
Expression	avg(ocnrf_forward_jetty_latency_seconds{namespace="$namespace",ServiceOperation="NfDiscover",quantile="0.9"})

Table 6-380 Forward Latency 50%

Field	Details
Description	Measures the number of forward request falling in 50 percentile of the latency value.
Expression	avg(ocnrf_forward_jetty_latency_seconds{namespace="$namespace",ServiceOperation="NfDiscover",quantile="0.5"})

Table 6-381 NfStatusSubscribe Latency 99%

Field	Details
Description	Measures the number of NF status subscription request falling in 99 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/subscriptions/**",Method="POST",quantile="0.99"})

Table 6-382 NfStatusSubscribe Latency 95%

Field	Details
Description	Measures the number of NF status subscription request falling in 95 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/subscriptions/**",Method="POST",quantile="0.95"})

Table 6-383 NfStatusSubscribe Latency 90%

Field	Details
Description	Measures the number of NF status subscription request falling in 90 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/subscriptions/**",Method="POST",quantile="0.9"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-384 NfStatusSubscribe Latency 50%

Field	Details
Description	Measures the number of NF status subscription request falling in 50 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/subscriptions/**",Method="POST",quantile="0.5"})

Table 6-385 NfStatusSubscribe Latency (PATCH) 99%

Field	Details
Description	Measures the number of NF status subscription (PATCH) request falling in 99 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/subscriptions/**",Method="PATCH",quantile="0.99"})

Table 6-386 NfStatusSubscribe Latency (PATCH) 95%

Field	Details
Description	Measures the number of NF status subscription (PATCH) request falling in 95 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/subscriptions/**",Method="PATCH",quantile="0.95"})

Table 6-387 NfStatusSubscribe Latency (PATCH) 90%

Field	Details
Description	Measures the number of NF status subscription (PATCH) request falling in 90 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/subscriptions/**",Method="PATCH",quantile="0.90"})

Table 6-388 NfStatusSubscribe Latency (PATCH) 50%

Field	Details
Description	Measures the number of NF status subscription (PATCH) request falling in 50 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/subscriptions/**",Method="PATCH",quantile="0.50"})

Table 6-389 NfStatusUnsubscribe Latency 99%

Field	Details
Description	Measures the number of status unsubscription request falling in 99 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/subscriptions/**",Method="DELETE",quantile="0.99"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-390 NfStatusUnsubscribe Latency 95%

Field	Details
Description	Measures the number of status unsubscription request falling in 95 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/subscriptions/**",Method="DELETE",quantile="0.95"})

Table 6-391 NfStatusUnsubscribe Latency 90%

Field	Details
Description	Measures the number of status unsubscription request falling in 90 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/subscriptions/**",Method="DELETE",quantile="0.90"})

Table 6-392 NfStatusUnsubscribe Latency 50%

Field	Details
Description	Measures the number of status unsubscription request falling in 50 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/subscriptions/**",Method="DELETE",quantile="0.50"})

Table 6-393 NfNotification trigger Success

Field	Details
Description	Measures the rate of success notification trigger sent to nfSubscription service from the nfregistration and nrfauditor services.
Expression	sum(rate(ocnrf_nfNotification_trigger_total{errorReason="noError",namespace="$namespace"}[2m]))

Table 6-394 NfNotification trigger Failure

Field	Details
Description	Measures the rate of error notification trigger sent to nfSubscription service from the nfregistration and nrfauditor services.
Expression	sum(rate(ocnrf_nfNotification_trigger_total{errorReason!="noError",namespace="$namespace"}[2m]))

Table 6-395 NfUpdate Latency 99%

Field	Details
Description	Measures the number of update request falling in 99 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/nf-instances/**",Method="PATCH",quantile="0.99"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-396 NfUpdate Latency 95%

Field	Details
Description	Measures the number of update request falling in 95 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/nf-instances/**",Method="PATCH",quantile="0.95"})

Table 6-397 NfUpdate Latency 90%

Field	Details
Description	Measures the number of update request falling in 90 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/nf-instances/**",Method="PATCH",quantile="0.9"})

Table 6-398 NfUpdate Latency 50%

Field	Details
Description	Measures the number of update request falling in 50 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/nf-instances/**",Method="PATCH",quantile="0.5"})

Table 6-399 NfRegister Latency 99%

Field	Details
Description	Measures the number of register request falling in 99 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/nf-instances/**",Method="PUT",quantile="0.99"})

Table 6-400 NfRegister Latency 95%

Field	Details
Description	Measures the number of register request falling in 95 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/nf-instances/**",Method="PUT",quantile="0.95"})

Table 6-401 NfRegister Latency 90%

Field	Details
Description	Measures the number of register request falling in 90 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/nf-instances/**",Method="PUT",quantile="0.9"})

Table 6-402 NfRegister Latency 50%

Field	Details
Description	Measures the number of register request falling in 50 percentile of the latency value.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-402 (Cont.) NfRegister Latency 50%

Field	Details
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/nf-instances/**",Method="PUT",quantile="0.5"})

Table 6-403 NfDeregister Latency 99%

Field	Details
Description	Measures the number of deregister request falling in 99 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/nf-instances/**",Method="DELETE",quantile="0.99"})

Table 6-404 NfDeregister Latency 95%

Field	Details
Description	Measures the number of deregister request falling in 95 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/nf-instances/**",Method="DELETE",quantile="0.95"})

Table 6-405 NfDeregister Latency 90%

Field	Details
Description	Measures the number of deregister request falling in 90 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/nf-instances/**",Method="DELETE",quantile="0.90"})

Table 6-406 NfDeregister Latency 50%

Field	Details
Description	Measures the number of deregister request falling in 50 percentile of the latency value.
Expression	avg(oc_ingressgateway_request_processing_latency_seconds{namespace="$namespace",Route_path="/nnrf-nfm/v1/nf-instances/**",Method="DELETE",quantile="0.50"})

Table 6-407 NfDiscovery Database Latency 99%

Field	Details
Description	Measures the number of discovery request falling in 99 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfDiscover",quantile="0.99"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-408 NfDiscovery Database Latency 95%

Field	Details
Description	Measures the number of discovery request falling in 95 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfDiscover",quantile="0.95"})

Table 6-409 NfDiscovery Database Latency 90%

Field	Details
Description	Measures the number of discovery request falling in 90 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfDiscover",quantile="0.90"})

Table 6-410 NfDiscovery Database Latency 50%

Field	Details
Description	Measures the number of discovery request falling in 50 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfDiscover",quantile="0.50"})

Table 6-411 NfRegistration Database Latency 99%

Field	Details
Description	Measures the number of NfRegistration request falling in 99 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfRegister",quantile="0.99"})

Table 6-412 NfRegistration Database Latency 95%

Field	Details
Description	Measures the number of NfRegistration request falling in 95 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfRegister",quantile="0.95"})

Table 6-413 NfRegistration Database Latency 90%

Field	Details
Description	Measures the number of NfRegistration request falling in 90 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfRegister",quantile="0.90"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-414 NfRegistration Database Latency 50%

Field	Details
Description	Measures the number of NfRegistration request falling in 50 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfRegister",quantile="0.50"})

Table 6-415 NfUpdate Database Latency 99%

Field	Details
Description	Measures the number of NfUpdate request falling in 99 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfUpdate",quantile="0.99"})

Table 6-416 NfUpdate Database Latency 95%

Field	Details
Description	Measures the number of NfUpdate request falling in 95 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfUpdate",quantile="0.95"})

Table 6-417 NfUpdate Database Latency 90%

Field	Details
Description	Measures the number of NfUpdate request falling in 90 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfUpdate",quantile="0.90"})

Table 6-418 NfUpdate Database Latency 50%

Field	Details
Description	Measures the number of NfUpdate request falling in 50 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfUpdate",quantile="0.50"})

Table 6-419 NfHeartBeat Database Latency 99%

Field	Details
Description	Measures the number of NfHeartBeat request falling in 99 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfHeartBeat",quantile="0.99"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-420 NfHeartBeat Database Latency 95%

Field	Details
Description	Measures the number of NfHeartBeat request falling in 95 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfHeartBeat",quantile="0.95"})

Table 6-421 NfHeartBeat Database Latency 90%

Field	Details
Description	Measures the number of NfHeartBeat request falling in 90 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfHeartBeat",quantile="0.9"})

Table 6-422 NfHeartBeat Database Latency 50%

Field	Details
Description	Measures the number of NfHeartBeat request falling in 50 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfHeartBeat",quantile="0.5"})

Table 6-423 NfDeregister Database Latency 99%

Field	Details
Description	Measures the number of NfDeregister request falling in 99 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfDeregister",quantile="0.99"})

Table 6-424 NfDeregister Database Latency 95%

Field	Details
Description	Measures the number of NfDeregister request falling in 95 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfDeregister",quantile="0.95"})

Table 6-425 NfDeregister Database Latency 90%

Field	Details
Description	Measures the number of NfDeregister request falling in 90 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfDeregister",quantile="0.90"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-426 NfDeregister Database Latency 50%

Field	Details
Description	Measures the number of NfDeregister request falling in 50 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfDeregister",quantile="0.50"})

Table 6-427 NfSubscribe Database Latency 99%

Field	Details
Description	Measures the number of NfSubscribe request falling in 99 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfSubscribe",quantile="0.99"})

Table 6-428 NfSubscribe Database Latency 95%

Field	Details
Description	Measures the number of NfSubscribe request falling in 95 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfSubscribe",quantile="0.95"})

Table 6-429 NfSubscribe Database Latency 90%

Field	Details
Description	Measures the number of NfSubscribe request falling in 90 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfSubscribe",quantile="0.90"})

Table 6-430 NfSubscribe Database Latency 50%

Field	Details
Description	Measures the number of NfSubscribe request falling in 50 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfSubscribe",quantile="0.50"})

Table 6-431 NfUnsubscribe Database Latency 99%

Field	Details
Description	Measures the number of NfUnsubscribe request falling in 99 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfSubscriptions",serviceOperation="NfUnsubscribe",quantile="0.99"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-432 NfStatusNotify Database Latency 99%

Field	Details
Description	Measures the number of NfStatusNotify request falling in 99 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfSubscriptions",serviceOperation="NfStatusNotify",quantile="0.99"})

Table 6-433 NfStatusNotify Database Latency 95%

Field	Details
Description	Measures the number of NfStatusNotify request falling in 95 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfSubscriptions",serviceOperation="NfStatusNotify",quantile="0.95"})

Table 6-434 NfStatusNotify Database Latency 90%

Field	Details
Description	Measures the number of NfStatusNotify request falling in 90 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfSubscriptions",serviceOperation="NfStatusNotify",quantile="0.90"})

Table 6-435 NfStatusNotify Database Latency 50%

Field	Details
Description	Measures the number of NfStatusNotify request falling in 50 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfSubscriptions",serviceOperation="NfStatusNotify",quantile="0.50"})

Table 6-436 NfAccessToken Database Latency 99%

Field	Details
Description	Measures the number of NfAccessToken request falling in 99 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfAccessToken",quantile="0.99"})

Table 6-437 NfAccessToken Database Latency 95%

Field	Details
Description	Measures the number of NfAccessToken request falling in 95 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfAccessToken",quantile="0.95"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-438 NfAccessToken Database Latency 90%

Field	Details
Description	Measures the number of NfAccessToken request falling in 90 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfAccessToken",quantile="0.90"})

Table 6-439 NfAccessToken Database Latency 50%

Field	Details
Description	Measures the number of NfAccessToken request falling in 50 percentile of the latency value.
Expression	avg(ocnrf_dbmetrics_round_trip_time_seconds{namespace="$namespace",tableName="NfInstances",serviceOperation="NfAccessToken",quantile="0.50"})

> **Note**
>
> For example, if min is 0 and max is 100, the number of request falling in 95 percentile, which is quantile="0.95", is measured in the Latency 95% KPI.

Table 6-440 NF Profile Size Average

Field	Details
Description	Measures the average size of the NF profiles registered at NRF. For more information about metric used, see NRF NF Metrics.
Expression	avg by (NfType) (ocnrf_nf_profile_size{namespace="$namespace"}))

Table 6-441 NF Profile Size Max

Field	Details
Description	Measures the maximum size of the NF profiles registered at NRF. For more information about metric used, see NRF NF Metrics.
Expression	max by (NfType) (ocnrf_nf_profile_size{namespace="$namespace"})

Table 6-442 Discovery Response Size

Field	Details
Description	Measures the NfDiscovery response size. For more information about metric used, see NRF NF Metrics.
Expression	histogram_quantile(0.99,sum(rate(ocnrf_nfDiscover_tx_response_size_bytes_bucket{namespace="$namespace",TargetNfType="<NF>"}[2m])) by (TargetNfType,le)) Where, <NF>- Name of Network Function. (For example, AUSF, AMF, CHF) For example, histogram_quantile(0.99, sum(rate(ocnrf_nfDiscover_tx_response_size_bytes_bucket{namespace="$namespace",TargetNfType="AMF"}[2m])) by (TargetNfType,le))

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-443 NF Profile Size Limit Breached

Field	Details
Description	Measures the rate of requests rejected due to breach in nfProfile size limit.
Expression	sum(rate(ocnrf_nfProfile_size_limit_breached_total{namespace="$namespace"}[2m])) by (NfInstanceId, ServiceOperation)

6.2.2 Resource Usage KPIs

This section lists the KPIs to measure the CPU, Memory usage, and Kubernetes Horizontal Pod Autoscaling (HPA) counts at NRF. These KPIs are calculated based on Kubernetes metrics.

Table 6-444 nfregistration

Field	Details
Description	Measures the amount of CPU resources consumed by the containers for nfregistration service.
Expression	sum(rate(container_cpu_usage_seconds_total{container="nfregistration",namespace="$namespace"}[2m]))

Table 6-445 nfsubscription

Field	Details
Description	Measures the amount of CPU resources consumed by the containers for nfsubscription service.
Expression	sum(rate(container_cpu_usage_seconds_total{container="nfsubscription",namespace="$namespace"}[2m]))

Table 6-446 nfdiscovery

Field	Details
Description	Measures the amount of CPU resources consumed by the containers for nfdiscovery service.
Expression	sum(rate(container_cpu_usage_seconds_total{container="nfdiscovery",namespace="$namespace"}[2m]))

Table 6-447 nrfauditor

Field	Details
Description	Measures the amount of CPU resources consumed by the containers for nrfauditor service.
Expression	sum(rate(container_cpu_usage_seconds_total{container="nrfauditor",namespace="$namespace"}[2m]))

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-448 nfaccesstoken

Field	Details
Description	Measures the amount of CPU resources consumed by the containers for nfaccesstoken service.
Expression	sum(rate(container_cpu_usage_seconds_total{container="nfaccesstoken",namespace="$namespace"}[2m]))

Table 6-449 nrfconfiguration

Field	Details
Description	Measures the amount of CPU resources consumed by the containers for nrfconfiguration service.
Expression	sum(rate(container_cpu_usage_seconds_total{container="nrfconfiguration",namespace="$namespace"}[2m]))

Table 6-450 ingress-gateway

Field	Details
Description	Measures the amount of CPU resources consumed by the containers for Ingress Gateway service.
Expression	sum(rate(container_cpu_usage_seconds_total{container="ingressgateway",namespace="$namespace"}[2m]))

Table 6-451 egress-gateway

Field	Details
Description	Measures the amount of CPU resources consumed by the containers for Egress Gateway service.
Expression	sum(rate(container_cpu_usage_seconds_total{container="egressgateway",namespace="$namespace"}[2m]))

Table 6-452 nfregistration

Field	Details
Description	Measures the amount of memory consumed by the containers for nfregistration service.
Expression	sum (container_memory_usage_bytes{container="nfregistration",namespace="$namespace"})

Table 6-453 nfsubscription

Field	Details
Description	Measures the amount of memory consumed by the containers for nfsubscription service.
Expression	sum (container_memory_usage_bytes{container="nfsubscription",namespace="$namespace"})

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-454 nfdiscovery

Field	Details
Description	Measures the amount of memory consumed by the containers for nfdiscovery service.
Expression	sum (container_memory_usage_bytes{container="nfdiscovery",namespace="$namespace"})

Table 6-455 nrfauditor

Field	Details
Description	Measures the amount of memory consumed by the containers for nrfauditor service.
Expression	sum (container_memory_usage_bytes{container="nrfauditor",namespace="$namespace"})

Table 6-456 nfaccesstoken

Field	Details
Description	Measures the amount of memory consumed by the containers for nfaccesstoken service.
Expression	sum (container_memory_usage_bytes{container="nfaccesstoken",namespace=$namespace"})

Table 6-457 nrfconfiguration

Field	Details
Description	Measures the amount of memory consumed by the containers for nrfconfiguration service.
Expression	sum (container_memory_usage_bytes{container="nrfconfiguration",namespace="$namespace"})

Table 6-458 ingress-gateway

Field	Details
Description	Measures the amount of memory consumed by the containers for Ingress Gateway service.
Expression	sum (container_memory_usage_bytes{container="ingressgateway",namespace="$namespace"})

Table 6-459 egress-gateway

Field	Details
Description	Measures the amount of memory consumed by the containers for Egress Gateway service.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-459 (Cont.) egress-gateway

Field	Details
Expression	sum (container_memory_usage_bytes{container="egressgateway",namespace="$namespace"})

Table 6-460 ocnrf-ingressgateway

Field	Details
Description	Measures the number of Ingress Gateway replicas currently available and managed by the autoscaler.
Expression	kube_horizontalpodautoscaler_status_current_replicas{namespace="$namespace",horizontalpodautoscaler=~".*-ingressgateway.*"}

Table 6-461 ocnrf-nfregistration

Field	Details
Description	Measures the number of nfregistration replicas currently available and managed by the autoscaler.
Expression	kube_horizontalpodautoscaler_status_current_replicas{namespace="$namespace",horizontalpodautoscaler=~".*-nfregistration.*"}

Table 6-462 ocnrf-nfsubscription

Field	Details
Description	Measures the number of nfsubscription replicas currently available and managed by the autoscaler.
Expression	kube_horizontalpodautoscaler_status_current_replicas{namespace="$namespace",horizontalpodautoscaler=~".*-nfsubscription.*"}

Table 6-463 ocnrf-nfdiscovery

Field	Details
Description	Measures the number of nfdiscovery replicas currently available and managed by the autoscaler.
Expression	kube_horizontalpodautoscaler_status_current_replicas{namespace="$namespace",horizontalpodautoscaler=~".*-nfdiscovery.*"}

Table 6-464 ocnrf-nfaccesstoken

Field	Details
Description	Measures the number of nfaccesstoken replicas currently available and managed by the autoscaler.
Expression	kube_horizontalpodautoscaler_status_current_replicas{namespace="$namespace",horizontalpodautoscaler=~".*-nfaccesstoken.*"}

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-465 ocnrf-egressgateway

Field	Details
Description	Measures the number of egressgateway replicas currently available and managed by the autoscaler.
Expression	kube_horizontalpodautoscaler_status_current_replicas{namespace="$namespace",horizontalpodautoscaler=~".*-egressgateway.*"}

6.2.3 Feature Specific KPIs

This section lists the KPIs related to Global Subscription Limit, Pod Protection, NF FQDN validation, Geo redundancy features at NRF. These KPIs are calculated based on Kubernetes metrics.

6.2.3.1 Global Subscription Limit Feature

6.2.3.1.1 Global Subscription Limit Level

Table 6-466 Active Subscriptions

Field	Details
Description	Measures the total number of active subscriptions.
Expression	ocnrf_nfset_active_subscriptions{kubernetes_namespace=\"$namespace\"}

Table 6-467 Subscription Limit level

Field	Details
Description	Measures the subscription limit.
Expression	ocnrf_nfset_limit_level{kubernetes_namespace=\"$namespace\"}

6.2.3.1.2 Subscription Limit Breached

Table 6-468 Total NfStatusSubscribe(Create) Rejected

Field	Details
Description	Measures the total number of create subscription requests that are rejected.
Expression	sum(rate(ocnrf_nfStatusSubscribe_tx_responses_total{namespace="$namespace",RejectionReason="SubscriptionLimitExceeded",OperationType="Create"}[2m]))

Table 6-469 Total NfStatusSubscribe(Update) Rejected

Field	Details
Description	Measures the total number of update subscription requests that are rejected.
Expression	sum(rate(ocnrf_nfStatusSubscribe_tx_responses_total{namespace="$namespace",RejectionReason="SubscriptionLimitExceeded",OperationType="Update"}[2m]))

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.2.3.2 NF FQDN Validation KPIs

This section lists the network function fully qualified domain name (FQDN) validation KPIs at NRF.

Table 6-470 NfRegister - FQDN Validation Failed

Field	Details
Description	Measures the total number of registration request for which FQDN based authentication failed at NRF. This is based on [NfFqdn :- {{ NfFqdn}}]. For more information about metric used, see NF Authentication Metrics.
Expression	increase(ocnrf_nf_authentication_failure_total{namespace=\"$namespace\",ServiceOperation=\"NfRegister\"}[5m])

Table 6-471 NfSubscribe - FQDN Validation Failed

Field	Details
Description	Measures the total number of subscription request for which FQDN based authentication failed at NRF. This is based on [NfFqdn :- {{ NfFqdn}}]. For more information about metric used, see NF Authentication Metrics.
Expression	increase(ocnrf_nf_authentication_failure_total{namespace=\"$namespace\",ServiceOperation=\"NfStatusSubscribe\"}[5m])

Table 6-472 NFDiscover - FQDN Validation Failed

Field	Details
Description	Measures the total number of discovery request for which FQDN based authentication failed at NRF. This is based on [NfFqdn :- {{ NfFqdn}}]. For more information about metric used, see NF Authentication Metrics.
Expression	increase(ocnrf_nf_authentication_failure_total{namespace=\"$namespace\",ServiceOperation=\"NFDiscover\"}[5m])

Table 6-473 NFAccessToken - FQDN Validation Failed

Field	Details
Description	Measures the total number of access token request for which FQDN based authentication failed at NRF. This is based on [NfFqdn :- {{ NfFqdn}}]. For more information about metric used, see NF Authentication Metrics.
Expression	increase(ocnrf_nf_authentication_failure_total{namespace=\"$namespace\",ServiceOperation=\"NFAccessToken\"}[5m])

Table 6-474 NFProfileRetrieval - FQDN Validation Failed

Field	Details
Description	Measures the total number of profile retrieval request for which FQDN based authentication failed at NRF. This is based on [NfFqdn :- {{ NfFqdn}}]. For more information about metric used, see NF Authentication Metrics.
Expression	increase(ocnrf_nf_authentication_failure_total{namespace=\"$namespace\",ServiceOperation=\"NFProfileRetrieval\"}[5m])

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-475 NFListRetrieval - FQDN Validation Failed

Field	Details
Description	Measures the total number of NF list retrieval request for which FQDN based authentication failed at NRF. This is based on [NfFqdn :- {{ NfFqdn}}]. For more information about metric used, see NF Authentication Metrics.
Expression	increase(ocnrf_nf_authentication_failure_total{namespace=\"$namespace\",ServiceOperation=\"NFListRetrieval\"}[5m])

6.2.3.3 Georedundancy KPIs

This section lists the georedundancy KPIs at NRF.

Table 6-476 Total NFs switch over

Field	Details
Description	Measures the number of NFInstances that got switched over from the mated site. This is based on [ServiceOperation :- {{ Service_Operation }} ,RemoteNrfInstanceId :- {{ RemoteNrfInstanceId }}]. For more information about metric used, see Georedundancy Metrics.
Expression	increase(ocnrf_nf_switch_over_total{kubernetes_namespace=\"$namespace\"}[5m])

Table 6-477 Total Subscriptions switch over

Field	Details
Description	Measures the number of NfSubscriptions that got switched over from the mated site. This is based on [ServiceOperation :- {{ ServiceOperation }} RemoteNrfInstanceId :- {{ RemoteNrfInstanceId }}]. For more information about metric used, see Georedundancy Metrics.
Expression	increase(ocnrf_nfSubscriptions_switch_over_total{kubernetes_namespace=\"$namespace\"}[5m])

Table 6-478 Total Stale NFs deleted

Field	Details
Description	Measures the number of NfInstances that get deleted by the NrfAuditor when it detects a record to be stale. For more information about metric used, see Georedundancy Metrics.
Expression	increase(ocnrf_stale_nf_deleted_total{namespace="$namespace"}[5m])

Table 6-479 Total Stale Subscriptions deleted

Field	Details
Description	Measures the number of NfSubscriptions that get deleted by the NrfAuditor when it detects a record to be stale. For more information about metric used, see Georedundancy Metrics.
Expression	increase(ocnrf_stale_nfSubscriptions_deleted_total{namespace="$namespace"}[5m])

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.2.3.4 Subscription Pod Protection

Table 6-480 Subscription Pod Overall Congestion State

Field	Details
Description	Tracks the overall congestion state per nfSubscription pod.
Expression	sum(ocnrf_pod_congestion_state{namespace='$namespace'}) by (pod)

Table 6-481 Subscription Pod CPU Congestion state

Field	Details
Description	Tracks the congestion state based on the CPU usage of the subscription pod.
Expression	sum(ocnrf_pod_cpu_congestion_state{namespace='$namespace'}) by (pod)

Table 6-482 Subscription Pod CPU Usage

Field	Details
Description	Measures the CPU usage of the pod.
Expression	sum(ocnrf_pod_cpu_usage{namespace="$namespace",container="nfsubscription"}) by (pod)

Table 6-483 Subscription Pod Pending Message Count Congestion state

Field	Details
Description	Tracks the congestion state based on the pending message count of the subscription pod.
Expression	sum(ocnrf_pod_pending_message_count_congestion_state{namespace='$namespace'}) by (pod)

Table 6-484 Subscription Pod Pending Message Count

Field	Details
Description	Measures the Pending Message Count value of the pod.
Expression	sum(ocnrf_pod_pending_message_count{namespace="$namespace",container="nfsubscription"}) by (pod)

Table 6-485 Subscription Pod Incoming Connections Rejected

Field	Details
Description	Measures the number of connections that have been rejected in the DoC or Congested state. This is based on {{pod }}.
Expression	sum(rate(ocnrf_pod_incoming_connection_rejected_total{namespace="$namespace"}[2m])) by (pod)

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-486 Incoming Connections

Field	Details
Description	Tracks the number of incoming connections. This is based on {{pod }}.
Expression	sum(irate(ocnrf_incoming_connections{namespace="$namespace"}[2m])) by (pod)

Table 6-487 Max Streams

Field	Details
Description	Tracks the maximum concurrent streams per connection. This is based on {{ pod }} : {{ connectionId }}.
Expression	sum(ocnrf_max_concurrent_streams{namespace="$namespace"}) by(pod,connectionId)

6.2.3.5 Controlled Shutdown of NRF

This section lists the controlled shutdown KPIs at NRF.

Table 6-488 Operational State {{ pod }}

Field	Details
Description	Measures the Operational State of NRF. The metric is pegged in the NrfAuditor microservice. If any one of the NrfAuditor pod shows the operational state as COMPLETE_SHUTDOWN, then the state shall be considered as Complete Shutdown. For more information about metric used, see NRF NF Metrics.
Expression	sum(ocnrf_operational_state{namespace="$namespace"}) by (pod)

Table 6-489 NRF Audit status

Field	Details
Description	Measures the audit status of NRF. The metric is pegged in the NrfAuditor microservice. If any one of the NrfAuditor pod shows the audit state as paused then the audit status is considered as paused. For more information about metric used, see NRF NF Metrics.
Expression	sum(ocnrf_audit_status{namespace="$namespace"})

6.2.3.6 Network Slice Specific KPIs

This section lists the network slice specific KPIs at NRF.

Table 6-490 Request [sNssai: {{ sNssai }}]

Field	Details
Description	Measures the total number of NfRegister requests received at a slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-490 (Cont.) Request [sNssai: {{ sNssai }}]

Field	Details
Expression	sum(irate(ocnrf_nfRegister_requests_perSnssai_total{namespace="$namespace"}[2m])) by(sNssai)

Table 6-491 Success [sNssai:{{ sNssai }}]

Field	Details
Description	Measures the total number of NfRegister successful response sent per slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfRegister_success_responses_perSnssai_total{namespace="$namespace"}[5m])) by (sNssai)

Table 6-492 Failures [sNssai: {{ sNssai }}]

Field	Details
Description	Measures the total number of NfRegister failure response sent per slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfRegister_failure_responses_perSnssai_total{namespace="$namespace"}[2m])) by (sNssai)

Table 6-493 Request [sNssai: {{ sNssai }}]

Field	Details
Description	Measures the total number of NfDeregister requests received at a slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfDeregister_requests_perSnssai_total{namespace="$namespace"}[2m])) by(sNssai)

Table 6-494 Success [sNssai:{{ sNssai }}]

Field	Details
Description	Measures the total number of NfDeregister successful response sent per slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfDeregister_success_responses_perSnssai_total{namespace="$namespace"}[5m])) by (sNssai)

Table 6-495 Failures [sNssai: {{ sNssai }}]

Field	Details
Description	Measures the total number of NfDeregister failure response sent per slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-495 (Cont.) Failures [sNssai: {{ sNssai }}]

Field	Details
Expression	sum(irate(ocnrf_nfDeregister_failure_responses_perSnssai_total{namespace="$namespace"}[2m])) by (sNssai)

Table 6-496 Success [Nsi :{{ Nsi }}]

Field	Details
Description	Measures the total number of NfDeregister successful response sent per slice that is identified by specific NSI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfDeregister_success_responses_perNsi_total{namespace="$namespace"}[5m])) by (Nsi)

Table 6-497 Failures [Nsi :{{ Nsi }}]

Field	Details
Description	Measures the total number of NfDeregister failure response sent per slice that is identified by specific NSI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfDeregister_failure_responses_perNsi_total{namespace="$namespace"}[2m])) by (Nsi)

Table 6-498 Requests [sNssai: {{ sNssai }}]

Field	Details
Description	Measures the total number of NfUpdate requests received at a slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfUpdate_requests_perSnssai_total{namespace="$namespace"}[2m])) by(sNssai)

Table 6-499 Success [sNssai:{{ sNssai }}]

Field	Details
Description	Measures the total number of NfUpdate successful response sent per slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfUpdate_success_responses_perSnssai_total{namespace="$namespace"}[5m])) by (sNssai)

Table 6-500 Failures [sNssai: {{ sNssai }}]

Field	Details
Description	Measures the total number of NfUpdate failure response sent per slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-500 (Cont.) Failures [sNssai: {{ sNssai }}]

Field	Details
Expression	sum(irate(ocnrf_nfUpdate_failure_responses_perSnssai_total{namespace="$namespace"}[2m])) by (sNssai)

Table 6-501 Requests [Nsi: {{ Nsi }}]

Field	Details
Description	Measures the total number of NfUpdate requests received at a slice that is identified by specific NSI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfUpdate_requests_perNsi_total{namespace="$namespace"}[2m])) by(Nsi)

Table 6-502 Success [Nsi :{{ Nsi }}]

Field	Details
Description	Measures the total number of NfUpdate successful response sent per slice that is identified by specific NSI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfUpdate_success_responses_perNsi_total{namespace="$namespace"}[5m])) by (Nsi)

Table 6-503 Failures [Nsi :{{ Nsi }}]

Field	Details
Description	Measures the total number of NfUpdate failure response sent per slice that is identified by specific NSI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfUpdate_failure_responses_perNsi_total{namespace="$namespace"}[2m])) by (Nsi)

Table 6-504 Requests [sNssai: {{ sNssai }}]

Field	Details
Description	Measures the total number of NF HeartBeat requests received at a slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfHeartBeat_requests_perSnssai_total{namespace="$namespace"}[2m])) by(sNssai)

Table 6-505 Success [sNssai:{{ sNssai }}]

Field	Details
Description	Measures the total number of NF HeartBeat successful response sent per slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-505 (Cont.) Success [sNssai:{{ sNssai }}]

Field	Details
Expression	sum(irate(ocnrf_nfHeartBeat_success_responses_perSnssai_total{namespace="$namespace"}[5m])) by (sNssai)

Table 6-506 Failures [sNssai: {{ sNssai }}]

Field	Details
Description	Measures the total number of NF HeartBeat failure response sent per slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfHeartBeat_failure_responses_perSnssai_total{namespace="$namespace"}[2m])) by (sNssai)

Table 6-507 Requests [Nsi: {{ Nsi }}]

Field	Details
Description	Measures the total number of NF HeartBeat requests received at a slice that is identified by specific NSI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfHeartBeat_requests_perNsi_total{namespace="$namespace"}[2m])) by(Nsi)

Table 6-508 Success [Nsi :{{ Nsi }}]

Field	Details
Description	Measures the total number of NF HeartBeat successful response sent per slice that is identified by specific NSI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfHeartBeat_success_responses_perNsi_total{namespace="$namespace"}[5m])) by (Nsi)

Table 6-509 Failures [Nsi :{{ Nsi }}]

Field	Details
Description	Measures the total number of NF HeartBeat failure response sent per slice that is identified by specific NSI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfHeartBeat_failure_responses_perNsi_total{namespace="$namespace"}[2m])) by (Nsi)

Table 6-510 Requests [sNssai: {{ sNssai }}]

Field	Details
Description	Measures the total number of NfDiscovery requests received at a slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-510 (Cont.) Requests [sNssai: {{ sNssai }}]

Field	Details
Expression	sum(irate(ocnrf_nfDiscover_rx_requests_perSnssai_total{namespace="$namespace"}[2m])) by(sNssai)

Table 6-511 Success [sNssai:{{ sNssai }}]

Field	Details
Description	Measures the total number of NfDiscovery successful response sent per slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfDiscover_tx_success_response_perSnssai_total{namespace="$namespace"}[5m])) by (sNssai)

Table 6-512 Failures [sNssai: {{ sNssai }}]

Field	Details
Description	Measures the total number of NfDiscovery failure response sent per slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfDiscover_tx_failure_response_perSnssai_total{namespace="$namespace"}[2m])) by (sNssai)

Table 6-513 Empty [sNssai: {{ sNssai }}]

Field	Details
Description	Measures the total number of NfDiscovery empty response sent per slice that is identified by specific SNSSAI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfDiscover_tx_empty_response_perSnssai_total{namespace="$namespace"}[2m])) by (sNssai)

Table 6-514 Requests [Nsi: {{ Nsi }}]

Field	Details
Description	Measures the total number of NfDiscovery requests received at a slice that is identified by specific NSI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfDiscover_rx_requests_perNsi_total{namespace="$namespace"}[2m])) by(Nsi)

Table 6-515 Success [Nsi :{{ Nsi }}]

Field	Details
Description	Measures the total number of nfDiscovery successful response sent per slice that is identified by specific NSI. For more information about metric used, see Network Slice Specific Metrics.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-515 (Cont.) Success [Nsi :{{ Nsi }}]

Field	Details
Expression	sum(irate(ocnrf_nfDiscover_tx_success_response_perNsi_total{namespace="$namespace"}[5m])) by (Nsi)

Table 6-516 Failures [Nsi :{{ Nsi }}]

Field	Details
Description	Measures the total number of NfDiscovery failure response sent per slice that is identified by specific NSI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfDiscover_tx_failure_response_perNsi_total{namespace="$namespace"}[2m])) by (Nsi)

Table 6-517 Empty [Nsi: {{ Nsi }}]

Field	Details
Description	Measures the total number of NfDiscovery empty response sent per slice that is identified by specific NSI. For more information about metric used, see Network Slice Specific Metrics.
Expression	sum(irate(ocnrf_nfDiscover_tx_empty_response_perNsi_total{namespace="$namespace"}[2m])) by (Nsi)

6.2.3.7 SCP Health Status

This section lists the controlled shutdown KPIs at NRF.

Table 6-518 Available Peers {{peerset}}

Field	Details
Description	Measures the total number of available peers in a peerset. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(oc_egressgateway_peer_available_count{namespace="$namespace"}) by (peerset)

Table 6-519 Ping Request

Field	Details
Description	Measures the rate of Ping Requests to the peers. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_egressgateway_peer_health_ping_request_total{namespace="$namespace"}[2m]))

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-520 Ping Response {{ status }}

Field	Details
Description	Measures the rate of Ping Responses to the peers. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(irate(oc_egressgateway_peer_health_ping_response_total{namespace="$namespace"}[2m])) by (status)

Table 6-521 SCP health status {{ peer }}

Field	Details
Description	Measures the health status of the peer. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(oc_egressgateway_peer_health_status{namespace="$namespace"}) by (peer)

Table 6-522 Total Peers {{peerset}}

Field	Details
Description	Measures the total number of peers in a peerset. For more information about metric used, see NRF Gateways Metrics.
Expression	sum(oc_egressgateway_peer_count{namespace="$namespace"}) by (peerset)

6.2.3.8 Ingress Gateway Pod Protection

Table 6-523 OCNRF Ingress-Gateway Pod Overall Congestion State

Field	Details
Description	Indicates the current state of the Ingress Gateway pod.
Expression	sum(oc_ingressgateway_pod_congestion_state{namespace='$namespace'}) by (pod)

Table 6-524 OCNRF Ingress-Gateway Pod CPU Congestion State

Field	Details
Description	Tracks the CPU utilization based on which Ingress Gateway pod is calculating its congestion state.
Expression	sum(oc_ingressgateway_pod_resource_state{type="CPU",namespace='$namespace'}) by (pod)

Table 6-525 OCNRF Ingress-Gateway Pod CPU Usage

Field	Details
Description	Tracks the CPU utilization based on which Ingress Gateway pod is calculating its congestion state.
Expression	sum(oc_ingressgateway_pod_resource_stress{type="CPU",namespace='$namespace'}) by (pod)

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-526 OCNRF Ingress-Gateway Incoming Pod Connections Rejected

Field	Details
Description	Tracks the number of connections dropped at Congested or DOC state.
Expression	sum(irate(oc_ingressgateway_incoming_pod_connections_rejected_total{namespace="$namespace"}[5m])) by (pod)

Table 6-527 OCNRF Ingress-Gateway Pod Pending Message Congestion State

Field	Details
Description	Tracks the congestion state based on the pending message count of the Ingress Gateway pod.
Expression	sum(oc_ingressgateway_pod_resource_state{type="PENDING_REQUEST",namespace='$namespace'}) by (pod)

Table 6-528 OCNRF Ingress Gateway Pod Pending Messages

Field	Details
Description	Measures the pending message count usage of the Ingress Gateway pod.
Expression	sum(oc_ingressgateway_pod_resource_stress{type="PENDING_REQUEST",namespace='$namespace'}) by (pod)

6.2.3.9 NRF Growth Specific KPIs

Table 6-529 Total Number of CDS Requests

Field	Details
Description	Measures the rate of incoming requests at Cache Data Service.
Expression	`sum(rate(ocnrf_cds_rx_requests_total{namespace="$namespace"}[2m]))`

Table 6-530 Total Number of CDS Responses

Field	Details
Description	Measures the rate of responses from Cache Data Service.
Expression	`sum(rate(ocnrf_cds_tx_responses_total{namespace="$namespace"}[2m])) by (HttpStatusCode)`

Table 6-531 Total Number of CDS Requests per Service Operation

Field	Details
Description	Measures the rate of outgoing requests by local NRF microservices to Cache Data Service for state data.
Expression	`sum(rate(ocnrf_query_cds_requests_total{namespace="$namespace"}[2m])) by (ServiceOperation)`

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-532 Total Number of CDS Responses per Service Operation

Field	Details
Description	Measures the rate of response received by local NRF microservices from Cache Data Service for state data.
Expression	`sum(rate(ocnrf_query_cds_responses_total{namespace="$namespace"}[2m])) by (ServiceOperation,HttpStatusCode)`

Table 6-533 CDS Latency 50%

Field	Details
Description	Measures the average amount of time taken to process the requests by Cache Data Service at the 50 percentile of the latency value.
Expression	`avg(ocnrf_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.5"})`

Table 6-534 CDS Latency 90%

Field	Details
Description	Measures the average amount of time taken to process the requests by Cache Data Service at the 90 percentile of the latency value.
Expression	`avg(ocnrf_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.9"})`

Table 6-535 CDS Latency 95%

Field	Details
Description	Measures the average amount of time taken to process the requests by Cache Data Service at the 95 percentile of the latency value.
Expression	`avg(ocnrf_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.95"})`

Table 6-536 CDS Latency 99%

Field	Details
Description	Measures the average amount of time taken to process the requests by Cache Data Service at the 99 percentile of the latency value.
Expression	`avg(ocnrf_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.99"})`

Table 6-537 CDS Request Latency 99% for NfAccessToken Service Operation

Field	Details
Description	Measures the average amount of time taken for querying state data from Cache Data Service for NfAccessToken service operation at the 99 percentile of the latency value.
Expression	`avg(ocnrf_query_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.99",ServiceOperation="NfAccessToken"})`

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-538 CDS Request Latency 99% for NfDiscover Service Operation

Field	Details
Description	Measures the average amount of time taken for querying state data from Cache Data Service for NfDiscover service operation at the 99 percentile of the latency value.
Expression	`avg(ocnrf_query_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.99",ServiceOperation="NfDiscover"})`

Table 6-539 CDS Request Latency 99% for NfListRetrieval Service Operation

Field	Details
Description	Measures the average amount of time taken for querying state data from Cache Data Service for NfListRetrieval service operation at the 99 percentile of the latency value.
Expression	`avg(ocnrf_query_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.99",ServiceOperation="NfListRetrieval"})`

Table 6-540 CDS Request Latency 99% for NfProfileRetrieval Service Operation

Field	Details
Description	Measures the average amount of time taken for querying state data from Cache Data Service for NfProfileRetrieval service operation at the 99 percentile of the latency value.
Expression	`avg(ocnrf_query_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.99",ServiceOperation="NfProfileRetrieval"})`

Table 6-541 CDS Request Latency 99% for NfSubscribe Service Operation

Field	Details
Description	Measures the average amount of time taken for querying state data from Cache Data Service for NfSubscribe service operation at the 99 percentile of the latency value.
Expression	`avg(ocnrf_query_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.99",ServiceOperation="NfSubscribe"})`

Table 6-542 CDS Request Latency 99% for NfUnsubscribe Service Operation

Field	Details
Description	Measures the average amount of time taken for querying state data from Cache Data Service for NfUnsubscribe service operation at the 99 percentile of the latency value.
Expression	`avg(ocnrf_query_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.99",ServiceOperation="NfUnsubscribe"})`

Table 6-543 CDS Request Latency 99% for notifyProfileDeregistration Service Operation

Field	Details
Description	Measures the average amount of time taken for querying state data from Cache Data Service for notifyProfileDeregistration service operation at the 99 percentile of the latency value.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-543 (Cont.) CDS Request Latency 99% for notifyProfileDeregistration Service Operation

Field	Details
Expression	`avg(ocnrf_query_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.99",ServiceOperation="notifyProfileDeregistration"})`

Table 6-544 CDS Request Latency 99% for notifyProfileRegistration Service Operation

Field	Details
Description	Measures the average amount of time taken for querying state data from Cache Data Service for notifyProfileRegistration service operation at the 99 percentile of the latency value.
Expression	`avg(ocnrf_query_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.99",ServiceOperation="notifyProfileRegistration"})`

Table 6-545 CDS Request Latency 99% for notifyProfileChange Service Operation

Field	Details
Description	Measures the average amount of time taken for querying state data from Cache Data Service for notifyProfileChange service operation at the 99 percentile of the latency value.
Expression	`avg(ocnrf_query_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.99",ServiceOperation="notifyProfileChange"})`

Table 6-546 CDS Request Latency 99% for nrfauditor Service Operation

Field	Details
Description	Measures the average amount of time taken for querying state data from Cache Data Service for nrfauditor service operation at the 99 percentile of the latency value.
Expression	`avg(ocnrf_query_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.99",ServiceOperation="nrfauditor"})`

Table 6-547 Total Number of CDS Requests per Request Type

Field	Details
Description	Measures the rate of incoming requests at Cache Data Service per request type.
Expression	`sum(rate(ocnrf_cds_rx_requests_total{namespace="$namespace"}[2m])) by (RequestType)`

Table 6-548 Total Number of CDS Responses per Request Type

Field	Details
Description	Measures the rate of responses from Cache Data Service per request type.
Expression	`sum(rate(ocnrf_cds_tx_responses_total{namespace="$namespace"}[2m])) by (RequestType, HttpStatusCode)`

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-549 Total Number of Remote CDS Requests

Field	Details
Description	Measures the rate of requests sent to the NRF of a remote NRF set by Cache Data Service.
Expression	`sum(rate(ocnrf_query_remote_cds_requests_total{namespace="$namespace"}[2m])) by (TargetNrfSetId)`

Table 6-550 Total Number of Remote CDS Responses

Field	Details
Description	Indicates the total number of responses received from the NRF in the remote set.
Expression	`sum(rate(ocnrf_query_remote_cds_responses_total{namespace="$namespace"}[2m])) by (TargetNrfSetId, HttpStatusCode)`

Table 6-551 Remote CDS Query Latency 50%

Field	Details
Description	Measures the average amount of time taken to process the requests sent by Cache Data Service towards remote NRF set and response received at the 50 percentile of the latency value.
Expression	`avg(ocnrf_query_remote_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.5"})`

Table 6-552 Remote CDS Query Latency 90%

Field	Details
Description	Measures the average amount of time taken to process the requests sent by Cache Data Service towards remote NRF set and response received at the 90 percentile of the latency value.
Expression	`avg(ocnrf_query_remote_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.9"})`

Table 6-553 Remote CDS Query Latency 95%

Field	Details
Description	Measures the average amount of time taken to process the requests sent by Cache Data Service towards remote NRF set and response received at the 95 percentile of the latency value.
Expression	`avg(ocnrf_query_remote_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.95"})`

Table 6-554 Remote CDS Query Latency 99%

Field	Details
Description	Measures the average amount of time taken to process the requests sent by Cache Data Service towards remote NRF set and response received at the 99 percentile of the latency value.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-554 (Cont.) Remote CDS Query Latency 99%

Field	Details
Expression	`avg(ocnrf_query_remote_cds_round_trip_time_seconds{namespace="$namespace",quantile="0.99"})`

Table 6-555 Database Fallback

Field	Details
Description	Measures the rate at which the service request was fulfilled from the database by the NRF microservices as the query towards the Cache Data Service failed.
Expression	`sum(rate(ocnrf_db_fallback_total{namespace="$namespace"}[2m])) by (ServiceOperation)`

Table 6-556 CDS Cache Sync

Field	Details
Description	Measures the rate at which state data entries are synched from the local database at CDS.
Expression	`sum(rate(ocnrf_cache_sync_count_total{namespace="$namespace",container="nrfcachedata"}[2m]))`

6.2.3.10 Egress Gateway Pod Throttling

Table 6-557 Egress Gateway Discards Due To Rate Limit

Field	Details
Description	Measures the requests discarded by Egress Gateway due to Egress Gateway Pod Throttling feature. For more information about metric used, see Egress Gateway Metrics.
Expression	sum(rate(oc_egressgateway_podlevel_throttling_discarded_total{namespace="$namespace"}[2m])) by (pod)

6.2.3.11 Ingress Gateway Pod Protection Using Rate Limiting

Table 6-558 Allowed Request Rate Per Route Id

Field	Details
Description	Measures the request rate allowed per Route Id. For more information about metric used, see Ingress Gateway Metrics.
Expression	sum(irate(oc_ingressgateway_http_request_ratelimit_values_total{Allowed="true",namespace="$namespace"}[2m])) by (Id)

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-559 Total Rejections Chain Length

Field	Details
Description	Measures the number of consecutive rejects before a successful accept. The metric indicates rejections due to traffic burst. For more information about metric used, see Ingress Gateway Metrics.
Expression	sum(rate(oc_ingressgateway_http_request_ratelimit_reject_chain_length_histogram_bucket{namespace="$namespace"}[2m])) by (le)

Table 6-560 Discard Request Action Traffic Rate

Field	Details
Description	Measures the rate of requests rejected or processed based on the discard request action. For more information about metric used, see Ingress Gateway Metrics.
Expression	sum(irate(oc_ingressgateway_http_request_ratelimit_denied_count_total{namespace="$namespace"}[2m])) by (Action)

Table 6-561 Pod Congestion Level

Field	Details
Description	Measures the congestion level of the pod. For more information about metric used, see Ingress Gateway Metrics.
Expression	sum(rate(oc_ingressgateway_congestion_level_bucket_total{namespace="$namespace"}[2m])) by (le)

6.2.3.12 Egress Gateway Pod Protection Using Rate Limiting

Table 6-562 Allowed Requests

Field	Details
Description	Measures the number of consecutive rejects before a successful accept. The metric indicates rejections due to traffic burst.For more information about metric used, see Egress Gateway Metrics.
Expression	sum(rate(oc_egressgateway_http_request_ratelimit_reject_chain_length_histogram_bucket{namespace="$namespace"}[2m])) by (le)

Table 6-563 Reject Chain Length

Field	Details
Description	Measures the number of consecutive rejects before a successful accept. The metric indicates rejections due to traffic burst. For more information about metric used, see Egress Gateway Metrics.
Expression	sum(rate(oc_egressgateway_http_request_ratelimit_reject_chain_length_histogram_bucket{namespace="$namespace"}[2m])) by (le)

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-564 Discard Action

Field	Details
Description	Measures the rate of requests rejected or processed based on the discard request action.For more information about metric used, see Egress Gateway Metrics.
Expression	sum(rate(oc_egressgateway_http_request_ratelimit_denied_count_total{namespace="$namespace"}[2m])) by (Action)

Table 6-565 Congestion Level

Field	Details
Description	Measures the congestion level of the pod.For more information about metric used, see Egress Gateway Metrics.
Expression	sum(rate(oc_egressgateway_congestion_level_bucket_total{namespace="$namespace"}[2m])) by (le)

6.3 NRF Alerts

This section includes information about the NRF alerts.

The following table describes the various alert levels generated by NRF:

Table 6-566 Alerts Levels or Severity Types

Alerts Levels/Severity Types	Definition
Critical	Indicates a severe issue that poses a significant risk to safety, security, or operational integrity. It requires immediate response to address the situation and prevent serious consequences. Raised for conditions may affect the service of NRF.
Major	Indicates a more significant issue that has an impact on operations or poses a moderate risk. It requires prompt attention and action to mitigate potential escalation. Raised for conditions may affect the service of NRF.
Minor	Indicates a situation that is low in severity and does not pose an immediate risk to safety, security, or operations. It requires attention but does not demand urgent action. Raised for conditions may affect the service of NRF.
Info or Warn (Informational)	Provides general information or updates that are not related to immediate risks or actions. These alerts are for awareness and do not typically require any specific response. WARN and INFO alerts may not impact the service of NRF.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> - Summary or dimensions may vary based on deployment.
> - The alert triggering time varies as per the environment in which it is deployed.
> - The performance and capacity of the NRF system may vary based on the call model, Feature or Interface configuration, and underlying CNE and hardware environment.

6.3.1 System Level Alerts

This section lists the system level alerts.

6.3.1.1 OcnrfNfStatusUnavailable

Table 6-567 OcnrfNfStatusUnavailable

Field	Details
Description	'OCNRF services unavailable'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : All OCNRF services are unavailable.'
Severity	Critical
Condition	When all the NRF services are unavailable, either because the NRF is getting deployed or purged. The NRF services considered are nfregistration, nfsubscription, nrfauditor, nrfconfiguration, nfaccesstoken, nfdiscovery, appinfo, ingressgateway, and egressgateway.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7016
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-567 (Cont.) OcnrfNfStatusUnavailable

Field	Details

Recommended Actions	The alert is cleared automatically when the NRF services restart.
	Steps:
	1. Check for service-specific alerts which may be causing the issues with service exposure.
	2. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	a. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	3. Refer to the application logs on Kibana and check for database related failures such as connectivity and invalid secrets. The logs can be filtered based on the services.
	4. Check for helm status to make sure there are no errors:
	`$ helm status <helm release name of the desired NF> -n <namespace>`
	If it is not in “STATUS : DEPLOYED”, then capture logs and event again.
	5. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on the Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.2 OcnrfPodsRestart

Table 6-568 OcnrfPodsRestart

Field	Details
Description	'Pod *<Pod Name>* has restarted.
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : A Pod has restarted'
Severity	Major
Condition	A pod belonging to any of the NRF services have restarted.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7017

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-568 (Cont.) OcnrfPodsRestart

Field	Details
Metric Used	'kube_pod_container_status_restarts_total' **Note**: This is a Kubernetes metric. If this metric is not available, use a similar metric as exposed by the monitoring system.

Recommended Actions	The alert is cleared automatically if the specific pod is up.
	Steps:
	1. Refer to the application logs on Kibana and filter based on pod name, check for database related failures such as connectivity and Kubernetes secrets.
	2. To check the orchestration logs for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	3. Check the database status. For more information, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
	4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on the Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.3 NnrfNFManagementServiceDown

Table 6-569 NnrfNFManagementServiceDown

Field	Details
Description	'OCNRF Nnrf_Management service <*nfregistration*|*nfsubscription*|*nrfauditor*> is down'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : NFManagement service is down'
Severity	Critical
Condition	This alert is raised when either NFRegistration, NFSubscription, or NrfAuditor services are unavailable.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7018

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-569 (Cont.) NnrfNFManagementServiceDown

Field	Details
Metric Used	"up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-569 (Cont.) NnrfNFManagementServiceDown

Field	Details

Recommended Actions	The alert is cleared when all the Nnrf_NFManagement services nfregistration, nfsubscription, and nrfauditor are available.
	Steps:
	1. Check if NfService specific alerts are generated to understand which service is down.
	Either some or all of the following alerts are generated based on which services are down
	• OcnrfRegistrationServiceDown
	• OcnrfSubscriptionServiceDown
	• OcnrfAuditorServiceDown
	2. To check the orchestration logs for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	3. Check for the POD's status if they are in "Running" state using the following command:
	`$ kubectl get pod –n <namespace>`
	If it is not in "Running" state, capture the pod logs and events by running the following command:
	`$ kubectl get events --sort-by=.metadata.creationTimestamp -n <namespace>`
	4. Refer to the application logs on Kibana and filter based on aforementioned service names. Check for ERROR WARNING logs for each of these services.
	5. Check the database status. For more information on how to check the database status, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
	6. Refer to the application logs on Kibana and filter the service appinfo, check for the service status of the nfregistration, nfsubscription, and nrfauditor services.
	7. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-569 (Cont.) NnrfNFManagementServiceDown

Field	Details
	Oracle Communications Cloud Native Core, Network Function Data Collector User Guide.
Available in OCI	No

6.3.1.4 NnrfAccessTokenServiceDown

Table 6-570 NnrfAccessTokenServiceDown

Field	Details
Description	'OCNRF Nnrf_NFAccessToken service nfaccesstoken is down'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : NFAccessToken service down'
Severity	Critical
Condition	This alert is raised when NFAccessToken service is unavailable.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7020
Metric Used	"up" **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available use a similar metric as exposed by the monitoring system.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-570 (Cont.) NnrfAccessTokenServiceDown

Field	Details

Recommended Actions	The alert is cleared when the Nnrf_AccessToken service is available.
	Steps:
	1. To check the orchestration logs of nfaccesstoken service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter based on nfaccesstoken service names. Check for ERROR WARNING logs.
	3. Check the DB status. For more information on how to check the DB status, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
	4. Depending on the failure reason, take the resolution steps.
	5. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.5 NnrfNFDiscoveryServiceDown

Table 6-571 NnrfNFDiscoveryServiceDown

Field	Details
Description	'OCNRF Nnrf_NFDiscovery service *nfdiscovery* is down'
Applicable in OCI	No
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : NFDiscovery service down'
Severity	Critical
Condition	NFDiscovery is unavailable.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7019

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-571 (Cont.) NnrfNFDiscoveryServiceDown

Field	Details
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.

Recommended Actions	The alert is cleared when the Nnrf_NFDiscovery service is available.
	Steps:
	1. To check the orchestration logs of nfdiscovery service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter based on nfdiscovery service names. Check for ERROR WARNING logs.
	3. Check the DB status. For more information on how to check the DB status, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
	4. Depending on the failure reason, take the resolution steps.
	5. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.6 OcnrfRegistrationServiceDown

Table 6-572 OcnrfRegistrationServiceDown

Field	Details
Description	'OCNRF NFRegistration service *nfregistration* is down'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : NFRegistration service is down'
Severity	Critical
Condition	None of the pods of the NFRegistration microservice is available.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7021

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-572 (Cont.) OcnrfRegistrationServiceDown

Field	Details
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.

Recommended Actions	The alert is cleared when the nfregistration service is available.
	Steps:
	1. To check the orchestration logs of nfregistration service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter based on nfregistration service names. Check for ERROR WARNING logs.
	3. Check the DB status. For more information on how to check the DB status, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*. Depending on the failure reason, take the resolution steps.
	4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.7 OcnrfSubscriptionServiceDown

Table 6-573 OcnrfSubscriptionServiceDown

Field	Details
Description	'OCNRF NFSubscription service *nfsubscription* is down.
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : NFSubscription service is down'
Severity	Critical
Condition	None of the pods of the NFSubscription microservice is available.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7022

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-573 (Cont.) OcnrfSubscriptionServiceDown

Field	Details
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.

Recommended Actions	The alert is cleared when the nfsubscription service is available.
	Steps:
	1. To check the orchestration logs of nfsubscription service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter based on nfsubcription service names. Check for ERROR WARNING logs.
	3. Check the DB status. For more information on how to check the DB status, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
	4. Depending on the failure reason, take the resolution steps.
	5. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.8 OcnrfDiscoveryServiceDown

Table 6-574 OcnrfDiscoveryServiceDown

Field	Details
Description	'OCNRF NFDiscovery service *nfdiscovery* is down'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : NFDiscovery service down'
Severity	Critical
Condition	None of the pods of the NFDiscovery microservice is available.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7023

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-574 (Cont.) OcnrfDiscoveryServiceDown

Field	Details
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.

Recommended Actions	The alert is cleared when the nfdiscovery service is available.
	Steps:
	1. To check the orchestration logs of nfregistration service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter based on nfdiscovery service names. Check for ERROR WARNING logs.
	3. Check the DB status. For more information on how to check the DB status, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
	4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.9 OcnrfAccessTokenServiceDown

Table 6-575 OcnrfAccessTokenServiceDown

Field	Details
Description	'OCNRF NFAccessToken service *nfaccesstoken* is down
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : NFAccesstoken service down'
Severity	Critical
Condition	None of the pods of the NFAccessToken microservice is available.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7024

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-575 (Cont.) OcnrfAccessTokenServiceDown

Field	Details
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.

Recommended Actions	The alert is cleared when the nfaccesstoken service is available.
	Steps:
	1. To check the orchestration logs of nfaccesstoken service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter based on nfaccesstoken service names. Check for ERROR WARNING logs.
	3. Check the DB status. For more information on how to check the DB status, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
	4. Depending on the failure reason, take the resolution steps.
	5. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.10 OcnrfAuditorServiceDown

Table 6-576 OcnrfAuditorServiceDown

Field	Details
Description	'OCNRF NrfAuditor service *nrfauditor* is down'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : NrfAuditor service down'
Severity	Critical
Condition	None of the pods of the NrfAuditor microservice is available.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7026

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-576 (Cont.) OcnrfAuditorServiceDown

Field	Details
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.

Recommended Actions	The alert is cleared when the nrfauditor service is available.
	Steps:
	1. To check the orchestration logs of nrfauditor service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter based on nrfauditor service names. Check for ERROR WARNING logs related to thread exceptions.
	3. Check the DB status. For more information on how to check the DB status, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
	4. Depending on the failure reason, take the resolution steps.
	5. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.11 OcnrfConfigurationServiceDown

Table 6-577 OcnrfConfigurationServiceDown

Field	Details
Description	'OCNRF NrfConfiguration service *nrfconfiguration* is down'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : NrfConfiguration service down'
Severity	Critical
Condition	None of the pods of the NrfConfiguration microservice is available.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7025

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-577 (Cont.) OcnrfConfigurationServiceDown

Field	Details
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.

Recommended Actions	The alert is cleared when the nrfconfiguration service is available.
	Steps:
	1. To check the orchestration logs of nrfconfiguration service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter based on nrfconfiguration service names. Check for ERROR WARNING logs.
	3. Check the DB status. For more information on how to check the DB status, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
	4. Depending on the failure reason, take the resolution steps.
	5. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.12 OcnrfAppInfoServiceDown

Table 6-578 OcnrfAppInfoServiceDown

Field	Details
Description	'OCNRF Appinfo service *appinfo* is down'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Appinfo service down'
Severity	Critical
Condition	None of the pods of the appinfo microservice is available.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7027

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-578 (Cont.) OcnrfAppInfoServiceDown

Field	Details
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.

Recommended Actions	The alert is cleared when the appinfo service is available.
	Steps:
	1. To check the orchestration logs of appinfo service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter based on appinfo service names. Check for ERROR WARNING logs related to thread exceptions.
	3. Depending on the failure reason, take the resolution steps.
	4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.13 OcnrfArtisanServiceDown

Table 6-579 OcnrfArtisanServiceDown

Field	Details
Description	'OCNRF NrfArtisan service {{$labels.app_kubernetes_io_name}} is down'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : NrfArtisan service is down'
Severity	Critical
Condition	NrfArtisan is unavailable.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7056

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-579 (Cont.) OcnrfArtisanServiceDown

Field	Details
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use the similar metric as exposed by the monitoring system.

Recommended Actions	The alert is cleared when the NrfArtisan service is available.
	Steps:
	1. To check the orchestration logs of NrfArtisan service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get pod -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter the logs based on NrfArtisan service names. Check for ERROR and WARNING logs related to thread exceptions.
	3. Check the database status. For more information, see the *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
	4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.14 OcnrfAlternateRouteServiceDown

Table 6-580 OcnrfAlternateRouteServiceDown

Field	Details
Description	'OCNRF AlternateRoute service {{$labels.app_kubernetes_io_name}} is down'
Applicable in OCI	No
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : AlternateRoute service is down'
Severity	Critical
Condition	AlternateRoute is unavailable.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7057

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-580 (Cont.) OcnrfAlternateRouteServiceDown

Field	Details
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use the similar metric as exposed by the monitoring system.

Recommended Actions	The alert is cleared when the alternate-route service is available.
	Steps:
	1. To check the orchestration logs of alternate-route service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get pod -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter the logs based on Alternate-Route service names. Check for ERROR and WARNING logs related to thread exceptions.
	3. Check the database status. For more information, see the *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
	4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

6.3.1.15 OcnrfPerfInfoServiceDown

Table 6-581 OcnrfPerfInfoServiceDown

Field	Details
Description	'OCNRF Perfinfo service {{$labels.app_kubernetes_io_name}} is down'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Perfinfo service down'
Severity	Critical
Condition	Perfinfo is unavailable.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7058
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use the similar metric as exposed by the monitoring system.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-581 (Cont.) OcnrfPerfInfoServiceDown

Field	Details

Recommended Actions	The alert is cleared when the Perfinfo service is available.
	Steps:
	1. To check the orchestration logs of Perfinfo service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get pod -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter the logs based on Perf-Info service names. Check for ERROR and WARNING logs related to thread exceptions.
	3. Check the database status. For more information, see the *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
	4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.16 OcnrfIngressGatewayServiceDown

Table 6-582 OcnrfIngressGatewayServiceDown

Field	Details
Description	'OCNRF Ingress-Gateway service *ingressgateway* is down'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Ingress-gateway service down'
Severity	Critical
Condition	None of the pods of the Ingress Gateway microservice is available.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7028
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-582 (Cont.) OcnrfIngressGatewayServiceDown

Field	Details

Recommended Actions	The alert is cleared when the Ingress Gateway service is available.
	Steps:
	1. To check the orchestration logs of Ingress Gateway service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter based on Ingress Gateway service names. Check for ERROR WARNING logs related to thread exceptions.
	3. Depending on the failure reason, take the resolution steps.
	4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.17 OcnrfEgressGatewayServiceDown

Table 6-583 OcnrfEgressGatewayServiceDown

Field	Details
Description	'OCNRF Egress-Gateway service *egressgateway* is down'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Egress-Gateway service down'
Severity	Critical
Condition	None of the pods of the Egress Gateway microservice is available.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7029
Metric Used	'up' **Note**: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-583 (Cont.) OcnrfEgressGatewayServiceDown

Field	Details

Recommended Actions	The alert is cleared when the Egress Gateway service is available.
	Steps:
	1. To check the orchestration logs of Egress Gateway service and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the *Running* state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the *Running* state.
	2. Refer to the application logs on Kibana and filter based on Egress Gateway service names. Check for ERROR WARNING logs related to thread exceptions.
	3. Depending on the failure reason, take the resolution steps.
	4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.1.18 OcnrfTotalIngressTrafficRateAboveMinorThreshold

Table 6-584 OcnrfTotalIngressTrafficRateAboveMinorThreshold

Field	Details
Description	'Total Ingress traffic Rate is above configured minor threshold. (current value is: {{ $value }})'
Summary	'timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Traffic Rate is above 80 Percent of Max requests per second'
Severity	Minor
Condition	The total NRF Ingress Message rate has crossed the configured minor threshold of 800 TPS. Default value of this alert trigger point in alert file is when NRF Ingress Rate crosses 80 % of 1000 (Maximum ingress request rate).
OID	1.3.6.1.4.1.323.5.3.36.1.2.7001
Metric Used	'oc_ingressgateway_http_requests_total'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-584 (Cont.) OcnrfTotalIngressTrafficRateAboveMinorThreshold

Field	Details
Recommended Actions	The alert is cleared either when the total Ingress Traffic rate falls below the minor threshold or when the total traffic rate crosses the major threshold, in which case the OcnrfTotalIngressTrafficRateAboveMajorThreshold alert is raised. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. **Steps**: 1. Refer Grafana to determine which service is receiving high traffic. 2. Refer **Ingress gateway** section in Grafana to determine the increase in 4xx and 5xx error codes. 3. Check **Ingress gateway** logs on Kibana to determine the reason for the errors.
Available in OCI	No

6.3.1.19 OcnrfTotalIngressTrafficRateAboveMajorThreshold

Table 6-585 OcnrfTotalIngressTrafficRateAboveMajorThreshold

Field	Details
Description	'Total Ingress traffic Rate is above major threshold. (current value is: {{ $value }})'
Summary	'timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Traffic Rate is above 90 Percent of Max requests per second'
Severity	Major
Condition	The total NRF Ingress Message rate has crossed the configured major threshold of 900 TPS. Default value of this alert trigger point in the alert file is when NRF Ingress Rate crosses 90 % of 1000 (Maximum ingress request rate).
OID	1.3.6.1.4.1.323.5.3.36.1.2.7002
Metric Used	'oc_ingressgateway_http_requests_total'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-585 (Cont.) OcnrfTotalIngressTrafficRateAboveMajorThreshold

Field	Details
Recommended Actions	The alert is cleared when the total Ingress Traffic rate falls below the major threshold or when the total traffic rate crosses the critical threshold, in which case the OcnrfTotalIngressTrafficRateAboveCriticalThreshold alert is raised. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. **Steps**: 1. Refer Grafana to determine which service is receiving high traffic. 2. Refer **Ingress gateway** section in Grafana to determine the increase in 4xx and 5xx error codes. 3. Check **Ingress gateway** logs on Kibana to determine the reason for the errors.
Available in OCI	No

6.3.1.20 OcnrfTotalIngressTrafficRateAboveCriticalThreshold

Table 6-586 OcnrfTotalIngressTrafficRateAboveCriticalThreshold

Field	Details
Description	'Total Ingress traffic Rate is above critical threshold.(current value is: {{ $value }})'
Summary	'timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Traffic Rate is more than 52000 requests per second'
Severity	Critical
Condition	The total NRF Ingress Message rate has crossed the configured critical threshold of 52000 TPS. Default value of this alert trigger point in the alert file is when NRF Ingress Rate crosses 52000 TPS.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7003
Metric Used	'oc_ingressgateway_http_requests_total'
Recommended Actions	The alert is cleared when the Ingress traffic rate falls below the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. **Steps**: 1. Refer Grafana to determine which service is receiving high traffic. 2. Refer **Ingress gateway** section in Grafana to determine the increase in 4xx and 5xx error codes. 3. Check **Ingress gateway** logs on Kibana to determine the reason for the errors.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-586 (Cont.) OcnrfTotalIngressTrafficRateAboveCriticalThreshold

Field	Details
Available in OCI	No

6.3.1.21 OcnrfTransactionErrorRateAbove0Dot1Percent

Table 6-587 OcnrfTransactionErrorRateAbove0Dot1Percent

Field	Details
Description	'Transaction Error rate is above 0.1 Percent of Total Transactions (current value is {{ $value }})'
Summary	'timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Transaction Error Rate detected above 0.1 Percent of Total Transactions'
Severity	Warning
Condition	The number of failed transactions is above 0.1 percent of the total transactions.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7004
Metric Used	'oc_ingressgateway_http_responses_total'
Recommended Actions	The alert is cleared when the number of failure transactions is below 0.1 percent of the total transactions or when the number of failed transactions crosses the 1% threshold, in which case the OcnrfTransactionErrorRateAbove1Percent is raised. **Steps**: 1. Check the service specific metrics to understand the specific service request errors. For example: ocnrf_nfDiscover_tx_responses_total with statusCode ~= 2xx. 2. Check metrics per service, per method: For example, discovery requests can be determined from the following metrics: Metrics="oc_ingressgateway_http_responses_total" Method="GET" NFServiceType="nnrf-disc" Route_path="/nnrf-disc/v1/nf-instances/**" Status="503 SERVICE_UNAVAILABLE" 3. If guidance is required, contact My Oracle Support.
Available in OCI	No

6.3.1.22 OcnrfTransactionErrorRateAbove1Percent

Table 6-588 OcnrfTransactionErrorRateAbove1Percent

Field	Details
Description	'Transaction Error rate is above 1 Percent of Total Transactions (current value is {{ $value }})'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-588 (Cont.) OcnrfTransactionErrorRateAbove1Percent

Field	Details
Summary	'timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Transaction Error Rate detected above 1 Percent of Total Transactions'
Severity	Warning
Condition	When the number of failed transactions is above 1 percent of the total transactions.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7005
Metric Used	'oc_ingressgateway_http_responses_total'
Recommended Actions	The alert is cleared when the number of failure transactions is below 1% of the total transactions or when the number of failed transactions crosses the 10% threshold, in which case the OcnrfTransactionErrorRateAbove10Percent is raised. **Steps**: 1. Check the service specific metrics to understand the specific service request errors. For example: ocnrf_nfDiscover_tx_responses_total with statusCode ~= 2xx. 2. Check metrics per service, per method: For example, discovery requests can be determined from the following metrics: Metrics="oc_ingressgateway_http_responses_total" Method="GET" NFServiceType="nnrf-disc" Route_path="/nnrf-disc/v1/nf-instances/**" Status="503 SERVICE_UNAVAILABLE" 3. If guidance is required, contact My Oracle Support.
Available in OCI	No

6.3.1.23 OcnrfTransactionErrorRateAbove10Percent

Table 6-589 OcnrfTransactionErrorRateAbove10Percent

Field	Details
Description	'Transaction Error rate is above 10 Percent of Total Transactions (current value is {{ $value }})'
Summary	'timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Transaction Error Rate detected above 10 Percent of Total Transactions'
Severity	Minor
Condition	The number of failed transactions has crossed the minor threshold of 10 percent of the total transactions.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7006
Metric Used	'oc_ingressgateway_http_responses_total'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-589 (Cont.) OcnrfTransactionErrorRateAbove10Percent

Field	Details
Recommended Actions	The alert is cleared when the number of failure transactions is below 10 percent of the total transactions or when the number of failed transactions crosses the 25 percent threshold, in which case the OcnrfTransactionErrorRateAbove25Percent is raised. **Steps**: 1. Check the service specific metrics to understand the specific service request errors. For example: ocnrf_nfDiscover_tx_responses_total with statusCode ~= 2xx. 2. Check metrics per service, per method: For example, discovery requests can be determined from the following metrics: Metrics="oc_ingressgateway_http_responses_total" Method="GET" NFServiceType="nnrf-disc" Route_path="/nnrf-disc/v1/nf-instances/**" Status="503 SERVICE_UNAVAILABLE" 3. If guidance is required, contact My Oracle Support.
Available in OCI	No

6.3.1.24 OcnrfTransactionErrorRateAbove25Percent

Table 6-590 OcnrfTransactionErrorRateAbove25Percent

Field	Details
Description	'Transaction Error rate is above 25 Percent of Total Transactions (current value is {{ $value }})'
Summary	'timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Transaction Error Rate detected above 25 Percent of Total Transactions'
Severity	Major
Condition	The number of failed transactions has crossed the minor threshold of 25 percent of the total transactions.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7007
Metric Used	'oc_ingressgateway_http_responses_total'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-590 (Cont.) OcnrfTransactionErrorRateAbove25Percent

Field	Details
Recommended Actions	The alert is cleared when the number of failure transactions is below 25 percent of the total transactions or when the number of failed transactions crosses the 50 percent threshold, in which case the OcnrfTransactionErrorRateAbove50Percent is raised. **Steps**: 1. Check the service specific metrics to understand the specific service request errors. For example: ocnrf_nfDiscover_tx_responses_total with statusCode ~= 2xx. 2. Check metrics per service, per method: For example, discovery requests can be determined from the following metrics: Metrics="oc_ingressgateway_http_responses_total" Method="GET" NFServiceType="nnrf-disc" Route_path="/nnrf-disc/v1/nf-instances/**" Status="503 SERVICE_UNAVAILABLE" 3. If guidance is required, contact My Oracle Support.
Available in OCI	No

6.3.1.25 OcnrfTransactionErrorRateAbove50Percent

Table 6-591 OcnrfTransactionErrorRateAbove50Percent

Field	Details
Description	'Transaction Error rate is above 50 Percent of Total Transactions (current value is {{ $value }})'
Summary	'timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Transaction Error Rate detected above 50 Percent of Total Transactions'
Severity	Critical
Condition	The number of failed transactions has crossed the minor threshold of 50 percent of the total transactions.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7008
Metric Used	'oc_ingressgateway_http_responses_total'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-591 (Cont.) OcnrfTransactionErrorRateAbove50Percent

Field	Details
Recommended Actions	The alert is cleared when the number of failure transactions is below 50 percent of the total transactions. **Steps**: 1. Check the service specific metrics to understand the specific service request errors. For example: ocnrf_nfDiscover_tx_responses_total with statusCode ~= 2xx. 2. Check metrics per service, per method: For example, discovery requests can be determined from the following metrics: Metrics="oc_ingressgateway_http_responses_total" Method="GET" NFServiceType="nnrf-disc" Route_path="/nnrf-disc/v1/nf-instances/**" Status="503 SERVICE_UNAVAILABLE" 3. If guidance is required, contact My Oracle Support.
Available in OCI	No

6.3.1.26 OcnrfTotalEgressTrafficRateAboveCriticalThreshold

Table 6-592 OcnrfTotalEgressTrafficRateAboveCriticalThreshold

Field	Details
Description	'Egress traffic rate is above the configured critical threshold. (current value is: {{ $value }})'
Summary	"kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Traffic Rate is above 51600 requests per second'
Severity	Critical
Condition	This alarm is raised when the Egress traffic rate is greater than the critical configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7109
Metric Used	oc_egressgateway_http_requests_total
Recommended Actions	The alert is cleared either when the total discovery rate falls below the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. Steps: 1. Refer Grafana to determine which service is receiving high traffic. 2. Refer Egress Gateway section in Grafana to determine the increase in 4xx and 5xx error codes. 3. Check Egress Gateway logs on Kibana to determine the reason for the errors.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.1.27 OcnrfTotalForwardingTrafficRateAboveCriticalThreshold

Table 6-593 OcnrfTotalForwardingTrafficRateAboveCriticalThreshold

Field	Details
Description	'NRF-NRF Forwarding Rate is above the configured critical threshold. (current value is: {{ $value }})'
Summary	'kubernetes_namespace: $labels.kubernetes_namespace, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Traffic Rate is above 5200 requests per second.'
Severity	Critical
Condition	This alarm is raised when the rate between NRF and NRF Forwarding is greater than the critical configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7110
Metric Used	ocnrf_forward_nfDiscover_tx_requests_total
Recommended Actions	The alert is cleared either when the total NRF Forwarding rate falls below the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. Steps: 1. Refer Grafana to determine which service is receiving high traffic. 2. Refer NRF Forwarding section in Grafana to determine the increase in 4xx and 5xx error codes. 3. Check NRF Forwarding logs on Kibana to determine the reason for the errors.
Available in OCI	No

6.3.1.28 OcnrfHeapUsageCrossedMinorThreshold

Table 6-594 OcnrfHeapUsageCrossedMinorThreshold

Field	Details
Description	'OCNRF Heap Usage for pod {{ $labels.pod }} has crossed the configured minor threshold (50%) (value={{ $value }}) of its limit.'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Heap Usage of pod exceeded 50% of its limit.'
Severity	Minor
Condition	This alert is raised when the Java memory heap usage of pods exceeds the configured minor threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7126
Metric Used	jvm_memory_used_bytes

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-594 (Cont.) OcnrfHeapUsageCrossedMinorThreshold

Field	Details
Recommended Actions	The alert is cleared when the heap usage of pods falls below the minor threshold. **Note**: The threshold is configurable in the alert file. If this alert is unexpected, contact My Oracle Support. **Steps:** 1. Identify the pods which are raised in the alert. 2. Refer Resource usage in Grafana dashboard to get the memory usage. 3. Collect the pod logs and top output of the pods. 4. Contact My Oracle Support.
Available in OCI	No

6.3.1.29 OcnrfHeapUsageCrossedMajorThreshold

Table 6-595 OcnrfHeapUsageCrossedMajorThreshold

Field	Details
Description	'OCNRF Heap Usage for pod {{ $labels.pod }} has crossed the configured major threshold (60%) (value={{ $value }}) of its limit.'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Heap Usage of pod is more than or equal to 60% and less than 70% of its limit.'
Severity	Major
Condition	This alert is raised when the Java memory heap usage of pods exceeds the configured major threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7127
Metric Used	jvm_memory_used_bytes
Recommended Actions	The alert is cleared when the heap usage of pods falls below the major threshold. **Note**: The threshold is configurable in the alert file. If this alert is unexpected, contact My Oracle Support. **Steps:** 1. Identify the pods which are raised in the alert. 2. Refer Resource usage in Grafana dashboard to get the memory usage. 3. Collect the pod logs and top output of the pods. 4. Contact My Oracle Support.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.1.30 OcnrfHeapUsageCrossedCriticalThreshold

Table 6-596 OcnrfHeapUsageCrossedCriticalThreshold

Field	Details
Description	'OCNRF Heap Usage for pod {{ $labels.pod }} has crossed the configured critical threshold (70%) (value={{ $value }}) of its limit.'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Heap Usage of pod is more than 70% of its limit.'
Severity	Critical
Condition	This alert is raised when the Java memory heap usage of pods exceeds the configured critical threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7128
Metric Used	jvm_memory_used_bytes
Recommended Actions	The alert is cleared when the heap usage of pods falls below the critical threshold. **Note**: The threshold is configurable in the alert file. If this alert is unexpected, contact My Oracle Support. **Steps:** 1. Identify the pods which are raised in the alert. 2. Refer Resource usage in Grafana dashboard to get the memory usage. 3. Collect the pod logs and top output of the pods. 4. Contact My Oracle Support.
Available in OCI	No

6.3.2 Service Level Alerts

This section lists the service level alerts.

6.3.2.1 OcnrfAccessTokenRequestsRejected

Table 6-597 OcnrfAccessTokenRequestsRejected

Field	Details
Description	'AccessToken request(s) have been rejected by OCNRF (current value is: {{ $value }})'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} AccessToken Request has been rejected by OCNRF.'
Severity	Warning
Condition	NRF rejected an AccessToken Request
OID	1.3.6.1.4.1.323.5.3.36.1.2.7014
Metric Used	'ocnrf_accessToken_tx_responses_total'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-597 (Cont.) OcnrfAccessTokenRequestsRejected

Field	Details
Recommended Actions	The alert is cleared automatically. **Steps**: 1. The Rejection Reason is present in the alert. 2. In case the RejectionReason is AuthScreeningFailed/ ClientNotAuthorized, either the configurations need to be reevaluated or check the consumer NF that has requested for unauthorized token. For more information about token information, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*. 3. For other reasons, follow the RejectionReason.
Available in OCI	No

6.3.2.2 OcnrfAuditorMultiplePodUnavailable

Table 6-598 OcnrfAuditorMultiplePodUnavailable

Field	Details
Description	Ocnrf Auditor Multiple Pods are Unavailable in deployment
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Ocnrf Auditor Multiple Pods are Unavailable'
Severity	Critical
Condition	Ocnrf Auditor Multiple Pods are Unavailable.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7075
Metric Used	NA
Recommended Actions	This alert is raised due to auditor multiple pods are unavailable. This alert is cleared automatically when the pods are available.
Available in OCI	No

6.3.2.3 OcnrfAppInfoMultiplePodUnavailable

Table 6-599 OcnrfAppInfoMultiplePodUnavailable

Field	Details
Description	Ocnrf AppInfo Multiple Pods are Unavailable in deployment
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Ocnrf AppInfo Multiple Pods are Unavailable'
Severity	Critical
Condition	Ocnrf Auditor Multiple Pods are Unavailable.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7076
Metric Used	NA
Recommended Actions	This alert is raised due to App-Info multiple pods are unavailable. This alert is cleared automatically when the pods are available.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-599 (Cont.) OcnrfAppInfoMultiplePodUnavailable

Field	Details
Available in OCI	No

6.3.2.4 OcnrfPerfInfoMultiplePodUnavailable

Table 6-600 OcnrfPerfInfoMultiplePodUnavailable

Field	Details
Description	Ocnrf PerfInfo Multiple Pods are Unavailable in deployment
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Ocnrf PerfInfo Multiple Pods are Unavailable'
Severity	Critical
Condition	Ocnrf PerfInfo Multiple Pods are Unavailable.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7077
Metric Used	NA
Recommended Actions	This alert is raised due to perf-Info multiple pods are unavailable. This alert is cleared automatically when the pods are available.
Available in OCI	No

6.3.2.5 OcnrfTotalSLFRateAboveCriticalThreshold

Table 6-601 OcnrfTotalSLFRateAboveCriticalThreshold

Field	Details
Description	'NRF-SLF Rate is above the configured critical threshold. (current value is: {{ $value }})'
Summary	'kubernetes_namespace: $labels.kubernetes_namespace, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Traffic Rate is above 45600 requests per second.'
Severity	Critical
Condition	This alarm is raised when the rate between NRF and SLF reaches is greater than the critical configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7111
Metric Used	ocnrf_SLF_tx_requests_total
Recommended Actions	The alert is cleared either when the total SLF rate falls below the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. Steps: 1. Refer Grafana to determine which service is receiving high traffic. 2. Refer SLF section in Grafana to determine the increase in 4xx and 5xx error codes. 3. Check SLF logs on Kibana to determine the reason for the errors.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.2.6 OcnrfTotalDiscoveryRateAboveCriticalThreshold

Table 6-602 OcnrfTotalDiscoveryRateAboveCriticalThreshold

Field	Details
Description	'Total Discovery Rate is above the configured critical threshold. (current value is: {{ $value }})'
Summary	'kubernetes_namespace: $labels.kubernetes_namespace, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Traffic Rate is above 51600 requests per second.'
Severity	Critical
Condition	This alarm is raised when the total discovery rate is greater than the critical configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7112
Metric Used	ocnrf_nfDiscover_rx_requests_total
Recommended Actions	The alert is cleared when the total discovery rate falls below the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. Steps: 1. Refer Grafana to determine which service is receiving high traffic. 2. Refer Discovery section in Grafana to determine the increase in 4xx and 5xx error codes. 3. Check Discovery logs on Kibana to determine the reason for the errors.
Available in OCI	No

6.3.2.7 OcnrfAccessTokenRequestsAboveThreshold

Table 6-603 OcnrfAccessTokenRequestsAboveThreshold

Field	Details
Description	'Total Access token request rate is above the configured critical threshold. (current value is: {{ $value }})'
Summary	'namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: Total Access token request rate is above 5'
Severity	Critical
Condition	The alert is raised when the rate of Access Token requests is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7115
Metric Used	ocnrf_accessToken_rx_requests_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-603 (Cont.) OcnrfAccessTokenRequestsAboveThreshold

Field	Details
Recommended Actions	The alert is cleared when the total number of access token request rate falls below the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. **Steps**: 1. Refer the NfAccessToken Section in Grafana to determine increase in TPS. 2. Refer the Grafana to determine increase in failure responses.
Available in OCI	No

6.3.2.8 OcnrfNfUpdateRequestsAboveThreshold

Table 6-604 OcnrfNfUpdateRequestsAboveThreshold

Field	Details
Description	'Total NfUpdate request rate is above the configured critical threshold. (current value is: {{ $value }})'
Summary	'namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: Total NfUpdate request rate is above 5'
Severity	Critical
Condition	This alert is raised when the total number of NfUpdate requests is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7116
Metric Used	ocnrf_nfUpdate_rx_requests_total
Recommended Actions	The alert is cleared when the total number of NfUpdate request falls below the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. **Steps**: 1. Refer the NfRegister Section in Grafana to determine increase in TPS. 2. Refer the Grafana to determine increase in failure responses.
Available in OCI	No

6.3.2.9 OcnrfRegisteredNfCountAboveThreshold

Table 6-605 OcnrfRegisteredNfCountAboveThreshold

Field	Details
Description	'Total Number of active registrations in OCNRF is above critical threshold. (current value is: {{ $value }})'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-605 (Cont.) OcnrfRegisteredNfCountAboveThreshold

Field	Details
Summary	'namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: Total Number of active registrations in OCNRF is above 260'
Severity	Critical
Condition	The alert is raised when the total number of NFs registered in the set is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7118
Metric Used	ocnrf_nf_registered_count
Recommended Actions	The alert is cleared when the total number active registrations in NRF falls below the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional registrations. If this alert is unexpected, contact My Oracle Support. **Step**: 1. Refer Grafana to determine the number of NFs per nfType.
Available in OCI	No

6.3.2.10 OcnrfNfHeartBeatRequestsAboveThreshold

Table 6-606 OcnrfNfHeartBeatRequestsAboveThreshold

Field	Details
Description	'Total NfHeartBeat request rate is above the configured critical threshold. (current value is: {{ $value }})'
Summary	'namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: Total NfHeartBeat request rate is above 52'
Severity	Critical
Condition	This alert is raised when the total number of NfHeartBeat requests is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7117
Metric Used	ocnrf_nfHeartBeat_rx_requests_total
Recommended Actions	The alert is cleared when the total number of NfHeartBeat request falls below the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. **Steps**: 1. Refer the NfRegister Section in Grafana to determine increase in TPS. 2. Refer the Grafana to determine increase in failure responses.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.2.11 OcnrfRegisteredNfCountAboveThreshold

Table 6-607 OcnrfRegisteredNfCountAboveThreshold

Field	Details
Description	'Total Number of active registrations in OCNRF is above critical threshold. (current value is: {{ $value }})'
Summary	'namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: Total Number of active registrations in OCNRF is above 260'
Severity	Critical
Condition	The alert is raised when the total number of NFs registered in the set is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7118
Metric Used	ocnrf_nf_registered_count
Recommended Actions	The alert is cleared when the total number active registrations in NRF falls below the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional registrations. If this alert is unexpected, contact My Oracle Support. **Step**: 1. Refer Grafana to determine the number of NFs per nfType.
Available in OCI	No

6.3.2.12 OcnrfTotalSubscriptionsAboveThreshold

Table 6-608 OcnrfTotalSubscriptionsAboveThreshold

Field	Details
Description	'Total Number of active subscriptions in OCNRF is above the critical threshold. (current value is: {{ $value }})'
Summary	'namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: Total Number of active subscriptions in OCNRF is above 1000.'
Severity	Critical
Condition	This alert is raised when the total number of active subscriptions in NRF is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7121
Metric Used	ocnrf_nfset_active_subscriptions

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-608 (Cont.) OcnrfTotalSubscriptionsAboveThreshold

Field	Details
Recommended Actions	The alert is cleared when the total number active subscriptions in NRF is less than the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF has received additional subscriptions (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. **Steps**: 1. Refer Grafana to determine the total number of subscriptions created. 2. Verify if Subscription Limit feature has been enabled using `subscriptionLimit.featureStatus` parameter. For more information, see *Oracle Communications Cloud Native Core, Network Repository Function User Guide*. 3. Assess which NFs are creating the additional subscriptions.
Available in OCI	No

6.3.2.13 OcnrfDiscoveryRequestsForUDRAboveThreshold

Table 6-609 OcnrfDiscoveryRequestsForUDRAboveThreshold

Field	Details
Description	'Total NfDiscover request rate for nfType UDR is above the configured critical threshold. (current value is: {{ $value }})'
Summary	'namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: Total NfDiscover request rate for nfType UDR is above above 700'
Severity	Critical
Condition	This alert is raised when the rate of nfDiscover requests for nfType UDR is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7122
Metric Used	ocnrf_nfDiscover_rx_requests_total
Recommended Actions	The alert is cleared when the rate of nfDiscover requests for nfType UDR is below than the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic for UDR. If this alert is unexpected, contact My Oracle Support.
Available in OCI	No

6.3.2.14 OcnrfDiscoveryRequestsForUDMAboveThreshold

Table 6-610 OcnrfDiscoveryRequestsForUDMAboveThreshold

Field	Details
Description	'Total NfDiscover request rate for nfType UDM is above the configured critical threshold. (current value is: {{ $value }})'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-610 (Cont.) OcnrfDiscoveryRequestsForUDMAboveThreshold

Field	Details
Summary	'namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: Total NfDiscover request rate for nfType UDM is above above 46000'
Severity	Critical
Condition	This alert is raised when the rate of nfDiscover requests for nfType UDM is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7123
Metric Used	ocnrf_nfDiscover_rx_requests_total
Recommended Actions	The alert is cleared when the rate of nfDiscover requests for nfType UDM is below than the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic for UDM. If this alert is unexpected, contact My Oracle Support.
Available in OCI	No

6.3.2.15 OcnrfDiscoveryRequestsForAMFAboveThreshold

Table 6-611 OcnrfDiscoveryRequestsForAMFAboveThreshold

Field	Details
Description	'Total NfDiscover request rate for nfType AMF is above the configured critical threshold. (current value is: {{ $value }})'
Summary	'namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: Total NfDiscover request rate for nfType AMF is above 2500'
Severity	Critical
Condition	This alert is raised when the rate of nfDiscover requests for nfType AMF is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7124
Metric Used	ocnrf_nfDiscover_rx_requests_total
Recommended Actions	The alert is cleared when the rate of nfDiscover requests for nfType AMF is below than the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic for AMF. If this alert is unexpected, contact My Oracle Support.
Available in OCI	No

6.3.2.16 OcnrfDiscoveryRequestsForSMFAboveThreshold

Table 6-612 OcnrfDiscoveryRequestsForSMFAboveThreshold

Field	Details
Description	'Total NfDiscover request rate for nfType SMF is above the configured critical threshold. (current value is: {{ $value }})'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-612 (Cont.) OcnrfDiscoveryRequestsForSMFAboveThreshold

Field	Details
Summary	'namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: Total NfDiscover request rate for nfType SMF is above 4500'
Severity	Critical
Condition	This alert is raised when the rate of nfDiscover requests for nfType SMF is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7125
Metric Used	ocnrf_nfDiscover_rx_requests_total
Recommended Actions	The alert is cleared when the rate of nfDiscover requests for nfType SMF is below than the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic for SMF. If this alert is unexpected, contact My Oracle Support.
Available in OCI	No

6.3.3 NfProfile Status Change Alerts

This section lists the alerts raised when there is status change in NfProfile.

6.3.3.1 OcnrfRegisteredPCFsBelowCriticalThreshold

Table 6-613 OcnrfRegisteredPCFsBelowCriticalThreshold

Field	Details
Description	'The number of registered NFs detected below critical threshold (current value is: {{ $value }})'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nftype: {{$labels.RequesterNfType}}, nrflevel:{{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The number of registered NFs detected below critical threshold.'
Severity	Critical
Condition	The number of NFs of the given NFType PCF currently registered with NRF is below the critical threshold. **Note**: Operator can add similar alerts for each NfType and configure the corresponding thresholds as required. Default value of this alert trigger point in the alert file is when registered PCFs count with NRF is below 2.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7009
Metric Used	'ocnrf_active_registrations_count'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-613 (Cont.) OcnrfRegisteredPCFsBelowCriticalThreshold

Field	Details
Recommended Actions	The alert is cleared when the number of registered PCFs is above the critical threshold. **Steps**: 1. Check if there is traffic for requests other than registration (for example, discovery requests). This ensures that NRF FQDN is reachable from other NFs and Ingress Gateway is up and running. 2. Check if ingress gateway pod is up and running: `kubectl get po -n <namespace>` 3. Check for registration pod logs on Kibana for ERROR WARN logs. 4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Notes	1. Operator can configure the threshold values to the number of NFs of type PCF expected within the network. 2. PCFs with NFStatus as 'SUSPENDED' or "UNDISCOVERABLE' are considered as unregistered. 3. Operator can configure the RequesterNfType expected within the network. 4. Operator can add similar alerts for each NfType and configure the corresponding thresholds as required.
Available in OCI	No

6.3.3.2 OcnrfRegisteredPCFsBelowMajorThreshold

Table 6-614 OcnrfRegisteredPCFsBelowMajorThreshold

Field	Details
Description	'The number of registered NFs detected below major threshold (current value is: {{ $value }})'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nftype: {{$labels.NfType}}, nrflevel:{{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The number of registered NFs detected below major threshold.'
Severity	Major
Condition	The number of NFs of the given NFType PCF currently registered with NRF is below the major threshold. **Note**: Operator can add similar alerts for each NfType and configure the corresponding thresholds as required. Default value of this alert trigger point in the alert file is when Registered PCFs count with NRF is greater than or equal to 2 and below 10.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-614 (Cont.) OcnrfRegisteredPCFsBelowMajorThreshold

Field	Details
OID	1.3.6.1.4.1.323.5.3.36.1.2.7010
Metric Used	'ocnrf_active_registrations_count'
Recommended Actions	The alert is cleared when the number of registered PCFs is above the major threshold. **Steps**: 1. Check if there is traffic for requests other than registration (for example, discovery requests). This ensures that NRF FQDN is reachable from other NFs and Ingress Gateway is up and running. 2. Check if Ingress Gateway pod is up and running: `kubectl get po -n <namespace>` 3. Check for registration pod logs on Kibana for ERROR WARN logs. 4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Notes	1. Operator can configure the threshold values with respect to the number of NFs of type PCF expected within the network. 2. PCFs with NFStatus as 'SUSPENDED' or "UNDISCOVERABLE' are considered as unregistered. 3. Operator can configure the RequesterNfType expected within the network. 4. Operator can add similar alerts for each NfType and configure the corresponding thresholds as required.
Available in OCI	No

6.3.3.3 OcnrfRegisteredPCFsBelowMinorThreshold

Table 6-615 OcnrfRegisteredPCFsBelowMinorThreshold

Field	Details
Description	'The number of registered NFs detected below minor threshold (current value is: {{ $value }})'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nftype: {{$labels.NfType}}, nrflevel:{{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The number of registered NFs detected below minor threshold.'
Severity	Minor

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-615 (Cont.) OcnrfRegisteredPCFsBelowMinorThreshold

Field	Details
Condition	The number of NFs of the given NFType PCF currently registered with NRF is below the minor threshold. **Note**: Operator can add similar alerts for each NfType and configure the corresponding thresholds as required. Default value of this alert trigger point in the alert file is when registered PCFs count with NRF is greater than or equal to 10 and below 20.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7011
Metric Used	'ocnrf_active_registrations_count'
Recommended Actions	The alert is cleared when the number of registered PCFs is above the minor threshold. **Steps**: 1. Check if there is traffic for requests other than registration (for example, discovery requests). This ensures that NRF FQDN is reachable from other NFs and Ingress Gateway is up and running. 2. Check if ingress gateway pod is up and running: `kubectl get po -n <namespace>` 3. Check for Registration pod logs on Kibana for ERROR WARN logs. 4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Notes	1. Operator can configure the threshold values with respect to the number of NFs of type PCF expected within the network. 2. PCFs with NFStatus as 'SUSPENDED' or "UNDISCOVERABLE' are considered as unregistered. 3. Operator can configure the RequesterNfType expected within the network. 4. Operator can add similar alerts for each NfType and configure the corresponding thresholds as required.
Available in OCI	No

6.3.3.4 OcnrfRegisteredPCFsBelowThreshold

Table 6-616 OcnrfRegisteredPCFsBelowThreshold

Field	Details
Description	'The number of registered NFs is approaching minor threshold (current value is: {{ $value }})'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-616 (Cont.) OcnrfRegisteredPCFsBelowThreshold

Field	Details
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nftype: {{$labels.NfType}}, nrflevel:{{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The number of registered NFs approaching minor threshold.'
Severity	Warning
Condition	The number of NFs of the given NFType PCF currently registered with NRF is approaching minor threshold. **Note**: Operator can add similar alerts for each NfType and configure the corresponding thresholds as required. Default value of this alert trigger point in the alert file is when registered PCFs count with NRF is greater than or equal to 20 and below 30.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7012
Metric Used	'ocnrf_active_registrations_count'
Recommended Actions	The alert is cleared when the number of registered PCFs is approaching minor threshold. **Steps**: 1. Check if there is traffic for requests other than registration (for example, discovery requests). This ensures that NRF FQDN is reachable from other NFs and Ingress Gateway is up and running. 2. Check if Ingress Gateway pod is up and running: `kubectl get po -n <namespace>` 3. Check for Registration pod logs on Kibana for ERROR WARN logs. 4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Notes	1. Operator can configure the threshold values with respect to the number of NFs of type PCF expected within the network. 2. PCFs with NFStatus as 'SUSPENDED' or "UNDISCOVERABLE' are considered as unregistered. 3. Operator can configure the RequesterNfType expected within the network. 4. Operator can add similar alerts for each NfType and configure the corresponding thresholds as required.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.3.5 OcnrfTotalNFsRegisteredBelowCriticalThreshold

Table 6-617 OcnrfTotalNFsRegisteredBelowCriticalThreshold

Field	Details
Description	'Number of active registrations in OCNRF (current value is: {{ $value }}) is below critical threshold'
Summary	kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Active registrations count.
Severity	Critical
Condition	The total number of NFs currently in "REGISTERED" state with the NRF is below the critical threshold. **Note**: The threshold values are provided as an example. User can configure the threshold value as per the requirement.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7042
Metric Used	'ocnrf_active_registrations_count'
Recommended Actions	The alert is cleared when the number of registered NFs is above the critical threshold. **Steps**: 1. Check if there is traffic for requests other than registration (for example, discovery requests). This ensures that NRF FQDN is reachable from other NFs and Ingress Gateway is up and running. 2. Check if Ingress Gateway pod is up and running: `kubectl get po -n <namespace>` 3. Check for registration pod logs on Kibana for ERROR WARN logs. 4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
Notes	1. Operator can configure the threshold values with respect to the number of NFs expected within the network. 2. NFs with NFStatus as 'SUSPENDED' or "UNDISCOVERABLE' are not considered as registered.
Available in OCI	Yes

6.3.3.6 OcnrfTotalNFsRegisteredBelowMajorThreshold

Table 6-618 OcnrfTotalNFsRegisteredBelowMajorThreshold

Field	Details
Description	'Number of active registrations in OCNRF (current value is: {{ $value }}) is below major threshold'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-618 (Cont.) OcnrfTotalNFsRegisteredBelowMajorThreshold

Field	Details
Summary	kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Active registrations count.
Severity	Major
Condition	The total number of NFs currently in "REGISTERED" state with the NRF is below the major threshold. **Note**: The threshold values are provided as an example. The user can configure the threshold value as per the requirement.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7043
Metric Used	'ocnrf_active_registrations_count'
Recommended Actions	The alert is cleared when the number of registered NFs is above the major threshold. **Steps**: 1. Check if there is traffic for requests other than registration (for example, discovery requests). This ensures that NRF FQDN is reachable from other NFs and Ingress Gateway is up and running. 2. Check if Ingress Gateway pod is up and running: `kubectl get po -n <namespace>` 3. Check for Registration pod logs on Kibana for ERROR WARN logs. 4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on capturing logs, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Notes	1. Operator can configure the threshold values with respect to the number of NFs expected within the network. 2. NFs with NFStatus as 'SUSPENDED' or "UNDISCOVERABLE' are not considered as registered.
Available in OCI	Yes

6.3.3.7 OcnrfTotalNFsRegisteredBelowMinorThreshold

Table 6-619 OcnrfTotalNFsRegisteredBelowMinorThreshold

Field	Details
Description	'Number of active registrations in OCNRF (current value is: {{ $value }}) is below minor threshold'
Summary	kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Active registrations count.
Severity	Minor

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-619 (Cont.) OcnrfTotalNFsRegisteredBelowMinorThreshold

Field	Details
Condition	The total number of NFs currently in "REGISTERED" state with the NRF is below the minor threshold. **Note**: The threshold values are provided as an example. The user can configure the threshold value as per the requirement.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7044
Metric Used	'ocnrf_active_registrations_count'
Recommended Actions	The alert is cleared when the number of registered NFs is above the minor threshold. **Steps**: 1. Check if there is traffic for requests other than registration (for example, discovery requests). This ensures that NRF FQDN is reachable from other NFs and Ingress Gateway is up and running. 2. Check if Ingress Gateway pod is up and running: `kubectl get po -n <namespace>` 3. Check for registration pod logs on Kibana for ERROR WARN logs. 4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on capturing logs, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Notes	1. Operator can configure the threshold values with respect to the number of NFs expected within the network. 2. NFs with NFStatus as 'SUSPENDED' or "UNDISCOVERABLE' are not considered as registered.
Available in OCI	Yes

6.3.3.8 OcnrfTotalNFsRegisteredApproachingMinorThreshold

Table 6-620 OcnrfTotalNFsRegisteredApproachingMinorThreshold

Field	Details
Description	'Number of active registrations in OCNRF (current value is: {{ $value }}) is approaching minor threshold'
Summary	kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Active registrations count.
Severity	Info
Condition	The total number of NFs currently in "REGISTERED" state with the NRF is approaching minor threshold. **Note**: The threshold values provided as an example. The user can configure the threshold as per need.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7045
Metric Used	'ocnrf_active_registrations_count'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-620 (Cont.) OcnrfTotalNFsRegisteredApproachingMinorThreshold

Field	Details
Recommended Actions	The alert is cleared when the number of registered NFs are approaching minor threshold. **Steps**: No action is required. This is an information alert.
Notes	1. Operator can configure the threshold values with respect to the number of NFs expected within the network. 2. NFs with NFStatus as 'SUSPENDED' or "UNDISCOVERABLE' are not considered as registered.
Available in OCI	Yes

6.3.3.9 OcnrfNFStatusTransitionToRegistered

Table 6-621 OcnrfNFStatusTransitionToRegistered

Field	Details
Description	'NF with NF profile fqdn {{$labels.NfProfileFqdn}} NF instance id {{$labels.NfInstanceId}} NF type {{$labels.NfType}} is REGISTERED , previous status was {{$labels.PreviousStatus}}'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}},podname: {{$labels.kubernetes_pod_name}},NfInstanceId: {{$labels.NfInstanceId}},NfProfileFqdn: {{$labels.NfProfileFqdn}},NfType: {{$labels.NfType}},PreviousStatus: {{$labels.PreviousStatus}},NewStatus: {{$labels.NewStatus}},timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} NF is REGISTERED.'
Severity	Info
Condition	NF Instance's status transitions to REGISTERED. **Note**: When multiple alerts are present for a given NF, the latest alert is always considered. The timestamp can also be seen in the "Active Since" field of the alert in Prometheus.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7046
Metric Used	ocnrf_nfInstance_status_change_total
Recommended Actions	The alert is cleared automatically after a window of 5 minutes. **Steps**: No action is required. This is an information alert.
Available in OCI	Yes

6.3.3.10 OcnrfNFServiceStatusTransitionToRegistered

Table 6-622 OcnrfNFServiceStatusTransitionToRegistered

Field	Details
Description	'NF service {{$labels.NfServiceName}} and service instance id {{$labels.NfServiceInstanceId}} of NF profile fqdn {{$labels.NfProfileFqdn}} and instance id {{$labels.NfInstanceId}} is REGISTERED, previous status was {{$labels.PreviousStatus}}'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-622 (Cont.) OcnrfNFServiceStatusTransitionToRegistered

Field	Details
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}},podname: {{$labels.kubernetes_pod_name}},NfInstanceId: {{$labels.NfInstanceId}},NfServiceName: {{$labels.NfServiceName}},NfServiceInstanceId: {{$labels.NfServiceInstanceId}},NfProfileFqdn: {{$labels.NfProfileFqdn}},NfServiceFqdn: {{$labels.NfServiceFqdn}},PreviousStatus: {{$labels.PreviousStatus}},NewStatus: {{$labels.NewStatus}},timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} NF service is REGISTERED.'
Severity	Info
Condition	Status of an NF Instance's service transitions to REGISTERED. **Note**: When multiple alerts are present for a given NF, the latest alert is always considered. The timestamp can also be seen in the "Active Since" field of the alert in Prometheus.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7047
Metric Used	ocnrf_nfService_status_change_total
Recommended Actions	The alert is cleared automatically after a window of 5 minutes. **Steps**: No action is required. This is an information alert.
Available in OCI	Yes

6.3.3.11 OcnrfNFStatusTransitionToSuspended

Table 6-623 OcnrfNFStatusTransitionToSuspended

Field	Details
Description	'NF with NF profile fqdn {{$labels.NfProfileFqdn}} NF instance id {{$labels.NfInstanceId}} NF type {{$labels.NfType}} is SUSPENDED, previous status was {{$labels.PreviousStatus}}'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}},podname: {{$labels.kubernetes_pod_name}},NfInstanceId: {{$labels.NfInstanceId}},NfProfileFqdn: {{$labels.NfProfileFqdn}},NfType: {{$labels.NfType}},PreviousStatus: {{$labels.PreviousStatus}},NewStatus: {{$labels.NewStatus}},timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} NF is SUSPENDED.'
Severity	Major
Condition	NF Instance's status transitions to SUSPENDED. **Note**: When multiple alerts are present for a given NF, the latest alert is always considered. The timestamp can also be seen in the "Active Since" field of the alert in Prometheus.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7048
Metric Used	ocnrf_nfInstance_status_change_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-623 (Cont.) OcnrfNFStatusTransitionToSuspended

Field	Details
Recommended Actions	The alert is cleared automatically after a window of 5 minutes. **Steps**: 1. Check logs in NRF registration pod for failing patch requests or check Jaeger traces to see traces for incoming requests. 2. Check Ingress Gateway logs to see if the requests are coming. 3. Check if the NRF pods are UP. 4. Check for the Ingress Gateway metrics in Prometheus for PATCH requests or responses in this time frame. Confirm if the responses have any non-2xx error codes. 5. Depending on the failure reason, take the resolution steps. 6. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on capturing logs, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Available in OCI	Yes

6.3.3.12 OcnrfNFServiceStatusTransitionToSuspended

Table 6-624 OcnrfNFServiceStatusTransitionToSuspended

Field	Details
Description	'NF service {{$labels.NfServiceName}} and service instance id {{$labels.NfServiceInstanceId}} of NF profile fqdn {{$labels.NfProfileFqdn}} and instance id {{$labels.NfInstanceId}} is SUSPENDED, previous status was {{$labels.PreviousStatus}}'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}},podname: {{$labels.kubernetes_pod_name}},NfInstanceId: {{$labels.NfInstanceId}},NfServiceName: {{$labels.NfServiceName}},NfServiceInstanceId: {{$labels.NfServiceInstanceId}},NfProfileFqdn: {{$labels.NfProfileFqdn}},NfServiceFqdn: {{$labels.NfServiceFqdn}},PreviousStatus: {{$labels.PreviousStatus}},NewStatus: {{$labels.NewStatus}},timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} NF service is SUSPENDED.'
Severity	Minor
Condition	Status of an NF Instance's service transitions to SUSPENDED. **Note**: When multiple alerts are present for a given NF, the latest alert is always considered. The timestamp can also be seen in the "Active Since" field of the alert in Prometheus.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7049
Metric Used	ocnrf_nfService_status_change_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-624 (Cont.) OcnrfNFServiceStatusTransitionToSuspended

Field	Details
Recommended Actions	The alert is cleared automatically after a window of 5 minutes. **Steps**: 1. Check logs in NRF registration pod for failing patch requests or check Jaeger traces to see traces for incoming requests. 2. Check Ingress Gateway logs to see if the requests are coming. 3. Check if the NRF pods are UP. 4. Check for the Ingress Gateway metrics in Prometheus for PATCH requests or responses in this time frame. Confirm if the responses have any non-2xx error codes. 5. Depending on the failure reason, take the resolution steps. 6. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on capturing logs, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Available in OCI	Yes

6.3.3.13 OcnrfNFStatusTransitionToUndiscoverable

Table 6-625 OcnrfNFStatusTransitionToUndiscoverable

Field	Details
Description	'NF with NF profile fqdn {{$labels.NfProfileFqdn}} NF instance id {{$labels.NfInstanceId}} NF type {{$labels.NfType}} is UNDISCOVERABLE, previous status was {{$labels.PreviousStatus}}'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}},podname: {{$labels.kubernetes_pod_name}},NfInstanceId: {{$labels.NfInstanceId}},NfProfileFqdn: {{$labels.NfProfileFqdn}},NfType: {{$labels.NfType}},PreviousStatus: {{$labels.PreviousStatus}},NewStatus: {{$labels.NewStatus}},timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} NF is UNDISCOVERABLE.'
Severity	Info
Condition	NF Instance's status transitions to UNDISCOVERABLE. **Note**: When multiple alerts are present for a given NF, the latest alert is always considered. The timestamp can also be seen in the "Active Since" field of the alert in Prometheus.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7050
Metric Used	ocnrf_nfInstance_status_change_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-625 (Cont.) OcnrfNFStatusTransitionToUndiscoverable

Field	Details
Recommended Actions	The alert is cleared automatically after a window of 5 minutes. **Steps**: 1. Check logs in NRF registration pod to verify if the NF has sent UNDISCOVERABLE status in NFRegister or NfUpdate requests or check Jaeger traces to see traces for incoming requests. 2. If there is no such incoming request, collect the logs and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on capturing logs, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Available in OCI	Yes

6.3.3.14 OcnrfNFServiceStatusTransitionToUndiscoverable

Table 6-626 OcnrfNFServiceStatusTransitionToUndiscoverable

Field	Details
Description	'NF service {{$labels.NfServiceName}} and service instance id {{$labels.NfServiceInstanceId}} of NF profile fqdn {{$labels.NfProfileFqdn}} and instance id {{$labels.NfInstanceId}} is UNDISCOVERABLE, previous status was {{$labels.PreviousStatus}}'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}},podname: {{$labels.kubernetes_pod_name}},NfInstanceId: {{$labels.NfInstanceId}},NfServiceName: {{$labels.NfServiceName}},NfServiceInstanceId: {{$labels.NfServiceInstanceId}},NfProfileFqdn: {{$labels.NfProfileFqdn}},NfServiceFqdn: {{$labels.NfServiceFqdn}},PreviousStatus: {{$labels.PreviousStatus}},NewStatus: {{$labels.NewStatus}},timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} NF service is UNDISCOVERABLE.'
Severity	Info
Condition	Status of an NF Instance's service transitions to UNDISCOVERABLE. **Note**: When multiple alerts are present for a given NF, the latest alert is always considered. The timestamp can also be seen in the "Active Since" field of the alert in Prometheus.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7051
Metric Used	ocnrf_nfService_status_change_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-626 (Cont.) OcnrfNFServiceStatusTransitionToUndiscoverable

Field	Details
Recommended Actions	The alert is cleared automatically after a window of 5 minutes. **Steps**: 1. Check logs in NRF registration pod to verify if the NF has sent UNDISCOVERABLE status in NFRegister or NfUpdate requests or check Jaeger traces to see traces for incoming requests. 2. If there is no such incoming request, collect the logs and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on capturing logs, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Available in OCI	Yes

6.3.3.15 OcnrfNFStatusTransitionToDeregistered

Table 6-627 OcnrfNFStatusTransitionToDeregistered

Field	Details
Description	'NF with NF profile fqdn {{$labels.NfProfileFqdn}} NF instance id {{$labels.NfInstanceId}} NF type {{$labels.NfType}} is DEREGISTERED, previous status was {{$labels.PreviousStatus}}'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}},podname: {{$labels.kubernetes_pod_name}},NfInstanceId: {{$labels.NfInstanceId}},NfProfileFqdn: {{$labels.NfProfileFqdn}},NfType: {{$labels.NfType}},PreviousStatus: {{$labels.PreviousStatus}},NewStatus: {{$labels.NewStatus}},timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} NF is DEREGISTERED.'
Severity	Info
Condition	NF Instance's status transitions to DEREGISTERED. **Note**: When multiple alerts are present for a given NF, the latest alert is always considered. The timestamp can also be seen in the "Active Since" field of the alert in Prometheus.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7052
Metric Used	ocnrf_nfInstance_status_change_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-627 (Cont.) OcnrfNFStatusTransitionToDeregistered

Field	Details
Recommended Actions	The alert is cleared automatically after a window of 5 minutes. **Steps**: 1. Check logs in NRF registration pod for failing patch requests or check Jaeger traces to see traces for incoming requests. 2. Check Ingress Gateway logs to see if the requests are coming. 3. Check if the NRF pods are UP. 4. Check for the Ingress Gateway metrics in Prometheus for PATCH requests or responses in this time frame. Confirm if the responses have any non 2xx error codes. 5. Depending on the failure reason, take the resolution steps. 6. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on capturing logs, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Available in OCI	Yes

6.3.3.16 OcnrfNFServiceStatusTransitionToDeregistered

Table 6-628 OcnrfNFServiceStatusTransitionToDeregistered

Field	Details
Description	'NF service {{$labels.NfServiceName}} and service instance id {{$labels.NfServiceInstanceId}} of NF profile fqdn {{$labels.NfProfileFqdn}} and instance id {{$labels.NfInstanceId}} is DEREGISTERED, previous status was {{$labels.PreviousStatus}}'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}},podname: {{$labels.kubernetes_pod_name}},NfInstanceId: {{$labels.NfInstanceId}},NfServiceName: {{$labels.NfServiceName}},NfServiceInstanceId: {{$labels.NfServiceInstanceId}},NfProfileFqdn: {{$labels.NfProfileFqdn}},NfServiceFqdn: {{$labels.NfServiceFqdn}},PreviousStatus: {{$labels.PreviousStatus}},NewStatus: {{$labels.NewStatus}},timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} NF service is DEREGISTERED.'
Severity	Info
Condition	Status of an NF Instance's service transitions to DEREGISTERED. **Note**: When multiple alerts are present for a given NF, the latest alert is always considered. The timestamp can also be seen in the "Active Since" field of the alert in Prometheus.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7053
Metric Used	ocnrf_nfService_status_change_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-628 (Cont.) OcnrfNFServiceStatusTransitionToDeregistered

Field	Details
Recommended Actions	The alert is cleared automatically after a window of 5 minutes. **Steps**: 1. Check logs in NRF registration pod for failing patch requests or check Jaeger traces to see traces for incoming requests. 2. Check Ingress Gateway logs to see if the requests are coming. 3. Check if the NRF pods are UP. 4. Check for the Ingress Gateway metrics in Prometheus for PATCH requests or responses in this time frame. Confirm if the responses have any non 2xx error codes. 5. Depending on the failure reason, take the resolution steps. 6. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on capturing logs, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Available in OCI	Yes

6.3.3.17 OcnrfNfProfileSizeAboveThreshold

Table 6-629 OcnrfNfProfileSizeAboveThreshold

Field	Details
Description	'The size of the NF profile is above the critical threshold. (current value is: {{ $value }})'
Summary	"namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: The size of the NF profile is above 12kB threshold'
Severity	Critical
Condition	This alert is raised when the size of the NF profile is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7119
Metric Used	ocnrf_nf_profile_size
Recommended Actions	The alert is cleared when the size of the NF profile is less than the critical threshold. **Note**: The threshold is configurable in the alert file. **Step**:Verify which NF has registered a nfProfile above the threshold size, using the `nfInstanceId` in the `ocnrf_nf_profile_size` metric.
Available in OCI	No

6.3.3.18 OcnrfDiscoveryResponseSizeAboveThreshold

Table 6-630 OcnrfDiscoveryResponseSizeAboveThreshold

Field	Details
Description	'The size of nfDiscover response is above the critical threshold. (current value is: {{ $value }})'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-630 (Cont.) OcnrfDiscoveryResponseSizeAboveThreshold

Field	Details
Summary	'namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: The size of nfDiscover response is above 45kB threshold''
Severity	Critical
Condition	This alert is raised when the size of the nfDiscover response is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7120
Metric Used	ocnrf_nfDiscover_tx_response_size_bytes_max
Recommended Actions	The alert is cleared when the size of the nfDiscover response is less than the critical threshold. **Note**: The threshold is configurable in the alert file. **Step**: 1. Refer Grafana to check for which `targetNfType` triggers discovery response with size greater than the threshold. Higher discovery response may impact NRF discovery performance. If the alert is unexpected, contact My Oracle Support.
Available in OCI	No

6.3.4 Feature Specific Alerts

This section lists the feature specific alerts.

6.3.4.1 KeyID for AccessToken Feature

This section lists the alerts that are specific to KeyID for AccessToken feature. For more information about the feature, see the Key-ID for Access Token section.

6.3.4.1.1 OcnrfAccessTokenCurrentKeyIdNotConfigured

Table 6-631 OcnrfAccessTokenCurrentKeyIdNotConfigured

Field	Details
Description	'AccessToken request(s) have been rejected by OCNRF (current value is: {{ $value }})'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} AccessToken Request has been rejected by OCNRF as Current Key Id is not configured.'
Severity	Critical
Condition	NRF Access Token Rejected due to CurrentKeyId not configured
OID	1.3.6.1.4.1.323.5.3.36.1.2.7033
Metric Used	'ocnrf_accessToken_tx_responses_total'
Recommended Actions	The alert is automatically cleared as it is raised when NRF receives Access Token Request, and at that point, Current Key Id is not selected. For more information about configuring `currentKeyID` parameter, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-631 (Cont.) OcnrfAccessTokenCurrentKeyIdNotConfigured

Field	Details
Available in OCI	No

6.3.4.1.2 OcnrfAccessTokenCurrentKeyIdInvalidDetails

Table 6-632 OcnrfAccessTokenCurrentKeyIdInvalidDetails

Field	Details
Description	'AccessToken request(s) have been rejected by OCNRF (current value is: {{ $value }})'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, KeyType: {{$labels.KeyType}}, RejectionReason: {{$labels.RejectionReason}},timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} AccessToken Request has been rejected by OCNRF as CurrentKeyId details are invalid.'
Severity	Critical
Condition	NRF Access Token Rejected due to token signing details corresponding to CurrentKeyId are invalid.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7034
Metric Used	'ocnrf_accessToken_tx_responses_total'
Recommended Actions	The alert is automatically cleared when NRF receives Access Token Request, and at that point, Current Key Id details are invalid. For more information about configuring `currentKeyID` parameter, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
Available in OCI	No

6.3.4.1.3 OcnrfOauthCurrentKeyNotConfigured

Table 6-633 OcnrfOauthCurrentKeyNotConfigured

Field	Details
Description	'OCNRF Oauth Access token Current Key Id is not configured'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} OCNRF Oauth Access token Current Key Id is not configured.'
Severity	Critical
Condition	Oauth Current Key ID is not configured
OID	1.3.6.1.4.1.323.5.3.36.1.2.7035
Metric Used	ocnrf_oauth_currentKeyId_configuredStatus

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-633 (Cont.) OcnrfOauthCurrentKeyNotConfigured

Field	Details
Recommended Actions	The alert is cleared when the current key ID is configured. **Steps**: Configure valid current key ID in Access Token Configuration. For more information about configuring `currentKeyID` parameter, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
Available in OCI	No

6.3.4.1.4 OcnrfOauthCurrentKeyDataHealthStatus

Table 6-634 OcnrfOauthCurrentKeyDataHealthStatus

Field	Details
Description	'OCNRF Oauth Access token Current Key Id status is not healthy'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, KeyId: {{$labels.KeyId}}, KeyType: {{$labels.KeyType}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} OCNRF Oauth Access token Current Key Id status is not healthy.'
Severity	Critical
Condition	Oauth Current Key ID details health is not good.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7036
Metric Used	ocnrf_oauth_keyData_healthStatus
Recommended Actions	The alert is cleared when the current key ID status is healthy. **Steps**: Rectify the condition by checking ErrorCondition For example: For ErrorCondition Invalid_Key_Details, check if the k8SecretName, k8SecretNameSpace, and filename combination exists correctly for both privateKey and certificate. Make sure that the pem file data is not corrupt or the certificate has not expired.
Available in OCI	No

6.3.4.1.5 OcnrfOauthNonCurrentKeyDataHealthStatus

Table 6-635 OcnrfOauthNonCurrentKeyDataHealthStatus

Field	Details
Description	'OCNRF Oauth Access token Non current Key Id status is not healthy'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, KeyId: {{$labels.KeyId}}, KeyType: {{$labels.KeyType}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} OCNRF Oauth Access token non current Key Id status is not healthy.'
Severity	Info
Condition	Oauth Non Current Key details health is not good
OID	1.3.6.1.4.1.323.5.3.36.1.2.7037
Metric Used	ocnrf_oauth_keyData_healthStatus

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-635 (Cont.) OcnrfOauthNonCurrentKeyDataHealthStatus

Field	Details
Recommended Actions	The alert is cleared when the current key ID status is healthy. **Steps**: Rectify the condition by checking ErrorCondition For example: For ErrorCondition Invalid_Key_Details, check if the k8SecretName, k8SecretNameSpace, and filename combination exists correctly for both privateKey and certificate. Make sure that the pem file data is not corrupt or the certificate has not expired.
Available in OCI	No

6.3.4.1.6 OcnrfOauthCurrentCertificateExpiringIn1Week

Table 6-636 OcnrfOauthCurrentCertificateExpiringIn1Week

Field	Details
Description	'OCNRF Oauth Access token current Key Id certificate is expiring in less than 1 week'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, KeyId: {{$labels.KeyId}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} OCNRF Oauth Access token current Key Id certificate is expiring in less than 1 week.'
Severity	Critical
Condition	Oauth Current Key ID details are expiring in less than 1 week
OID	1.3.6.1.4.1.323.5.3.36.1.2.7038
Metric Used	ocnrf_oauth_keyData_expiryStatus
Recommended Actions	The alert is cleared when the key expiry time is more than 1 week. **Steps**: Replace expiring certificate key pair with new ones. For more information on creating certificate key pair, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
Available in OCI	No

6.3.4.1.7 OcnrfOauthNonCurrentCertificateExpiringIn1Week

Table 6-637 OcnrfOauthNonCurrentCertificateExpiringIn1Week

Field	Details
Description	'OCNRF Oauth Access token non current Key Id certificate is expiring in less than 1 week'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, KeyId: {{$labels.KeyId}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} OCNRF Oauth Access token non current Key Id certificate is expiring in less than 1 week.'
Severity	Info
Condition	Oauth Non Current Key ID details are expiring in less than 1 week

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-637 (Cont.) OcnrfOauthNonCurrentCertificateExpiringIn1Week

Field	Details
OID	1.3.6.1.4.1.323.5.3.36.1.2.7039
Metric Used	ocnrf_oauth_keyData_expiryStatus
Recommended Actions	The alert is cleared when the key expiry time is more than 1 week. **Steps**: Replace expiring certificate key pair with new ones. For more information on creating certificate key pair, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
Available in OCI	No

6.3.4.1.8 OcnrfOauthCurrentCertificateExpiringIn30days

Table 6-638 OcnrfOauthCurrentCertificateExpiringIn30days

Field	Details
Description	'OCNRF Oauth Access token current Key Id certificate is expiring in less than 30 days'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, KeyId: {{$labels.KeyId}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} OCNRF Oauth Access token current Key Id certificate is expiring in less than 30 days.'
Severity	Major
Condition	Oauth Current Key ID details are expiring in more than 24 hours and less than 30 days
OID	1.3.6.1.4.1.323.5.3.36.1.2.7040
Metric Used	ocnrf_oauth_keyData_expiryStatus
Recommended Actions	The alert is cleared when certificate for the current key id's expiry time is more than 30 days. **Steps**: Replace expiring certificate key pair with new ones. For more information on creating certificate key pair, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
Available in OCI	No

6.3.4.1.9 OcnrfOauthNonCurrentCertificateExpiringIn30days

Table 6-639 OcnrfOauthNonCurrentCertificateExpiringIn30days

Field	Details
Description	'OCNRF Oauth Access token non current Key Id certificate is expiring in less than 30 days'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-639 (Cont.) OcnrfOauthNonCurrentCertificateExpiringIn30days

Field	Details
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, KeyId: {{$labels.KeyId}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} OCNRF Oauth Access token non current Key Id certificate is expiring in less than 30 days.'
Severity	Info
Condition	Oauth Non Current Key ID details are expiring in more than 24 hours and less than 30 days
OID	1.3.6.1.4.1.323.5.3.36.1.2.7041
Metric Used	ocnrf_oauth_keyData_expiryStatus
Recommended Actions	The alert is cleared when certificate for the non-current key id's certificate expiry time is more than 30 days. **Steps**: Replace expiring certificate key pair with new ones. For more information on creating certificate key pair, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
Available in OCI	No

6.3.4.2 Overload Control Based on Percentage Discards Feature

This section lists the alerts that are specific to Overload Control Based on Percentage Discards feature. For more information about the feature, see the Overload Control section.

6.3.4.2.1 OcnrfMemoryUsageCrossedMinorThreshold

Table 6-640 OcnrfMemoryUsageCrossedMinorThreshold

Field	Details
Description	'OCNRF Memory Usage for pod <*Pod name*> has crossed the configured minor threshold (50 %) (value={{ $value }}) of its limit.'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Memory Usage of pod exceeded 50% of its limit.'
Severity	Minor
Condition	A pod has reached the configured minor threshold (50%) of its memory resource limits.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7030
Metric Used	'container_memory_usage_bytes' and 'container_spec_memory_limit_bytes' **Note**: This is a Kubernetes metric used for instance availability monitoring. If the metric is not available, use the similar metric as exposed by the monitoring system.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-640 (Cont.) OcnrfMemoryUsageCrossedMinorThreshold

Field	Details
Recommended Actions	The alert gets cleared when the memory utilization falls below the minor threshold or crosses the major threshold, in which case OcnrfMemoryUsageCrossedMajorThreshold alert is raised. **Note**: The threshold is configurable in the alerts file. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Available in OCI	Yes

6.3.4.2.2 OcnrfMemoryUsageCrossedMajorThreshold

Table 6-641 OcnrfMemoryUsageCrossedMajorThreshold

Field	Details
Description	'OCNRF Memory Usage for pod <*Pod name*> has crossed the major threshold (60%) (value = {{ $value }}) of its limit.'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Memory Usage of pod exceeded 60% of its limit.'
Severity	Major
Condition	A pod has reached the configured major threshold (60%) of its memory resource limits.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7031
Metric Used	'container_memory_usage_bytes' and 'container_spec_memory_limit_bytes' **Note**: This is a Kubernetes metric used for instance availability monitoring. If the metric is not available, use the similar metric as exposed by the monitoring system.
Recommended Actions	The alert gets cleared when the memory utilization falls below the major threshold or crosses the critical threshold, in which case OcnrfMemoryUsageCrossedCriticalThreshold alert is raised. **Note**: The threshold is configurable in the alert file. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Available in OCI	Yes

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.2.3 OcnrfMemoryUsageCrossedCriticalThreshold

Table 6-642 OcnrfMemoryUsageCrossedCriticalThreshold

Field	Details
Description	'OCNRF Memory Usage for pod <*Pod name*> has crossed the configured critical threshold (70%) (value = {{ $value }}) of its limit.'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Memory Usage of pod exceeded 70% of its limit.'
Severity	Critical
Condition	A pod has reached the configured critical threshold (70%) of its memory resource limits.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7032
Metric Used	'container_memory_usage_bytes' and 'container_spec_memory_limit_bytes' **Note**: This is a Kubernetes metric used for instance availability monitoring. If the metric is not available, use a similar metric as exposed by the monitoring system.
Recommended Actions	The alert gets cleared when the memory utilization falls below the critical threshold. **Note**: The threshold is configurable in the alert file. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Available in OCI	Yes

6.3.4.2.4 OcnrfOverloadThresholdBreachedL1

Table 6-643 OcnrfOverloadThresholdBreachedL1

Field	Details
Description	'Overload Level of {{$labels.app_kubernetes_io_name}} service is L1'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: Overload Level of {{$labels.app_kubernetes_io_name}} service is L1'
Severity	Warning
Condition	NRF Services have breached its configured threshold of Level L1 for any of the aforementioned metrics. Thresholds are configured for CPU, svc_failure_count, svc_pending_count, and memory.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7059
Metric Used	load_level

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-643 (Cont.) OcnrfOverloadThresholdBreachedL1

Field	Details
Recommended Actions	The alert is cleared when the Ingress Traffic rate falls below the configured L1 threshold. **Note**: The thresholds can be configured using REST API. **Steps**: 1. Reassess the reasons leading to NRF receiving additional traffic. 2. Refer to alert to determine which service is receiving high traffic. It may be due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. 3. Check the service pod logs on Kibana to determine the reason for the errors. 4. If this is expected traffic, then the thresholds levels may be reevaluated as per the call rate and reconfigured as mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*. 5. If this is the unexpected traffic, contact My Oracle Support.
Available in OCI	Yes

6.3.4.2.5 OcnrfOverloadThresholdBreachedL2

Table 6-644 OcnrfOverloadThresholdBreachedL2

Field	Details
Description	'Overload Level of {{$labels.app_kubernetes_io_name}} service is L2'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: Overload Level of {{$labels.app_kubernetes_io_name}} service is L2'
Severity	Warning
Condition	NRF Services have breached its configured threshold of Level L2 for any of the aforementioned metrics. Thresholds are configured for CPU, svc_failure_count, svc_pending_count, and memory.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7060
Metric Used	load_level
Recommended Actions	The alert is cleared when the Ingress Traffic rate falls below the configured L2 threshold. **Note**: The thresholds can be configured using REST API. **Steps**: 1. Reassess the reasons leading to NRF receiving additional traffic. 2. Refer to alert to determine which service is receiving high traffic. It may be due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. 3. Check the service pod logs on Kibana to determine the reason for the errors. 4. If this is expected traffic, then the thresholds levels may be reevaluated as per the call rate and reconfigured as mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*. 5. If this is the unexpected traffic, contact My Oracle Support.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-644 (Cont.) OcnrfOverloadThresholdBreachedL2

Field	Details
Available in OCI	Yes

6.3.4.2.6 OcnrfOverloadThresholdBreachedL3

Table 6-645 OcnrfOverloadThresholdBreachedL3

Field	Details
Description	'Overload Level of {{$labels.app_kubernetes_io_name}} service is L3'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: Overload Level of {{$labels.app_kubernetes_io_name}} service is L3'
Severity	Warning
Condition	NRF Services have breached its configured threshold of Level L3 for any of the aforementioned metrics. Thresholds are configured for CPU, svc_failure_count, svc_pending_count, and memory.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7061
Metric Used	load_level
Recommended Actions	The alert is cleared when the Ingress Traffic rate falls below the configured L3 threshold. **Note**: The thresholds can be configured using REST API. **Steps**: 1. Reassess the reasons leading to NRF receiving additional traffic. 2. Refer to alert to determine which service is receiving high traffic. It may be due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. 3. Check the service pod logs on Kibana to determine the reason for the errors. 4. If this is expected traffic, then the thresholds levels may be reevaluated as per the call rate and reconfigured as mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*. 5. If this is the unexpected traffic, contact My Oracle Support.
Available in OCI	Yes

6.3.4.2.7 OcnrfOverloadThresholdBreachedL4

Table 6-646 OcnrfOverloadThresholdBreachedL4

Field	Details
Description	'Overload Level of {{$labels.app_kubernetes_io_name}} service is L4'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: Overload Level of {{$labels.app_kubernetes_io_name}} service is L4'
Severity	Warning

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-646 (Cont.) OcnrfOverloadThresholdBreachedL4

Field	Details
Condition	NRF Services have breached its configured threshold of Level L4 for any of the aforementioned metrics. Thresholds are configured for CPU, svc_failure_count, svc_pending_count, and memory.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7062
Metric Used	load_level
Recommended Actions	The alert is cleared when the Ingress Traffic rate falls below the configured L4 threshold. **Note**: The thresholds can be configured using REST API. **Steps**: 1. Reassess the reasons leading to NRF receiving additional traffic. 2. Refer to alert to determine which service is receiving high traffic. It may be due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. 3. Check the service pod logs on Kibana to determine the reason for the errors. 4. If this is expected traffic, then the thresholds levels may be reevaluated as per the call rate and reconfigured as mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*. 5. If this is the unexpected traffic, contact My Oracle Support.
Available in OCI	Yes

6.3.4.3 DNS NAPTR Update Feature

This section lists the alerts that are specific to DNS NAPTR Update feature. For more information about the feature, see the DNS NAPTR Update section.

6.3.4.3.1 OcnrfDnsNaptrFailureResponseStatus

Table 6-647 OcnrfDnsNaptrFailureResponseStatus

Field	Details
Description	OCNRF DNS NAPTR Response status is not healthy
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, NfInstanceId: {{$labels.NfInstanceId}}, NfSetFqdn: {{$labels.NfSetFqdn}}, Replacement: {{$labels.Replacement}},timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} OCNRF Dns Naptr Response status is not healthy.'
Severity	Major
Condition	The DNS NAPTR response towards DNS Server is not successful.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7063
Metric Used	ocnrf_dns_naptr_failure_rx_response
Recommended Actions	This alert is cleared when DNS NAPTR response is successful either automatic through service operations, or manual trigger for update and delete NAPTR requests.

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.3.2 OcnrfAlternateRouteUpstreamDnsRetryExhausted

Table 6-648 OcnrfAlternateRouteUpstreamDnsRetryExhausted

Field	Details
Description	OCNRF alternate route upstream DNS retry exhausted
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, FQDNS_Name: {{$labels.FQDNS_Name}}, Replacement_Name: {{$labels.Replacement_Name}},timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} OCNRF alternate route upstream dns retry exhausted'
Severity	Major
Condition	The DNS NAPTR retry is exhausted.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7064
Metric Used	oc_alternate_route_upstream_dns_retry_exhausted
Recommended Actions	This alert is cleared automatically in 2 minutes.
Available in OCI	No

6.3.4.4 Notification Retry Feature

This section lists the alerts that are specific to Notification Retry feature. For more information about the feature, see the Notification Retry section.

6.3.4.4.1 OcnrfNotificationRetryExhausted

Table 6-649 OcnrfNotificationRetryExhausted

Field	Details
Description	'OCNRF NotificationRetry Exhausted'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, SubscriptionId: {{$labels.SubscriptionId}}, NotificationHostPort: {{$labels.NotificationHostPort}}'
Severity	Major
Condition	This alarm is raised when number of retries are exhausted.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7065
Metric Used	ocnrf_nfStatusNotify_rx_responses_total
Recommended Actions	The alert is cleared automatically after 5 minutes. **Steps**: Check logs in NF management pod to check the reason for retry query failures. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Available in OCI	Yes

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.4.2 OcnrfNotificationFailureOtherThanRetryExhausted

Table 6-650 OcnrfNotificationFailureOtherThanRetryExhausted

Field	Details
Description	'OCNRF notification failure other than retry exhausted'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, SubscriptionId: {{$labels.SubscriptionId}}, NotificationHostPort: {{$labels.NotificationHostPort}}, NumberOfRetriesAttempted: {{$labels.NumberOfRetriesAttempted}}'
Severity	Major
Condition	This alarm is raised when notification failure occurs with reason other than retry count exhausted.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7066
Metric Used	ocnrf_nfStatusNotify_rx_responses_total
Recommended Actions	The alert is cleared automatically after 5 minutes. Steps: Check logs in NF management pod to check the reason for retry query failures. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Available in OCI	Yes

6.3.4.5 NRF Message Feed Feature

This section lists the alerts that are specific to NRF Message Feed feature. For more information about the feature, see the NRF Message Feed section.

6.3.4.5.1 OcnrfIngressGatewayDDUnreachable

Table 6-651 OcnrfIngressGatewayDDUnreachable

Field	Details
Description	OCNRF Ingress Gateway Data Director unreachable
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} OCNRF Ingress Gateway Data Director unreachable'
Severity	Major
Condition	This alarm is raised when data director is not reachable from Ingress Gateway.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7067
Metric Used	oc_ingressgateway_dd_unreachable
Recommended Actions	Alert gets cleared automatically when the connection with data director is established.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.5.2 OcnrfEgressGatewayDDUnreachable

Table 6-652 OcnrfEgressGatewayDDUnreachable

Field	Details
Description	OCNRF Egress Gateway Data Director unreachable
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} OCNRF Egress Gateway Data Director unreachable'
Severity	Major
Condition	This alarm is raised when data director is not reachable from Egress Gateway.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7068
Metric Used	oc_egressgateway_dd_unreachable
Recommended Actions	Alert gets cleared automatically when the connection with data director is established.
Available in OCI	No

6.3.4.6 Subscription Limit Feature

This section lists the alerts that are specific to Subscription Limit feature. For more information about the feature, see the Subscription Limit section.

6.3.4.6.1 OcnrfSubscriptionGlobalCountWarnThresholdBreached

Table 6-653 OcnrfSubscriptionGlobalCountWarnThresholdBreached

Field	Details
Description	The total number of subscriptions has breached the configured WARN level threshold.
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}: The total number of subscriptions has breached the configured WARN level threshold'
Severity	Warning
Condition	This alarm is raised when the total number of subscriptions has breached the configured WARN level threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7069
Metric Used	ocnrf_nfset_limit_level
Recommended Actions	The alert is cleared automatically when the count comes down due to unsubscription. **Note**: The thresholds can be configured using REST API. **Steps**: 1. Reassess the reasons for new or renewal of subscription. 2. If this is expected subscription, then the subscription limit may be reevaluated as mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*. 3. If this is the unexpected subscription, contact My Oracle Support.
Available in OCI	Yes

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.6.2 OcnrfSubscriptionGlobalCountMinorThresholdBreached

Table 6-654 OcnrfSubscriptionGlobalCountMinorThresholdBreached

Field	Details
Description	The total number of subscriptions has breached the configured MINOR level threshold
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}: The total number of subscriptions has breached the configured MINOR level threshold'
Severity	Minor
Condition	This alarm is raised when the total number of subscriptions has breached the configured MINOR level threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7070
Metric Used	ocnrf_nfset_limit_level
Recommended Actions	The alert is cleared automatically when the count comes down due to unsubscription. **Note**: The thresholds can be configured using REST API. **Steps**: 1. Reassess the reasons for new or renewal of subscription. 2. If this is expected subscription, then the subscription limit may be reevaluated as mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*. 3. If this is the unexpected subscription, contact My Oracle Support.
Available in OCI	Yes

6.3.4.6.3 OcnrfSubscriptionGlobalCountMajorThresholdBreached

Table 6-655 OcnrfSubscriptionGlobalCountMajorThresholdBreached

Field	Details
Description	The total number of subscriptions has breached the configured MAJOR level threshold
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}: The total number of subscriptions has breached the configured MAJOR level threshold'
Severity	MAJOR
Condition	This alarm is raised when the total number of subscriptions has breached the configured MAJOR level threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7071
Metric Used	ocnrf_nfset_limit_level

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-655 (Cont.) OcnrfSubscriptionGlobalCountMajorThresholdBreached

Field	Details
Recommended Actions	The alert is cleared automatically when the count comes down due to unsubscription. **Note**: The thresholds can be configured using REST API. **Steps**: 1. Reassess the reasons for new or renewal of subscription. 2. If this is expected subscription, then the subscription limit may be reevaluated as mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*. 3. If this is the unexpected subscription, contact My Oracle Support.
Available in OCI	Yes

6.3.4.6.4 OcnrfSubscriptionGlobalCountCriticalThresholdBreached

Table 6-656 OcnrfSubscriptionGlobalCountCriticalThresholdBreached

Field	Details
Description	The total number of subscriptions has breached the configured CRITICAL level threshold
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}: The total number of subscriptions has breached the configured CRITICAL level threshold'
Severity	Critical
Condition	This alarm is raised when the total number of subscriptions has breached the configured CRITICAL level threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7072
Metric Used	ocnrf_nfset_limit_level
Recommended Actions	The alert is cleared automatically when the count comes down due to unsubscription. **Note**: The thresholds can be configured using REST API. **Steps**: 1. Reassess the reasons for new or renewal of subscription. 2. If this is expected subscription, then the subscription limit may be reevaluated as mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*. 3. If this is the unexpected subscription, contact My Oracle Support.
Available in OCI	Yes

6.3.4.6.5 OcnrfSubscriptionMigrationInProgressWarn

Table 6-657 OcnrfSubscriptionMigrationInProgressWarn

Field	Details
Description	The subscription migration is pending and subscriptionLimit feature is disabled

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-657 (Cont.) OcnrfSubscriptionMigrationInProgressWarn

Field	Details
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nrflevel: {{$labels.NrfLevel}}, subscriptionLimitFeatureStatus: {{$labels.subscriptionLimitFeatureStatus}}: The subscription migration is pending and subscriptionLimit feature is disabled'
Severity	Warning
Condition	The subscription migration is pending and subscriptionLimit feature is disabled.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7073
Metric Used	ocnrf_subscription_migration_status
Recommended Actions	This alert is cleared automatically when the migration is complete.
Available in OCI	No

6.3.4.6.6 OcnrfSubscriptionMigrationInProgressCritical

Table 6-658 OcnrfSubscriptionMigrationInProgressCritical

Field	Details
Description	The subscription migration is pending and subscriptionLimit feature is enabled
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nrflevel: {{$labels.NrfLevel}}, subscriptionLimitFeatureStatus: {{$labels.subscriptionLimitFeatureStatus}}: The subscription migration is pending and subscriptionLimit feature is enabled'
Severity	Warning
Condition	The subscription migration is pending and subscriptionLimit feature is enabled.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7074
Metric Used	ocnrf_subscription_migration_status
Recommended Actions	This alert is cleared automatically when the migration is complete. **Steps**: Disable the Subscription Limit feature. For more information, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
Available in OCI	No

6.3.4.7 Pod Protection Support for NRF Subscription Microservice

This section lists the alerts that are specific to Pod Protection Support for NRF Subscription Microservice feature. For more information about the feature, see the Pod Protection Support for NRF Subscription Microservice section.

6.3.4.7.1 OcnrfPodInDangerOfCongestionState

Table 6-659 OcnrfPodInDangerOfCongestionState

Field	Details
Description	'The pod {{$labels.kubernetes_pod_name}} of service {{$labels.app_kubernetes_io_name}} is in Danger of Congestion state'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-659 (Cont.) OcnrfPodInDangerOfCongestionState

Field	Details
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The pod is in Danger of Congestion state'
Severity	Major
Condition	A pod of a service is in Danger Of Congestion state. This could be due to CPU Usage or Pending Message Count above configured thresholds. This alert is raised when the Pod Protection feature is enabled for nfSubscription service. Currently this is applicable for NfSubscription service only.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7079
Metric Used	ocnrf_pod_congestion_state
Recommended Actions	The alert is cleared when the CPU or Pending Message Count goes below the configured thresholds for the Danger of Congested state. **Note**: The thresholds can be viewed using REST API. Reassess if the NRF is receiving additional traffic. If this is unexpected, contact My Oracle Support. **Steps**: 1. Refer to alert to determine which pod is receiving high traffic. It may due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. Check if NF is sending high number of updates, register or deregister. 2. Check for the corresponding congestion alert for CPU and Pending Message Count to understand the reason for pod congestion. 3. Check the service pod logs on Kibana to determine the reason for the errors. 4. If this is expected traffic, then the thresholds levels may need to be re-evaluated as per the call rate and reconfigured as mentioned in Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide.
Available in OCI	No

6.3.4.7.2 OcnrfPodPendingMessageCountInDangerOfCongestionState

Table 6-660 OcnrfPodPendingMessageCountInDangerOfCongestionState

Field	Details
Description	'The pod {{$labels.kubernetes_pod_name}} of service {{$labels.app_kubernetes_io_name}} is in Danger of Congestion state due to Pending Message Count above threshold'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The pod is in Danger of Congestion state due to Pending Message Count above threshold'
Severity	Major
Condition	A pod of a service is in Danger Of Congestion state due to its Pending Message Count above configured thresholds. Currently this is applicable for NfSubscription service only.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7081

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-660 (Cont.) OcnrfPodPendingMessageCountInDangerOfCongestionState

Field	Details
Metric Used	ocnrf_pod_pending_message_count_congestion_state
Recommended Actions	The alert is cleared when the pending message count goes below the configured thresholds for the Danger of Congested state. **Note**: The thresholds can be viewed using REST API. **Steps**: Reassess if the NRF is receiving additional traffic. If this is unexpected, contact My Oracle Support. 1. Refer to alert to determine which pod is receiving high traffic. It may due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. Check if NF is sending high number of updates, register, or deregister. 2. Check the service pod logs on Kibana to determine the reason for the errors. 3. If this is expected traffic, then the thresholds levels may need to be re-evaluated as per the call rate and reconfigured as mentioned in Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide.

6.3.4.7.3 OcnrfPodInCongestedState

Table 6-661 OcnrfPodInCongestedState

Field	Details
Description	'The pod {{$labels.kubernetes_pod_name}} of service {{$labels.app_kubernetes_io_name}} is in Congested state'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The pod is in Congested state'
Severity	Major
Condition	One or more pods of a service are in congested state. This could be due to CPU usage or Pending Message Count above configured thresholds. Currently this is applicable for NfSubscription service only.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7082
Metric Used	ocnrf_pod_congested_state

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-661 (Cont.) OcnrfPodInCongestedState

Field	Details
Recommended Actions	The alert is cleared when the CPU usage or Pending Message Count goes below the configured thresholds for the congested state. **Note**: The thresholds can be viewed using REST API. **Steps**: Reassess if the NRF is receiving additional traffic. If this is unexpected, contact My Oracle Support. 1. Refer to alert to determine which pod is receiving high traffic. It may due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. Check if NF is sending high number of updates, register, or deregister. 2. Check the service pod logs on Kibana to determine the reason for the errors. 3. If this is expected traffic, then the thresholds levels may need to be re-evaluated as per the call rate and reconfigured as mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
Available in OCI	No

6.3.4.7.4 OcnrfPodCpuUsageInCongestedState

Table 6-662 OcnrfPodCpuUsageInCongestedState

Field	Details
Description	'The pod {{$labels.kubernetes_pod_name}} of service {{$labels.app_kubernetes_io_name}} is in Congested state due to CPU usage above threshold'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The pod is in Congested state due to CPU usage above threshold'
Severity	Major
Condition	A pod of a service is in Congested state due to its CPU Usage above configured thresholds. Currently this is applicable for NfSubscription service only.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7083
Metric Used	ocnrf_pod_cpu_congestion_state

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-662 (Cont.) OcnrfPodCpuUsageInCongestedState

Field	Details
Recommended Actions	The alert is cleared when the CPU usage goes below the configured thresholds for the congested state. **Note**: The thresholds can be viewed using REST API. **Steps**: Reassess if the NRF is receiving additional traffic. If this is unexpected, contact My Oracle Support. 1. Refer to alert to determine which pod is receiving high traffic. It may due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. Check if NF is sending high number of updates, register or deregister. 2. Check the service pod logs on Kibana to determine the reason for the errors. 3. If this is expected traffic, then the thresholds levels may need to be re-evaluated as per the call rate and reconfigured as mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
Available in OCI	No

6.3.4.7.5 OcnrfPodCpuUsageInDangerOfCongestionState

Table 6-663 OcnrfPodCpuUsageInDangerOfCongestionState

Field	Details
Description	'The pod {{$labels.kubernetes_pod_name}} of service {{$labels.app_kubernetes_io_name}} is in Danger of Congestion state due to CPU usage above threshold'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The pod is in Danger of Congestion state due to CPU usage above threshold'
Severity	Major
Condition	A pod of a service is in Danger Of Congestion state due to its CPU above configured thresholds. This alert is raised when the Pod Pretoectoin feature is enabled for nfSubscription service. Currently this is applicable for NfSubscription service only.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7080
Metric Used	ocnrf_pod_cpu_congestion_state

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-663 (Cont.) OcnrfPodCpuUsageInDangerOfCongestionState

Field	Details
Recommended Actions	The alert is cleared when the CPU goes below the configured thresholds for the Danger of Congested state. **Note**: The thresholds can be viewed using REST API. **Steps**: Reassess if the NRF is receiving additional traffic. If this is unexpected, contact My Oracle Support. 1. Refer to alert to determine which pod is receiving high traffic. It may due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. Check if NF is sending sending high number of updates, register or deregister. 2. Check the service pod logs on Kibana to determine the reason for the errors. 3. If this is expected traffic, then the thresholds levels may need to be re-evaluated as per the call rate and reconfigured as mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
Available in OCI	No

6.3.4.7.6 OcnrfPodPendingMessageCountInCongestedState

Table 6-664 OcnrfPodPendingMessageCountInCongestedState

Field	Details
Description	'The pod {{$labels.kubernetes_pod_name}} of service {{$labels.app_kubernetes_io_name}} is in Congested state due to Pending Message Count above threshold'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The pod is in Congested state due to Pending Message Count above threshold'
Severity	Major
Condition	A pod of a service is in Congested state due to its Pending Message Count above configured thresholds. Currently this is applicable for NfSubscription service only.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7084
Metric Used	ocnrf_pod_pending_message_count_congestion_state

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-664 (Cont.) OcnrfPodPendingMessageCountInCongestedState

Field	Details
Recommended Actions	The alert is cleared when the pending message count goes below the configured thresholds for the congested state. **Note**: The thresholds can be viewed using REST API. **Steps**: Reassess if the NRF is receiving additional traffic. If this is unexpected, contact My Oracle Support. 1. Refer to alert to determine which pod is receiving high traffic. It may due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. Check if NF is sending high number of updates, register or deregister. 2. Check the service pod logs on Kibana to determine the reason for the errors. 3. If this is expected traffic, then the thresholds levels may need to be re-evaluated as per the call rate and reconfigured as mentioned in *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
Available in OCI	No

6.3.4.8 Controlled Shutdown of NRF Feature

This section lists the alerts that are specific to Controlled Shutdown of NRF feature. For more information about the feature, see the Controlled Shutdown of NRF section.

6.3.4.8.1 OcnrfOperationalStateCompleteShutdown

Table 6-665 OcnrfOperationalStateCompleteShutdown

Field	Details
Description	'The operational state of NRF is Complete Shutdown.'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The Operational state of NRF is Complete Shutdown'
Severity	Warning
Condition	The operator has changed the operational state of NRF to Complete Shutdown.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7085
Metric Used	ocnrf_operational_state
Recommended Actions	The alert is cleared when the user changes the operational state to NORMAL • If the alert is not cleared automatically after the operational state changes to NORMAL, collect the following: – all the logs as mentioned in the NrfConfiguration, Ingress Gateway, Egress Gateway, NrfAuditor microservices – the database dump from the site – REST output of operationalState, operationalStateHistory, and controlledShutdownOptions • Contact My Oracle Support.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.8.2 OcnrfAuditOperationsPaused

Table 6-666 OcnrfAuditOperationsPaused

Field	Details
Description	'The Audit procedures at NRF have been paused.'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The Audit procedures at NRF has been paused'
Severity	Warning
Condition	The NrfAuditor microservice has paused all audit procedures. This occurs during any of the following scenarios: 1. The NRF is in COMPLETE_SHUTDOWN operational state or just transitioned from COMPLETE_SHUTDOWN to a NORMAL operational state. 2. The database has been down for a prolonged period of time. To restore the database, see section "Database Corruption" in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*. 3. If the NrfAuditor pod has transitioned from READY to NOT_READY state.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7086
Metric Used	ocnrf_audit_status
Recommended Actions	The alert is expected to clear automatically, after the waiting period, and once all the above conditions are resolved. • If the alert is not cleared automatically, collect the following: – all the logs as mentioned in the NrfConfiguration microservice, and NrfAuditor pod logs, – the database dump from the site, – REST output of operationalState, operationalStateHistory, and controlled ShutdownOptions • Contact My Oracle Support.
Notes	NrfAuditor continues to remain in the paused state for some time, even after `OcnrfOperationalStateCompleteShutdown` alarm is cleared. For more information, see From CONTROLLED_SHUTDOWN to NORMAL subsection under *"Controlled Shutdown of NRF"* section in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.
Available in OCI	No

6.3.4.9 Monitoring the Availability of SCP Using SCP Health APIs Feature

This section lists the alerts that are specific to Monitoring the Availability of SCP Using SCP Health APIs feature. For more information about the feature, see the Monitoring the Availability of SCP Using SCP Health APIs section.

6.3.4.9.1 OcnrfAllSCPsMarkedAsUnavailable

Table 6-667 OcnrfAllSCPsMarkedAsUnavailable

Field	Details
Description	'All SCPs have been marked unavailable.'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-667 (Cont.) OcnrfAllSCPsMarkedAsUnavailable

Field	Details
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : All SCPs have been marked as unavailable'
Severity	Critical
Condition	All SCPs have been marked unavailable.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7088
Metric Used	'oc_egressgateway_peer_count and oc_egressgateway_peer_available_count'
Recommended Actions	NF clears the critical alarm when atleast 1 SCP peer in a peerset becomes available such that all other SCP peers in the given peerset are still unavailable.
Available in OCI	Yes

6.3.4.9.2 OcnrfSCPMarkedAsUnavailable

Table 6-668 OcnrfSCPMarkedAsUnavailable

Field	Details
Description	'An SCP has been marked unavailable.'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : One of the SCP has been marked unavailable'
Severity	Major
Condition	One of the SCPs has been marked unhealthy.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7087
Metric Used	oc_egressgateway_peer_health_status
Recommended Actions	This alert gets cleared when unavailable SCPs become available.
Available in OCI	Yes

6.3.4.10 CCA Header Validation in NRF for Access Token Service Operation Feature

This section lists the alerts that are specific to CCA Header Validation in NRF for Access Token Service Operation feature. For more information about the feature, see the CCA Header Validation in NRF for Access Token Service Operation section.

6.3.4.10.1 OcnrfCcaRootCertificateExpiringIn4Hours

Table 6-669 OcnrfCcaRootCertificateExpiringIn4Hours

Field	Details
Description	'The CCA Root Certificates expiring in 4 hours'.
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : CCA Root Certificate is expiring in 4 Hours'
Severity	Critical

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-669 (Cont.) OcnrfCcaRootCertificateExpiringIn4Hours

Field	Details
Condition	Indicates the expiry dates of the CCA Root certificates that are expiring in four hours.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7091
Metric Used	'oc_ingressgateway_cca_certificate_info'
Recommended Actions	The alert is cleared when the expiring CCA root certificates are replaced with new ones. **Steps**: Replace expiring certificate key pair with new ones. For more information on creating certificate key pair, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
Available in OCI	No

6.3.4.10.2 OcnrfCcaRootCertificateExpiringIn1Day

Table 6-670 OcnrfCcaRootCertificateExpiringIn1Day

Field	Details
Description	'The CCA Root Certificates expiring in 1 day'.
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : CCA Root Certificate is expiring in 1 Day'
Severity	Major
Condition	Indicates the expiry dates of the CCA Root certificates that are expiring in one day.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7090
Metric Used	'oc_ingressgateway_cca_certificate_info'
Recommended Actions	The alert is cleared when the expiring CCA root certificates are replaced with new ones. **Steps**: Replace expiring certificate key pair with new ones. For more information on creating certificate key pair, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
Available in OCI	No

6.3.4.10.3 OcnrfCcaRootCertificateExpiringIn5Days

Table 6-671 OcnrfCcaRootCertificateExpiringIn5Days

Field	Details
Description	'The CCA Root Certificates expiring in 5 days.'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : CCA Root Certificate is expiring in 5 Days'
Severity	Minor
Condition	Indicates the expiry dates of the CCA Root certificates that are expiring in five days.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-671 (Cont.) OcnrfCcaRootCertificateExpiringIn5Days

Field	Details
OID	1.3.6.1.4.1.323.5.3.36.1.2.7089
Metric Used	'oc_ingressgateway_cca_certificate_info'
Recommended Actions	The alert is cleared when the expiring CCA root certificates are replaced with new ones. **Steps**: Replace expiring certificate key pair with new ones. For more information on creating certificate key pair, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
Available in OCI	No

6.3.4.11 NRF Georedundancy Feature

This section lists the alerts that are specific to NRF Georedundancy feature. For more information about the feature, see the NRF Georedundancy section.

6.3.4.11.1 OcnrfDbReplicationStatusInactive

Table 6-672 OcnrfDbReplicationStatusInactive

Field	Details
Description	'The Database Replication Status is currently INACTIVE.'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nftype: {{$labels.NfType}}, nrflevel:{{$labels.NrfLevel}}, remoteNrfInstanceId: {{$labels.nrfInstanceId}}, remoteSiteName: {{$labels.siteName}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The database replication status is INACTIVE.'
Severity	Critical
Condition	The database replication channel status between the given site and the georedundant site(s) is inactive. The alert is raised per replication channel. The alarm is raised or cleared only if the georedundancy feature is enabled.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7013
Metric Used	'ocnrf_dbreplication_status'
Recommended Actions	The alert is cleared when the database channel replication status between the given site and the georedundant site(s) is up. For more information on how to check the database replication status, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
Notes	The alarm is included only if the georedundancy feature is enabled.
Available in OCI	No

6.3.4.11.2 OcnrfReplicationStatusMonitoringInactive

Table 6-673 OcnrfReplicationStatusMonitoringInactive

Field	Details
Description	'OCNRF Replication Status Monitoring Inactive'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-673 (Cont.) OcnrfReplicationStatusMonitoringInactive

Field	Details
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Pod {{ $labels.kubernetes_pod_name}} are not monitoring the replication status'
Severity	Critical
Condition	This alarm is raised when one or more pods are not monitoring the replication status.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7078
Metric Used	ocnrf_replication_status_monitoring_inactive
Recommended Actions	Resolution Steps: 1. Identify the pod for which the alert is raised. 2. Run the following command to restart the pod: `kubectl delete pod <pod_name> -n <namespace>`

6.3.4.12 XFCC Header Validation Feature

This section lists the alert that is specific to XFCC Header Validation feature. For more information about the feature, see the XFCC Header Validation section.

6.3.4.12.1 OcnrfNfAuthenticationFailureRequestsRejected

Table 6-674 OcnrfNfAuthenticationFailureRequestsRejected

Field	Details
Description	'Service request(s) received from NF have been rejected by OCNRF (current value is: {{ $value }})'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Request rejected for Nf FQDN based Authentication failure.'
Severity	Warning
Condition	NRF rejected a service request due to NF authentication failure
OID	1.3.6.1.4.1.323.5.3.36.1.2.7015
Metric Used	'ocnrf_nf_authentication_failure_total'
Recommended Actions	The alert is cleared automatically. **Steps**: Filter out nfAccessToken application ERROR logs on Kibana for more details.
Available in OCI	No

6.3.4.13 Enhanced NRF Set Based Deployment (NRF Growth) Feature

This section lists the alert that is specific to Enhanced NRF Set Based Deployment (NRF Growth) feature. For more information about the feature, see the Enhanced NRF Set Based Deployment (NRF Growth) section.

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.13.1 OcnrfRemoteSetNrfSyncFailed

Table 6-675 OcnrfRemoteSetNrfSyncFailed

Field	Details
Description	'A sync request to the NRF in the remote set has failed.' **Note**: The alert must be configured only if the NRF Growth feature is enabled.
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : A sync request to the NRF in the remote set has failed.'
Severity	Minor
Condition	Sync request to the NRF in the remote NRF set has failed.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7098
Metric Used	ocnrf_query_remote_cds_responses_total
Recommended Actions	The alert is cleared when the synchronization with the remote NRF set is successful. **Steps**: 1. Verify the remote NRF set is up. 2. Verify the connectivity between the local NRF set and remote NRF set. 3. Collect logs from local NRF and remote NRF(s). Contact My Oracle Support.
Available in OCI	No

6.3.4.13.2 OcnrfSyncFailureFromAllNrfsOfAnyRemoteSet

Table 6-676 OcnrfSyncFailureFromAllNrfsOfAnyRemoteSet

Field	Details
Description	'Sync requests to all the NRFs of a remote set has failed.' **Note**: The alert must be configured only if the NRF Growth feature is enabled.
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Sync requests to all the NRFs in any of the remote sets have failed'
Severity	Major
Condition	The sync requests to all the NRFs in the remote sets has failed.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7099
Metric Used	ocnrf_remote_set_unavailable_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-676 (Cont.) OcnrfSyncFailureFromAllNrfsOfAnyRemoteSet

Field	Details
Recommended Actions	The alert is cleared when synchronization is successful with at least one NRF of the remote NRF set. **Steps**: 1. Verify the remote NRF sets are up. 2. Verify the host details configured in the `nrfHostConfig` attribute using REST API. For more information about the attribute, see *Oracle Communications, Cloud Native Core Network Repository Function REST Specifications Guide*. 3. Verify the connectivity between the local NRF set and remote NRF set. 4. Collect logs from local NRF and remote NRF(s). Contact My Oracle Support.
Available in OCI	No

6.3.4.13.3 OcnrfSyncFailureFromAllNrfsOfAllRemoteSets

Table 6-677 OcnrfSyncFailureFromAllNrfsOfAllRemoteSets

Field	Details
Description	'Sync request to all the NRFs in all the remote sets have failed.' **Note**: The alert must be configured only if the NRF Growth feature is enabled.
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Sync request to all the NRFs in all the remote sets have failed'
Severity	critical
Condition	Sync requests to all the NRFs in all the remote sets have failed.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7100
Metric Used	ocnrf_all_remote_sets_unavailable_total
Recommended Actions	The alert is cleared when synchronization is successful with at least one NRF of the remote set(s). **Steps**: 1. Verify the remote NRF sets are up. 2. Verify the host details configured in the `nrfHostConfig` attribute using REST API. For more information about the attribute, see *Oracle Communications, Cloud Native Core Network Repository Function REST Specifications Guide*. 3. Verify the connectivity between the local NRF set and remote NRF set. 4. Collect logs from local NRF and remote NRF(s). Contact My Oracle Support.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.13.4 OcnrfCacheDataServiceDown

Table 6-678 OcnrfCacheDataServiceDown

Field	Details
Description	'OCNRF NrfCacheData service {{$labels.app_kubernetes_io_name}} is down'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Cache Data Service is down'
Severity	Critical
Condition	Cache Data Service is unavailable.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7101
Metric Used	up

Recommended Actions	The alert is cleared when the Cache Data Service (CDS) is available.
	Steps:
	1. To check the orchestration logs of the CDS and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the running state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the Running state.
	2. Refer to the application logs on Kibana and filter based on service names. Check for ERROR WARNING logs.
	3. Check the DB status. For more information on how to check the DB status, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*. Depending on the failure reason, take the resolution steps.
	4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.
	Note: Use the CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using the Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.

Available in OCI	No

6.3.4.13.5 OcnrfDatabaseFallbackUsed

Table 6-679 OcnrfDatabaseFallbackUsed

Field	Details
Description	'A service operation is unable to get data from the Cache Data Service, and hence gets the data from the cnDBTier to fulfill the service operation'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-679 (Cont.) OcnrfDatabaseFallbackUsed

Field	Details
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : A service Operation is unable to get data from the Cache Data Service, so falling back to DB'
Severity	Major
Condition	When a service operation is unable to get data from the Cache Data Service, and hence gets the data from the database to fulfill the service operation.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7102
Metric Used	ocnrf_db_fallback_total

Recommended Actions	The alert is cleared automatically.
	Steps:
	1. To check the orchestration logs of the CDS and check for liveness or readiness probe failures, do the following:
	a. Run the following command to check the pod status:
	`$ kubectl get po -n <namespace>`
	b. Run the following command to analyze the error condition of the pod that is not in the running state:
	`$ kubectl describe pod <pod name not in Running state> -n <namespace>`
	Where <pod name not in Running state> indicates the pod that is not in the Running state.
	2. Refer to the application logs on Kibana and filter based on service names. Check for ERROR WARNING logs.
	3. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.

Available in OCI	No

6.3.4.13.6 OcnrfTotalNFsRegisteredAtSegmentBelowMinorThreshold

Table 6-680 OcnrfTotalNFsRegisteredAtSegmentBelowMinorThreshold

Field	Details
Description	The alert is raised when the number of NFs registered at the segment is below the configured minor threshold.
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The number of NFs registered at the segment is below minor threshold'
Severity	Minor

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-680 (Cont.) OcnrfTotalNFsRegisteredAtSegmentBelowMinorThreshold

Field	Details
Condition	The number of NFs registered at the segment is below minor threshold. **Note**: This alert is triggered when the registered NF count is greater than or equal to 10 and below 20. This default value can be modified in the `ocnrf_alertrules_25.1.201.yaml` or `ocnrf_alertrules_promha_25.1.201.yaml` file depending on Prometheus version.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7103
Metric Used	ocnrf_nf_registered_count
Recommended Actions	The alert is cleared when the number of registered NFs in the segment is above the minor threshold. **Steps**: 1. Check if there is traffic for requests other than registration (for example, discovery requests). This ensures that NRF FQDN is reachable from other NFs and Ingress Gateway is up and running in all NRF Sets. 2. Check if the Ingress Gateway pod is up and running in all NRF sets. `kubectl get po -n <namespace>` 3. Validate that the CDS synchronization with remote NRF sets is successful. Validate below alerts are not present in the system: a. OcnrfSyncFailureFromAllNrfsOfAnyRemoteSet b. OcnrfSyncFailureFromAllNrfsOfAllRemoteSets 4. Check for registration pod logs on Kibana for ERROR WARN logs. 5. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use the CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using the Data Collector tool, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.
Available in OCI	No

6.3.4.13.7 OcnrfTotalNFsRegisteredAtSegmentBelowMajorThreshold

Table 6-681 OcnrfTotalNFsRegisteredAtSegmentBelowMajorThreshold

Field	Details
Description	The alert is raised when the number of NFs registered at the segment is below the configured major threshold.
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The number of NFs registered at the segment is below major threshold
Severity	Major

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-681 (Cont.) OcnrfTotalNFsRegisteredAtSegmentBelowMajorThreshold

Field	Details
Condition	The number of NFs registered at the segment is below major threshold. **Note**: This alert is triggered when the registered NF count is greater than or equal to 2 and below 10. This default value can be modified in the `ocnrf_alertrules_25.1.201.yaml` or `ocnrf_alertrules_promha_25.1.201.yaml` file depending on Prometheus version.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7104
Metric Used	ocnrf_nf_registered_count
Recommended Actions	The alert is cleared when the number of registered NFs in the segment is above the major threshold. **Steps**: 1. Check if there is traffic for requests other than registration (for example, discovery requests). This ensures that NRF FQDN is reachable from other NFs and Ingress Gateway is up and running in all NRF sets. 2. Check if the Ingress Gateway pod is up and running in all NRF sets. `kubectl get po -n <namespace>` 3. Validate that the CDS synchronization with remote NRF sets is successful. Validate below alerts are not present in the system: a. OcnrfSyncFailureFromAllNrfsOfAnyRemoteSet b. OcnrfSyncFailureFromAllNrfsOfAllRemoteSets 4. Check for registration pod logs on Kibana for ERROR WARN logs. 5. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use the CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using the Data Collector tool, see *Oracle Communications Cloud Native Core, cnDBTier User Guide.*
Available in OCI	No

6.3.4.13.8 OcnrfTotalNFsRegisteredAtSegmentBelowCriticalThreshold

Table 6-682 OcnrfTotalNFsRegisteredAtSegmentBelowCriticalThreshold

Field	Details
Description	The alert is raised when the number of NFs registered at the segment is below the configured critical threshold.
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The number of NFs registered at the segment is below critical threshold'
Severity	Critical
Condition	The number of NFs registered at the segment is below critical threshold. **Note**: This alert is triggered when the registered NF count is below 2. This default value can be modified in the `ocnrf_alertrules_25.1.201.yaml` or `ocnrf_alertrules_promha_25.1.201.yaml` file depending on Prometheus version.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-682 (Cont.) OcnrfTotalNFsRegisteredAtSegmentBelowCriticalThreshold

Field	Details
OID	1.3.6.1.4.1.323.5.3.36.1.2.7105
Metric Used	ocnrf_nf_registered_count
Recommended Actions	The alert is cleared when the number of registered NFs in the segment is above the critical threshold. **Steps**: 1. Check if there is traffic for requests other than registration (for example, discovery requests). This ensures that NRF FQDN is reachable from other NFs and Ingress Gateway is up and running in all NRF sets. 2. Check if the Ingress Gateway pod is up and running in all NRF sets. `kubectl get po -n <namespace>` 3. Validate that the CDS synchronization with remote NRF sets is successful. Validate below alerts are not present in the system: a. OcnrfSyncFailureFromAllNrfsOfAnyRemoteSet b. OcnrfSyncFailureFromAllNrfsOfAllRemoteSets 4. Check for registration pod logs on Kibana for ERROR WARN logs. 5. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support. **Note**: Use the CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using the Data Collector tool, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.

6.3.4.14 Ingress Gateway Pod Protection Feature

This section lists the alerts that are specific to Ingress Gateway Pod Protection feature. For more information about the feature, see the Ingress Gateway Pod Protection section.

6.3.4.14.1 OcnrfIngressGatewayPodInDangerOfCongestionState

Table 6-683 OcnrfIngressGatewayPodInDangerOfCongestionState

Field	Details
Description	'The pod {{$labels.kubernetes_pod_name}} is in Danger of Congestion state'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The pod is in Danger of Congestion state'
Severity	Major
Condition	When Ingress Gateway pod is in Danger Of Congestion state.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7092
Metric Used	oc_ingressgateway_pod_congestion_state

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-683 (Cont.) OcnrfIngressGatewayPodInDangerOfCongestionState

Field	Details
Recommended Actions	The alert is cleared when the pod is out of Danger Of Congestion (DoC) state. **Note**: The thresholds can be viewed using REST API. **Steps**: Reassess if the NRF is receiving additional traffic. If this is unexpected, contact My Oracle Support. 1. Refer to alert to determine which pod is receiving high traffic. It may due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. Check if NF is sending high number of updates, register, or deregister. 2. Check the service pod logs on Kibana to determine the reason for the errors. 3. If this is expected traffic, then the thresholds levels may need to be re-evaluated as per the call rate and reconfigured as mentioned in Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide.
Available in OCI	No

6.3.4.14.2 OcnrfIngressGatewayPodInCongestedState

Table 6-684 OcnrfIngressGatewayPodInCongestedState

Field	Details
Description	'The pod {{$labels.kubernetes_pod_name}} is in Congested state'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The pod is in Congested state'
Severity	Critical
Condition	When Ingress Gateway pod is in Congested state.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7093
Metric Used	oc_ingressgateway_pod_congestion_state
Recommended Actions	The alert is cleared when the pod is out of Congested state. **Note**: The thresholds can be viewed using REST API. **Steps**: Reassess if the NRF is receiving additional traffic. If this is unexpected, contact My Oracle Support. 1. Refer to alert to determine which pod is receiving high traffic. It may due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. Check if NF is sending high number of updates, register, or deregister. 2. Check the service pod logs on Kibana to determine the reason for the errors. 3. If this is expected traffic, then the thresholds levels may need to be re-evaluated as per the call rate and reconfigured as mentioned in Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.14.3 OcnrfIngressGatewayPodCpuUsageInCongestedState

Table 6-685 OcnrfIngressGatewayPodCpuUsageInCongestedState

Field	Details
Description	'The pod {{$labels.kubernetes_pod_name}} is in Congested state'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The pod is in Congested state'
Severity	Critical
Condition	Ingress Gateway pod is in Congested state due to CPU consumption above the configured thresholds.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7094
Metric Used	oc_ingressgateway_pod_resource_state
Recommended Actions	The alert is cleared when the CPU consumption goes below the configured thresholds for the Congested state. **Note**: The thresholds can be viewed using REST API. **Steps**: Reassess if the NRF is receiving additional traffic. If this is unexpected, contact My Oracle Support. 1. Refer to alert to determine which pod is receiving high traffic. It may due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. Check if NF is sending high number of updates, register, or deregister. 2. Check the service pod logs on Kibana to determine the reason for the errors. 3. If this is expected traffic, then the thresholds levels may need to be re-evaluated as per the call rate and reconfigured as mentioned in Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide.
Available in OCI	No

6.3.4.14.4 OcnrfIngressGatewayPodCpuUsageInDangerOfCongestionState

Table 6-686 OcnrfIngressGatewayPodCpuUsageInDangerOfCongestionState

Field	Details
Description	'The pod {{$labels.kubernetes_pod_name}} is in Danger of Congestion state due to CPU usage above threshold'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The pod is in Danger of Congestion state due to CPU usage above threshold'
Severity	Major
Condition	Ingress Gateway pod is in Danger of Congestion state due to CPU consumption above the configured thresholds.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7095
Metric Used	oc_ingressgateway_pod_resource_state

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-686 (Cont.) OcnrfIngressGatewayPodCpuUsageInDangerOfCongestionState

Field	Details
Recommended Actions	The alert is cleared when the CPU consumption is not as per the configured thresholds value for the Danger of Congestion state. **Note**: The thresholds can be viewed using REST API. **Steps**: Reassess if the NRF is receiving additional traffic. If this is unexpected, contact My Oracle Support. 1. Refer to alert to determine which pod is receiving high traffic. It may due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. Check if NF is sending high number of updates, register, or deregister. 2. Check the service pod logs on Kibana to determine the reason for the errors. 3. If this is expected traffic, then the thresholds levels may need to be re-evaluated as per the call rate and reconfigured as mentioned in Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide.
Available in OCI	No

6.3.4.14.5 OcnrfIngressGatewayPodPendingMessageInCongestedState

Table 6-687 OcnrfIngressGatewayPodPendingMessageInCongestedState

Field	Details
Description	'The pod {{$labels.kubernetes_pod_name}} is in Congested state'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The pod is in Congested state'
Severity	Critical
Condition	Ingress Gateway pod is in Congested state due to pending message count above the configured thresholds.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7096
Metric Used	oc_ingressgateway_pod_resource_state
Recommended Actions	The alert is cleared when the pending message count is not as per the configured thresholds value for the Congested state. **Note**: The thresholds can be viewed using REST API. **Steps**: Reassess if the NRF is receiving additional traffic. If this is unexpected, contact My Oracle Support. 1. Refer to alert to determine which pod is receiving high traffic. It may due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. Check if NF is sending high number of updates, register, or deregister. 2. Check the service pod logs on Kibana to determine the reason for the errors. 3. If this is expected traffic, then the thresholds levels may need to be re-evaluated as per the call rate and reconfigured as mentioned in Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.14.6 OcnrfIngressGatewayPodPendingMessageInDangerOfCongestionState

Table 6-688 OcnrfIngressGatewayPodPendingMessageInDangerOfCongestionState

Field	Details
Description	'The pod {{$labels.kubernetes_pod_name}} is in Danger of Congestion state due to Pending Message above threshold'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}},podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : The pod is in Danger of Congestion state due to Pending Message above threshold'
Severity	Major
Condition	Ingress Gateway pod is in Danger of Congestion state due to pending message count above the configured thresholds.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7097
Metric Used	oc_ingressgateway_pod_resource_state
Recommended Actions	The alert is cleared when the pending message count is not as per the configured thresholds value for the Danger of Congestion state. **Note**: The thresholds can be viewed using REST API. **Steps**: Reassess if the NRF is receiving additional traffic. If this is unexpected, contact My Oracle Support. 1. Refer to alert to determine which pod is receiving high traffic. It may due to a sudden spike in traffic. For example: When one mated site goes down, the NFs move to the given site. Check if NF is sending high number of updates, register, or deregister. 2. Check the service pod logs on Kibana to determine the reason for the errors. 3. If this is expected traffic, then the thresholds levels may need to be re-evaluated as per the call rate and reconfigured as mentioned in Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide.
Available in OCI	No

6.3.4.15 Subscriber Location Function Feature

This section lists the alert that is specific to Subscriber Location Function feature. For more information about the feature, see the Subscriber Location Function section.

6.3.4.15.1 OcnrfMaxSlfAttemptsExhausted

Table 6-689 OcnrfMaxSlfAttemptsExhausted

Field	Details
Description	'NF discovery request with fqdn {{$labels.NfProfileFqdn}} NF type {{$labels.NfType}} has exhausted maximum SLF attempts'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, nrflevel: {{$labels.NrfLevel}}, podname: {{$labels.kubernetes_pod_name}}, NfProfileFqdn: {{$labels.NfProfileFqdn}}, NfType: {{$labels.NfType}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The maximum slf attempts have exhausted.'

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-689 (Cont.) OcnrfMaxSlfAttemptsExhausted

Field	Details
Severity	Critical
Condition	NF discovery request with FQDN of the given NFType UDR has exhausted maximum SLF attempts. This alert is raised when the `ocnrf_max_slf_attempts_exhausted_total` metric is pegged. **Note**: This alert is included if SLF selection from registered profiles is enabled.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7054
Metric Used	'ocnrf_max_slf_attempts_exhausted_total'
Recommended Actions	The alert is cleared automatically after 5 minutes. **Steps**: 1. Check logs in NF discovery pod to check the reason for SLF query failures. 2. In DISCOVERED_SLF_CONFIG_MODE, make sure that SLFs are registered with valid IPV4, PV6, or FQDN information. Verify the same in the `slfDiscoveredCandidateList` from the `slfOptions`. 3. In STATIC_SLF_CONFIG_MODE, verify if slfHostConfig details are configured correctly. **Note**: Use CNC NF Data Collector tool for capturing logs. For more information on how to collect logs using Data Collector tool, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Available in OCI	Yes

6.3.4.16 EmptyList in Discovery Response Feature

This section lists the alert that is specific to EmptyList in Discovery Response feature. For more information about the feature, see the EmptyList in Discovery Response section.

6.3.4.16.1 OcnrfNFDiscoveryEmptyListObservedNotification

Table 6-690 OcnrfNFDiscoveryEmptyListObservedNotification

Field	Details
Description	'Empty List observed with received discovery request with NfType $labels.NfType Feature Status $labels.FeatureStatus'
Summary	'namespace: $labels.namespace, nrflevel:$labels.NrfLevel, podname: $labels.pod, NfType: $labels.NfType, FeatureStatus: $labels.FeatureStatus: Empty List observed with received discovery request'
Severity	Critical
Condition	This alarm is raised when profiles do not match the discovery request. Also, this alarm is raised when the SUSPENDED profile is in response to incoming request and Empty List feature is enabled.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7055
Metric Used	ocnrf_nfDiscover_emptyList_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-690 (Cont.) OcnrfNFDiscoveryEmptyListObservedNotification

Field	Details
Recommended Actions	The alert is cleared automatically after a duration of 5 minutes. **Steps**: 1. Collect the logs. 2. Check logs for the following conditions: • Verify if the NF has sent Empty List in response in NRF Discovery. • Check if NF has sent SUSPENDED profiles in response for incoming requests when EmptyList feature is ENABLED. • If the response is not Empty List or does not contain SUSPENDED profiles. 3. If the alert still persists, contact My Oracle Support. **Note**: Use CNC NF Data Collector tool for capturing logs. For more details, see *Oracle Communications Cloud Native Core, Network Function Data Collector User Guide*.
Available in OCI	No

6.3.4.17 Support for TLS

This section lists the alert that is specific to Support for TLS feature. For more information about the feature, see the Support for TLS section.

6.3.4.17.1 OcnrfTLSCertificateExpireMinor

Table 6-691 OcnrfTLSCertificateExpireMinor

Field	Details
Description	'TLS certificate to expire in 6 months'.
Summary	'namespace: {{$labels.namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }} {{ end }} : TLS certificate to expire in 6 months'
Severity	Minor
Condition	This alert is raised when the TLS certificate is about to expire in six months.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7106
Metric Used	security_cert_x509_expiration_seconds
Recommended Actions	The alert is cleared when the TLS certificate is renewed. For more information about certificate renewal, see "Creating Private Keys and Certificate " section in the *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.17.2 OcnrfTLSCertificateExpireMajor

Table 6-692 OcnrfTLSCertificateExpireMajor

Field	Details
Description	'TLS certificate to expire in 3 months.'
Summary	'namespace: {{$labels.namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }} {{ end }} : TLS certificate to expire in 3 months'
Severity	Major
Condition	This alert is raised when the TLS certificate is about to expire in three months.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7107
Metric Used	security_cert_x509_expiration_seconds
Recommended Actions	The alert is cleared when the TLS certificate is renewed. For more information about certificate renewal, see "Creating Private Keys and Certificate" section in the *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
Available in OCI	No

6.3.4.17.3 OcnrfTLSCertificateExpireCritical

Table 6-693 OcnrfTLSCertificateExpireCritical

Field	Details
Description	'TLS certificate to expire in one month.'
Summary	'namespace: {{$labels.namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }} {{ end }} : TLS certificate to expire in 1 month'
Severity	Critical
Condition	This alert is raised when the TLS certificate is about to expire in one month.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7108
Metric Used	security_cert_x509_expiration_seconds
Recommended Actions	The alert is cleared when the TLS certificate is renewed. For more information about certificate renewal, see "Creating Private Keys and Certificate" section in the *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.18 Egress Gateway Pod Throttling

6.3.4.18.1 OcnrfEgressPerPodDiscardRateAboveMajorThreshold

Table 6-694 OcnrfEgressPerPodDiscardRateAboveMajorThreshold

Field	Details
Description	'Egressgateway PerPod Discard Rate is greater than the configured major threshold. (current value is: {{ $value }})'
Summary	'kubernetes_namespace: {{$labels.namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: Egressgateway PerPod Discard Rate is more than 1 request per second.'
Severity	Major
Condition	This alert is raised when the Egress Gateway pods discard traffic due to its request limit is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7113
Metric Used	oc_egressgateway_podlevel_throttling_discarded_total
Recommended Actions	The alert is cleared when the Egress Gateway pods discard traffic rate falls below the major threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. Steps: 1. Refer Egress Gateway section in Grafana to determine which service is sending high traffic. 2. Refer Egress Gateway section in Grafana to determine the increase in 4xx and 5xx error codes. 3. Check Egress Gateway logs on Kibana to determine the reason for the errors.
Available in OCI	No

6.3.4.18.2 OcnrfEgressPerPodDiscardRateAboveCriticalThreshold

Table 6-695 OcnrfEgressPerPodDiscardRateAboveCriticalThreshold

Field	Details
Description	'Egressgateway PerPod Discard Rate is greater than the configured critical threshold. (current value is: {{ $value }})'
Summary	'kubernetes_namespace: {{$labels.namespace}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}: Egressgateway PerPod Discard Rate is more than 100 requests per second.'
Severity	Critical
Condition	This alert is raised when the Egress Gateway pods discard traffic due to its request limit is greater than the configured threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7114
Metric Used	oc_egressgateway_podlevel_throttling_discarded_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-695 (Cont.) OcnrfEgressPerPodDiscardRateAboveCriticalThreshold

Field	Details
Recommended Actions	The alert is cleared when the Egress Gateway pods discard traffic rate falls below the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. Steps: 1. Refer Egress Gateway section in Grafana to determine which service is sending high traffic. 2. Refer Egress Gateway section in Grafana to determine the increase in 4xx and 5xx error codes. 3. Check Egress Gateway logs on Kibana to determine the reason for the errors.
Available in OCI	No

6.3.4.19 Ingress Gateway Pod Protection Using Rate Limiting

6.3.4.19.1 OcnrfIngressDiscardDueToRateLimitMajorThreshold

Table 6-696 OcnrfIngressDiscardDueToRateLimitMajorThreshold

Field	Details
Description	'Ingress Gateway discards due to rate limit exceeds the configured major threshold. (current value is: {{ $value }})'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Ingressgateway Discard due to Rate Limit is more than or equal to 1 requests per second and less than 100 requests per second.'
Severity	Major
Condition	This alert is raised when Ingress Gateway discard requests as rate limit exceeds the configured major threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7129
Metric Used	oc_ingressgateway_http_request_ratelimit_denied_count_total
Recommended Actions	The alert is cleared when the Ingress Gateway pods discard traffic rate falls below the major threshold or exceeds the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. **Steps**: 1. Refer Ingress Gateway Pod Protection By Rate Limit section in Grafana to determine which pods are overloaded and the rate of traffic received. 2. Refer Ingress Gateway section in Grafana to determine the increase in 4xx and 5xx error codes. 3. Refer the Grafana dashboard to determine which service traffic is above expectation.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.4.19.2 OcnrfIngressDiscardDueToRateLimitCriticalThreshold

Table 6-697 OcnrfIngressDiscardDueToRateLimitCriticalThreshold

Field	Details
Description	'Ingress gateway discards due to rate limit exceeds the configured critical threshold. (current value is: {{ $value }})'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Ingressgateway Discard due to Rate Limit is more than or equal to 100 requests per second.'
Severity	Critical
Condition	This alert is raised when Ingress Gateway discard requests as rate limit exceeds the configured critical threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7130
Metric Used	oc_ingressgateway_http_request_ratelimit_denied_count_total
Recommended Actions	The alert is cleared when the Ingress Gateway pods discard traffic rate falls below the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. **Steps:** 1. Refer Ingress Gateway Pod Protection By Rate Limit section in Grafana to determine which pods are overloaded and the rate of traffic received. 2. Refer Ingress Gateway section in Grafana to determine the increase in 4xx and 5xx error codes. 3. Refer the Grafana dashboard to determine which service traffic is above expectation.
Available in OCI	No

6.3.4.20 Egress Gateway Pod Protection Using Rate Limiting

6.3.4.20.1 OcnrfEgressDiscardDueToRateLimitMajorThreshold

Table 6-698 OcnrfEgressDiscardDueToRateLimitMajorThreshold

Field	Details
Description	'Egress Gateway discards due to rate limit exceeds the configured major threshold. (current value is: {{ $value }})'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Egressgateway Discard due to Rate Limit is more than or equal to 1 requests per second and less than 100 requests per second.'
Severity	Major
Condition	This alert is raised when Egress Gateway discard requests as rate limit exceeds the configured major threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7131
Metric Used	oc_egressgateway_http_request_ratelimit_denied_count_total

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table 6-698 (Cont.) OcnrfEgressDiscardDueToRateLimitMajorThreshold

Field	Details
Recommended Actions	The alert is cleared when the Egress Gateway pods discard traffic rate falls below the major threshold or exceeds the critical threshold. **Note**: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. **Steps**: 1. Refer Egress Gateway Pod Protection By Rate Limit section in Grafana to determine which pods are overloaded and the rate of traffic received. 2. Refer Egress Gateway section in Grafana to determine the increase in 4xx and 5xx error codes. 3. Refer the Grafana dashboard to determine which service traffic is above expectation.
Available in OCI	No

6.3.4.20.2 OcnrfEgressDiscardDueToRateLimitCriticalThreshold

Table 6-699 OcnrfEgressDiscardDueToRateLimitCriticalThreshold

Field	Details
Description	'Egress gateway discards due to rate limit exceeds the configured critical threshold. (current value is: {{ $value }})'
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Egressgateway Discard due to Rate Limit is more than or equal to 100 requests per second.'
Severity	Critical
Condition	This alert is raised when Egress Gateway discard requests as rate limit exceeds the configured critical threshold.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7132
Metric Used	oc_egressgateway_http_request_ratelimit_denied_count_total
Recommended Actions	The alert is cleared when the Egress Gateway pods discard traffic rate falls below the critical threshold. Note: The threshold is configurable in the alert file. Reassess why the NRF is receiving additional traffic (for example, Mated site NRF is unavailable in georedundancy scenario). If this alert is unexpected, contact My Oracle Support. Steps: 1. Refer Egress Gateway Pod Protection By Rate Limit section in Grafana to determine which pods are overloaded and the rate of traffic received. 2. Refer Egress Gateway section in Grafana to determine the increase in 4xx and 5xx error codes. 3. Refer the Grafana dashboard to determine which service traffic is above expectation.
Available in OCI	No

Copyright © 2019, 2025, Oracle and/or its affiliates.

6.3.5 NRF Alert Configuration

This section describes the measurement based alert rules configuration for NRF. The Alert Manager uses the Prometheus measurements values as reported by microservices in conditions under alert rules to trigger alerts.

> **Note**
>
> - Alert file is packaged with NRF custom templates. The NRF CSAR package can be downloaded from MOS. Unzip the NRF CSAR package file to get the Alertrules.yaml file.
> - Review the Alertrules.yaml file and edit the value of the parameters in the Alertrules.yaml file (if needed to be changed from default values) before configuring the alerts. See below table for details.
> - kubernetes_namespace is configured as Kubernetes namespace in which NRF is deployed. Default value is NRF. Update the Alertrules.yaml file to reflect the correct NRF Kubernetes namespace.

Alert details which can be updated in Alertrules.yaml file before configuration

Table 6-700 Alerts

Alert Name	Details	Default Value	Notes
OcnrfTotalIngressTrafficRateAboveMinorThreshold	Traffic Rate is above 80 Percent of Max requests per second	Greater than/equal to 800 and Less than 900	Maximum Ingress rate considered is 1000 requests per second. So, here in default value 800 is 80% of 1000 and 900 is 90% of 1000. For example, if value need to be updated then depending upon maximum ingress request rate, set [90% of Max Ingress Request Rate] and [80% of Max Ingress Request Rate] for this alert
OcnrfTotalIngressTrafficRateAboveMajorThreshold	Traffic Rate is above 90 Percent of Max requests per second	Greater than/equal to 900 and Less than 950	Maximum Ingress rate considered is 1000 requests per second. So, here in default value 900 is 90% of 1000 and 950 is 95% of 1000. For example, if value need to be updated then depending upon maximum ingress request rate, set [90% of Max Ingress Request Rate] and [95% of Max Ingress Request Rate] for this alert
OcnrfTotalIngressTrafficRateAboveCriticalThreshold	Traffic Rate is above 95 Percent of Max requests per second	Greater than/equal to 950	Maximum Ingress rate considered is 1000 requests per second. So, here in default value 950 is 95% of 1000. For example, if value need to be updated then depending upon maximum ingress request rate, set [95% of Max Ingress Request Rate] for this alert

NRF Alert configuration in Prometheus

Copyright © 2019, 2025, Oracle and/or its affiliates.

Update NRF alerts for CNE 1.8.x and previous versions

The following procedure describes how to update the NRF alerts for CNE version 1.8.x and previous versions:

**NAME** :- Helm Release of Prometheus

**Namespace** :- Kubernetes NameSpace in which Prometheus is installed

1. Find the config map to configure alerts in Prometheus server using the following command:

   ```
   kubectl get configmap -n <Namespace>
   ```

   where, <Namespace> is the prometheus server namespace used in helm install command. For example, assuming Prometheus server is under occne-infra namespace, run the following command to find the config map:

   ```
   kubectl get configmaps -n occne-infra | grep prometheus-server
   ```

2. Take backup of current configuration map of Prometheus:

   ```
   kubectl get configmaps _NAME_-server -o yaml -n _Namespace_ > /tmp/
   tempConfig.yaml
   ```

3. Check if **alertsnrf** is present in the **tempConfig.yaml** file by running the following command:

   ```
   cat /tmp/tempConfig.yaml | grep alertsnrf
   ```

   Depending on the outcome of the previous step, perform anyone of the following steps:

   a. If **alertsnrf** is present, delete the **alertsnrf** entry from the **tempConfig.yaml** file, by running the following commands:

      ```
      sed -i '/etc\/config\/alertsnrf/d' /tmp/tempConfig.yaml
      ```

      ```
      sed -i '/rule_files:/a\  \- /etc/config/alertsnrf' /tmp/tempConfig.yaml
      ```

      > **Note**
      >
      > This command should be run only once.

   b. If **alertsnrf** is not present, add the **alertsnrf** entry in the **tempConfig.yaml** file by running the following command:

      ```
      sed -i '/rule_files:/a\  \- /etc/config/alertsnrf' /tmp/tempConfig.yaml
      ```

4. Update configuration map with updated file name of NRF alert file:

   ```
   kubectl replace configmap _NAME_-server -f /tmp/tempConfig.yaml
   ```


Copyright © 2019, 2025, Oracle and/or its affiliates.

5. Add NRF alert rules in configuration map under file name of NRF alert file:

```
kubectl patch configmap _NAME_-server -n _Namespace_--type merge --patch
"$(cat ~/NrfAlertrules.yaml)"
```

Update NRF alerts for CNE 1.9.x

This section describes the measurement based Alert rules configuration for NRF in Prometheus. Use the ocnrf_alerting_rules_promha_25.1.201.yaml file updated in NRF Alert configuration section.

1. Run the following command to apply the prometheusrules CRD:

```
$ kubectl apply -f ocnrf_alerting_rules_promha_25.1.201.yaml --namespace
<namespace>
```

Example:

```
$ kubectl apply -f ocnrf_alerting_rules_promha_25.1.201.yaml --namespace
ocnrf
  prometheusrule.monitoring.coreos.com/ocnrf-alerting-rules created
```

2. Run the following command to check NRF alert file is added to prometheusrules:

```
$ kubectl get prometheusrules --namespace <namespace>
```

Example:

```
$ kubectl get prometheusrules --namespace ocnrf
```

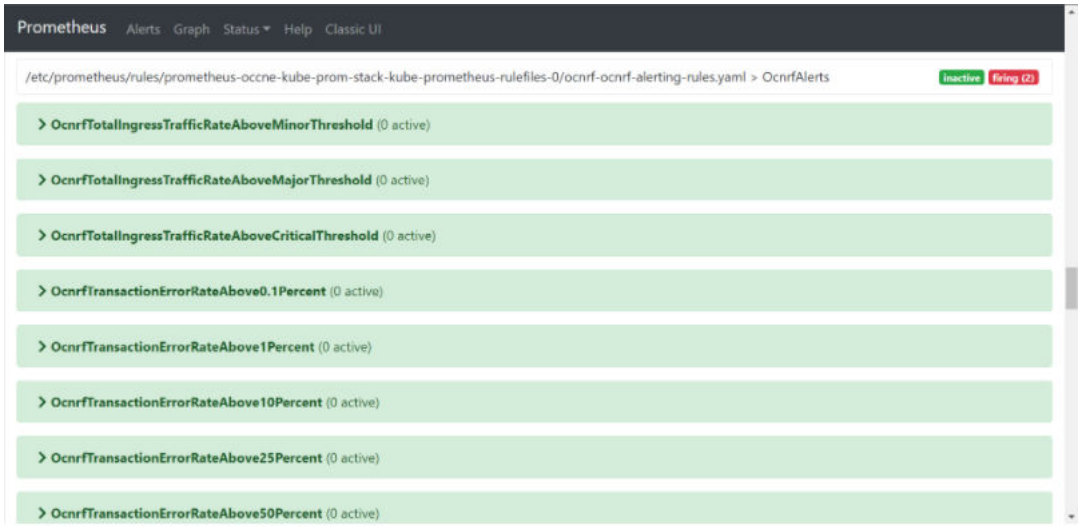
Sample output:

```
NAME                          AGE
ocnrf-alerting-rules          1m
```

3. Log in to Prometheus GUI and verify the alerts.
The following alert configuration file must be loaded as shown in the figure.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Figure 6-2 Prometheus Alert Manager



> **Note**
>
> The Prometheus server takes an updated configuration map that is automatically reloaded after approximately 60 seconds. Refresh the Prometheus GUI to confirm that the NRF Alerts have been reloaded.

Validating Alerts

After configuring the alerts in Prometheus server, a user can verify the same by following steps:

- Open the Prometheus server from your browser using the <IP>:<Port>
- Navigate to **Status** and then **Rules**.
- Search **Ocnrf**. OcnrfAlerts list will appear.

> **Note**
>
> If you are unable to see the alerts, it means the alert file is not loaded in a format that Prometheus server accepts. Modify the file and try again.

6.3.5.1 Disable Alerts

This section explains the procedure to disable the alerts in NRF.

1. Edit **NrfAlertrules-25.1.201.yaml** file to remove a specific alert.
2. Remove complete content of a specific alert from the **NrfAlertrules-25.1.201.yaml** file. For example: If you want to remove `OcnrfTrafficRateAboveMinorThreshold` alert, remove the complete content:

```
## ALERT SAMPLE START##

      - alert: OcnrfTrafficRateAboveMinorThreshold
        annotations:
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
          description: 'Ingress traffic Rate is above minor threshold i.e. 800 mps
(current value is: {{ $value }})'
          summary: 'Traffic Rate is above 80 Percent of Max requests per
second(1000)'
        expr:
sum(rate(oc_ingressgateway_http_requests_total{app_kubernetes_io_name="ingressgateway
",kubernetes_namespace="ocnrf"}[2m])) >= 800 < 900
        labels:
          severity: Minor
## ALERT SAMPLE END##
```

3. Perform Alert configuration. For more information about configuring alerts, see NRF Alert Configuration section.

6.3.5.2 Configuring SNMP Notifier

This section describes the procedure to configure SNMP Notifier.

Configure the IP and port of the SNMP trap receiver in the SNMP Notifier using the following procedure:

1. Run the following command to edit the deployment:

   ```
   $ kubectl edit deploy <snmp_notifier_deployment_name> -n <namespace>
   ```

   Example:

   ```
   $ kubectl edit deploy occne-snmp-notifier -n occne-infra
   ```

   SNMP deployment yaml file is displayed.

2. Edit the SNMP destination in the deployment yaml file as follows:

   ```
   --snmp.destination=<destination_ip>:<destination_port>
   ```

   Example:

   ```
   --snmp.destination=10.75.203.94:162
   ```

3. Save the file.

Checking SNMP Traps
Following is an example on how to capture the logs of the trap receiver server to view the generated SNMP traps:

```
$ docker logs <trapd_container_id>
```

Sample output:

```
2020-04-29 15:34:24 10.75.203.103 [UDP: [10.75.203.103]:2747-
>[172.17.0.4]:162]:DISMAN-EVENT-MIB::sysUpTimeInstance = Timeticks:
(158510800) 18 days, 8:18:28.00        SNMPv2-MIB::snmpTrapOID.0 = OID:
SNMPv2-SMI::enterprises.323.5.3.36.1.2.7003    SNMPv2-
SMI::enterprises.323.5.3.36.1.2.7003.1 = STRING:
"1.3.6.1.4.1.323.5.3.36.1.2.7003[]"  SNMPv2-
SMI::enterprises.323.5.3.36.1.2.7003.2 = STRING: "critical"      SNMPv2-
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
SMI::enterprises.323.5.3.36.1.2.7003.3 = STRING: "Status: critical- Alert:
OcnrfActiveSubscribersBelowCriticalThreshold  Summary: namespace: ocnrf,
nftype:5G_EIR, nrflevel:6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, podname: ocnrf-
nrfauditor-6b459f5db5-4kvt4,
        timestamp: 2020-04-29 15:33:24.408 +0000 UTC: Current number of
registered NFs detected below critical threshold.  Description: The number of
registered NFs detected below critical threshold (current value
          is: 0)
```

MIB Files for NRF

There are two MIB files which are used to generate the traps. The user need to update these files along with the Alert file in order to fetch the traps in their environment.

- ocnrf_mib_tc_25.1.201.mib
  This is considered as NRF top level mib file, where the objects and their data types are defined.
- ocnrf_mib_25.1.201.mib
  This file fetches the objects from the top level mib file and based on the alert notification, these objects can be selected for display.
- toplevel_25.1.201.mib: This defines the OIDs for all NFs.

> **Note**
>
> MIB files are packaged along with the release package. Download the file from MOS. For more information on downloading the release package, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

6.3.6 NRF Alert Configuration in OCI

The following procedure describes how to configure the NRF alerts for OCI. The OCI supports metric expressions written in MQL (Metric Query Language) and thus, requires a new NRF alert file for configuring alerts in OCI observability platform.

The following are the steps:

1. Run the following command to extract the .zip file:

   ```
   unzip ocnrf_oci_alertrules_<version>.zip
   ```

   The `ocnrf_oci` and `ocnrf_oci_resources` folders are available in the zip file.

   > **Note**
   >
   > The zip file is available in the Scripts folder of CSAR package.

2. Open the `ocnrf_oci` folder, in the `notifications.tf file`, update the parameter `endpoint` with the email id of the user.
3. Open the `ocnrf_oci_resources` folder, in the `notifications.tf file`, update the parameter `endpoint` with the email id of the user (replace test@gmail.com with the email id of the user).

Copyright © 2019, 2025, Oracle and/or its affiliates.

4. Log in to the OCI Console.

> **Note**
>
> For more details about logging in to the OCI, refer to Signing In to the OCI Console.

5. Open the navigation menu and select **Developer Services**. The **Developer Services** window appears in the right pane.
6. Under the **Developer Services**, select **Resource Manager**.
7. Under **Resource Manager**, select **Stacks**. The **Stacks** window appears.
8. Click **Create Stack**.
9. Select the default **My Configuration** radio button.
10. Under Stack configuration, select the folder radio button and upload the `ocnrf_oci` folder.
11. Enter the **Name** and **Description** and select the **compartment**.
12. Select the latest Terraform version from the **Terraform version** drop-down.
13. Click **Next**. The **Edit Stack** screen appears.
14. Enter the required inputs to create the NRF alerts or alarms and click **Save** and **Run Apply**.
15. Verify that the alarms are created in the Alarm Definitions screen (**OCI Console**> **Observability & Management**> **Monitoring**>**Alarm Definitions**) provided.
    The required inputs are:
    - **Alarms Configuration**
      - **Compartment Name** - Choose name of compartment from the drop-down
      - **Metric namespace** - Metric namespace that the user provided while deploying OCI Adaptors.
      - **Topic Name** - Any user configurable name. Must contain fewer than 256 characters. Only alphanumeric characters plus hyphens (-) and underscores (_) are allowed.
      - **Message Format** - Keep it as **ONS_OPTIMIZED**. (This is pre-populated)
      - **Alarm is_enabled** - Keep it as **True**. (This is pre-populated)
16. The steps 6 to 16 must be repeated for uploading the `ocnrf_oci_resources` folder. Keep **Metric namespace** as **mgmtagent_kubernetes_metrics** (This is pre-populated).

Copyright © 2019, 2025, Oracle and/or its affiliates.

A
HTTP Status Code Details

A.1 HTTP Status Code details

This section describes the HTTP status codes usage on SBI. The "`:status`" pseudo header field in HTTP/2 indicates the HTTP status codes. The status codes are defined in subclause 8.1.2.4 in IETF RFC 7540.

The following table specifies HTTP status codes per HTTP method that is supported on SBI. The support of HTTP status code is categorized as:

- mandatory, which is marked in table as "**M**". This means that all 3GPP NFs supports the processing of the specific HTTP status code for the specific HTTP method, when received in an HTTP response message. In such cases the 3GPP NF also supports the handling of the "ProblemDetails" JSON object with the content-type header field set to the value "`application/problem+json`" for HTTP status codes 4xx and 5xx, if the corresponding API definition in the related technical specification does not specify another response body for the corresponding status code.
- service specific, which is marked in table as "**SS**" and means that the requirement to process the HTTP status code depends on the definition of the specific API.
- not applicable, which is marked in table as "**N/A**". This means that the specific HTTP status code is not used for the specific HTTP method within the 3GPP NFs.

HTTP Status Code Supported on SBI

Table A-1 HTTP Status Codes

HTTP Status Code	HTTP Method				
-	**DELETE**	**GET**	**PATCH**	**POST**	**PUT**
100 Continue	N/A	N/A	N/A	N/A	N/A
200 OK (NOTE 1)	SS	M	SS	SS	SS
201 Created	N/A	N/A	N/A	SS	SS
202 Accepted	SS	N/A	SS	SS	SS
204 No Content (NOTE 2)	M	N/A	SS	SS	SS
300 Multiple Choices	N/A	N/A	N/A	N/A	N/A
303 See Other	SS	SS	N/A	SS	SS
307 Temporary Redirect	SS	SS	SS	SS	SS
308 Permanent Redirect	SS	SS	SS	SS	SS
400 Bad Request	M	M	M	M	M
401 Unauthorized	M	M	M	M	M
403 Forbidden	SS	SS	SS	SS	SS
404 Not Found	SS	SS	SS	SS	SS
405 Method Not Allowed	SS	SS	SS	SS	SS
406 Not Acceptable	N/A	N/A	N/A	N/A	N/A

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table A-1 (Cont.) HTTP Status Codes

HTTP Status Code	HTTP Method				
408 Request Timeout	SS	SS	SS	SS	SS
409 Conflict	N/A	N/A	SS	SS	SS
410 Gone	SS	SS	SS	SS	SS
411 Length Required	N/A	N/A	M	M	M
412 Precondition Failed	SS	SS	SS	SS	SS
413 Payload Too Large	N/A	N/A	M	M	M
414 URI Too Long	N/A	M	N/A	N/A	N/A
415 Unsupported Media Type	N/A	N/A	M	M	M
422 Unprocessable Entity	N/A	N/A	SS	N/A	N/A
500 Internal Server Error	M	M	M	M	M
501 Not Implemented	SS	SS	SS	SS	SS
503 Service Unavailable	M	M	M	M	M
504 Gateway Timeout	SS	SS	SS	SS	SS

> **Note**
>
> "200 OK" response used on SBI contains body.
> If the NF acting as an HTTP Client receives 2xx response code not appearing in table, the NF shall treat the received 2xx response:
>
> - as "204 No Content" if 2xx response does not contain body; and
> - as "200 OK" if 2xx response contains body.

NF as HTTP Client

An NF acting as an HTTP server generates HTTP status codes specified in the above table per indicated HTTP method.

An HTTP method which is not supported by 5GC SBI API specification is rejected with the HTTP status code "501 Not Implemented".

> **Note**
>
> In this case, the NF does not need to include in the HTTP response the "cause" attribute indicating corresponding error since the HTTP status code "501 Not Implemented" itself provides enough information of the error, that is, the NF does not recognize the HTTP method.

If the specified target resource does not exist, the NF rejects the HTTP method with the HTTP status code "404 Not Found".

If the NF supports the HTTP method but not by a target resource, the NF rejects the HTTP method with the HTTP status code "405 Method Not Allowed" and includes in the response an Allow header field containing the supported method(s) for that resource.

Copyright © 2019, 2025, Oracle and/or its affiliates.

> **Note**
>
> In this case, the NF does not need to include in the HTTP response the "cause" attribute indicating corresponding error since the HTTP status code "405 Method Not Allowed" itself provides enough information of the error and hence the Allow header field lists HTTP method(s) supported by the target resource.

If received HTTP request contains incorrect optional IE, the NF discards the incorrect IE.

If the NF supports the HTTP method by a target resource but the NF cannot successfully fulfill the received request, the following requirements applies:

- The NF as HTTP Server must map application errors to the most similar 3xx/4xx/5xx HTTP status code specified in Table A-2. If no such code is applicable, it should use "400 Bad Request" status code for errors caused by client side or "500 Server Internal Error" status code for errors caused on server side.
- If the received HTTP request contains an unsupported payload format, the NF rejects the HTTP request with the HTTP status code "415 Unsupported Media Type". If the HTTP PATCH method is rejected, the NF includes the Accept-Patch header field set to the value of supported patch document media types for a target resource. That is, either "application/merge-patch+json" if the NF supports "JSON Merge Patch" or "application/json-patch+json" if the NF supports "JSON Patch". If the HTTP PATCH request received contains "JSON Merge Patch" and "JSON Patch" documents, and the NF supports only one document, the NF ignores the unsupported patch document.

  > **Note**
  >
  > The format problem might be due to the request's indicated Content-Type or Content-Encoding header fields, or as a result of inspecting the payload body directly.

- If the received HTTP request contains payload body larger than the NF is able to process, the NF rejects the HTTP request with the HTTP status code "413 Payload Too Large".
- If the result of the received HTTP POST request used for resource creation is as same as the existing resource, the NF rejects the HTTP request with the HTTP status code "303 See Other" and includes a location header field set to the URI of the existing resource in the HTTP response.

HTTP Status Code support in NRF as Server

Table A-2 HTTP Status Code support in NRF as Server

HTTP Status Code	Service Operation											
-	NFRegister (PUT)	NFUpdate - Full Replacement (PUT)	NFUpdate Partial Replacement (PATCH)	NFUpdate Partial Replacement - Heartbeat(PATCH)	NFDeregister (DELETE)	NFListRetrieval (GET)	NFProfileRetrieval (GET)	NFStatusSubscribe (POST)	NFStatusSubscribe (PATCH)	NFStatusUnSubscribe (DELETE)	NFDiscover (GET)	AccessTokenRequest (POST)

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table A-2 (Cont.) HTTP Status Code support in NRF as Server

HTTP Status Code	Service Operation											
100 Continue	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
200 OK	N/A	Supported	Supported	Supported	N/A	Supported	Supported	N/A	Supported	N/A	Supported	Supported
201 Created	Supported	N/A	N/A	N/A	N/A	N/A	N/A	Supported	N/A	N/A	N/A	N/A
202 Accepted	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
204 No Content	N/A	N/A	N/A	Supported	Supported	N/A	N/A	N/A	N/A	Supported	N/A	N/A
300 Multiple Choices	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
303 See Other	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
307 Temporary Redirect	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported
308 Permanent Redirect	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported
400 Bad Request	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported
401 Unauthorized	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported
403 Forbidden	N/A	N/A	N/A	N/A	N/A	NotSupported	NotSupported	Supported	N/A	N/A	Supported	N/A
404 Not Found	N/A	N/A	Supported	Supported	Supported	Supported	Supported	N/A	Supported	Supported	N/A	N/A

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table A-2 (Cont.) HTTP Status Code support in NRF as Server

HTTP Status Code	Service Operation											
405 Method Not Allowed	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported
406 Not Acceptable	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
408 Request Timeout	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported	NotSupported
409 Conflict	N/A	N/A	Supported	Supported	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
410 Gone	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
411 Length Required	NotSupported	NotSupported	NotSupported	NotSupported	N/A	N/A	N/A	NotSupported	NotSupported	N/A	N/A	NotSupported
412 Precondition Failed	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
413 Payload Too Large	NotSupported	NotSupported	NotSupported	NotSupported	N/A	N/A	N/A	NotSupported	NotSupported	N/A	N/A	NotSupported
414 URI Too Long	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A	NotSupported	N/A
415 Unsupported Media Type	Supported	Supported	Supported	Supported	N/A	N/A	N/A	NotSupported	NotSupported	N/A	N/A	N/A
422 Unprocessable Entity	N/A	N/A	Supported	Supported	N/A	N/A	N/A	N/A	N/A	N/A	N/A	N/A
500 Internal Server Error	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported
501 Not Implemented	Supported	Supported	Supported	Supported	N/A	N/A	N/A	Supported	N/A	N/A	N/A	N/A

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table A-2 (Cont.) HTTP Status Code support in NRF as Server

HTTP Status Code	Service Operation											
503 Service Unavailable	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported
504 Gateway Timeout	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported	Supported

Copyright © 2019, 2025, Oracle and/or its affiliates.

B
Handling 3GPP Attributes

B.1 vsmf-support-ind

NRF supports `vsmf-support-ind` attribute in the NF Discover service operation query as per the 3GPP standards. For more information about the attribute, see *NRF Compliance Matrix*.

The discovery query is processed as follows:

- If `vsmf-support-ind` attribute is present in the NFDiscover query and the value of `targetNfType` is other than SMF, NRF does not reject the discovery query. In this case, NF profiles are not returned.
- When the `requester-feature` attribute is present or its value does not have any impact on the processing of `vsmf-support-ind` in the discovery query.
- When SMF profile contains `SMFInfoList` attribute and if one of the SMFInfo element in it doesn't have `vsmf-support-ind`, then NRF can consider such SMF profile as "V-SMF Capability Support of the SMF is not specified".

B.2 nrfSupportedFeatures

As per 3GPP TS 29.510, `nrfSupportedFeatures` attribute indicates the features supported by NRF for the NFDiscovery service. This attribute is included in the discovery response if at least one feature is supported by NRF. From release 23.4.0, the value of this attribute is 172 (in hexadecimal and 101110010 in binary), by reading the bits from left to right it indicates that it supports the 2, 5, 6, 7, and 9 Feature Number as per the following table:

Table B-1 nrfSupportedFeatures List

Feature Number	Feature	Details	nrfSupportedFeatures by NRF
1	Complex-Query	This is an optional parameter. Support of Complex Query expression (see clause 6.2.3.2.3.1)	NRF does not support this feature.
2	Query-Params-Ext1	This is an optional parameter. Support of the following query parameters: - limit - max-payload-size - required-features - pdu-session-types	NRF supports the following query parameters: • *limit* • *max-payload-size* • *required-features* • *pdu-session-types*

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table B-1 (Cont.) nrfSupportedFeatures List

Feature Number	Feature	Details	nrfSupportedFeatures by NRF
3	Query-Param-Analytics	This is an optional parameter. Support of the query parameters for Analytics identifier: - event-id-list - nwdaf-event-list	NRF does not support this feature.
4	MAPDU	This is an optional parameter. This feature indicates whether the NRF supports selection of UPF with Access Traffic Steering, Switching and Splitting (ATSSS) capability.	NRF does not support this feature.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table B-1 (Cont.) nrfSupportedFeatures List

Feature Number	Feature	Details	nrfSupportedFeatures by NRF
5	Query-Params-Ext2	This is an optional parameter. Support of the following query parameters: - requester-nf-instance-id - upf-ue-ip-addr-ind - pfd-data - target-snpn - af-ee-data - w-agf-info - tngf-info - twif-info - target-nf-set-id - target-nf-service-set-id - preferred-tai - nef-id - preferred-nf-instances - notification-type - serving-scope - internal-group-identity - preferred-api-versions - v2x-support-ind - redundant-gtpu - redundant-transport - lmf-id - an-node-type - rat-type - ipups - scp-domain-list - address-domain - ipv4-addr - ipv6-prefix - served-nf-set-id - remote-plmn-id - data-forwarding - preferred-full-plmn - requester-snpn-list - max-payload-size-ext	NRF supports the following query parameters: • *target-nf-set-id* • *preferred-tai* • *notification-type* • *serving-scope*
6	Service-Map	This is a mandatory parameter. This feature indicates whether it is supported to identify the list of NF Service Instances as a map (that is, the "nfServiceList" attribute of NFProfile is supported).	NRF supports the *nfServiceList* attribute in NFProfile.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table B-1 (Cont.) nrfSupportedFeatures List

Feature Number	Feature	Details	nrfSupportedFeatures by NRF
7	Query-Params-Ext3	This is an optional parameter. Support of the following query parameters: - ims-private-identity - ims-public-identity - msisdn - requester-plmn-specific-snssai-list - n1-msg-class - n2-info-class	NRF supports the following query parameters: • *n1-msg-class* • *n2-info-class*
8	Query-Params-Ext4	This is an optional parameter. Support of the following query parameters: - realm-id - storage-id	NRF does not support this feature.
9	Query-Param-vSmf-Capability	This is an optional parameter. Support of the query parameters for V-SMF capability: - vsmf-support-ind	NRF supports *vsmf-support-ind* query parameter for V-SMF capability.

The value of `nrfSupportedFeatures` attribute in the discovery response varies based on the value of the `requester-features` attribute sent by the Consumer NF. The `requester-features` attribute indicates the features that are supported by the Consumer NF.

The value of the `nrfSupportedFeatures` attribute is calculated based on the following scenarios:

- value of `nrfSupportedFeatures` attribute when there are common supported features between consumer NF and NRF
  If the `requester-features` attribute has the value of 12 (this is hexadecimal value) in the discovery search query, the corresponding binary value is 00010010. It indicates that the 2nd and 5th bit are enabled and Consumer NF supports the Feature Number 2 and 5. As per the current support at NRF, the common supported features between Consumer NF and NRF are 2 and 5. Hence, the value of `nrfSupportedFeatures` attribute in the discovery response is 12.
- value of `nrfSupportedFeatures` attribute when there are partial common supported features between consumer NF and NRF
  If the `requester-features` attribute has the value of 6 (this is hexadecimal value) in the discovery search query, the corresponding binary value is 0110. It indicates that the 2nd and 3rd bit are enabled and Consumer NF supports Feature Number 2 and 3. As per the current support at NRF, the common supported feature between Consumer NF and NRF is 2. Hence, the value of the `nrfSupportedFeatures` attribute in the discovery response is 2.
- value of `nrfSupportedFeatures` attribute when there is no common supported features between consumer NF and NRF
  If the `requester-features` attribute has the value of 8 (this is hexadecimal value) in the discovery search query, the corresponding binary value is 1000. It indicates that no common features are available between the Consumer NF and NRF. Hence, the value of the `nrfSupportedFeatures` attribute in the discovery response is 0.

Copyright © 2019, 2025, Oracle and/or its affiliates.

- value of `nrfSupportedFeatures` attribute is 172
  When the `requester-features` attribute is unavailable in the discovery search query of Consumer NF, the value of the `nrfSupportedFeatures` attribute in the discovery response is 172.

B.3 DataSetId Enhancements

As per 3GPP TS 29.510 v17.7, NRF supports the following enumeration (ENUM) values within DataSetId enumeration.

Table B-2 Additional DataSetId Enumerations

ENUM Value	Description
"A_PFD"	ApplicationData subset: Packet Flow Descriptions
"A_AFTI"	ApplicationData subset: AF Traffic Influence Data
"A_IPTV"	ApplicationData subset: IPTV Config Data
"A_BDT"	ApplicationData subset: Background Data Transfer
"A_SPD"	ApplicationData subset: Service Parameter Data
"A_EASD"	ApplicationData subset: EAS Deployment Information
"A_AMI"	ApplicationData subset: AM Influence Data
"P_UE"	PolicyData subset: UE Specific Data
"P_SCD"	PolicyData subset: Sponsored Connectivity Data
"P_BDT"	PolicyData subset: Background Data Transfer
"P_PLMNUE"	PolicyData subset: PLMN-specific UE policy data
"P_NSSCD"	PolicyData subset: Network Slice Specific Control Data
"P_MBSCD"	PolicyData subset: MBS Session Policy Control Data

NRF supports the additional ENUM values added in the `DataSetId` attribute in the following service operations as per the 3GPP TS 29.510 v17 standards:

- NFRegister
- NFProfileRetrieval
- NFUpdate (Partial/Complete)
- NFStatusNotify
- NFDiscover

If UDR registers with `DataSetId` attribute, only the following enums are allowed:

- SUBSCRIPTION
- POLICY
- EXPOSURE
- APPLICATION
- A_PFD
- A_AFTI
- A_IPTV
- A_BDT
- A_SPD

Copyright © 2019, 2025, Oracle and/or its affiliates.

- A_EASD
- A_AMI
- P_UE
- P_SCD
- P_BDT
- P_PLMNUE
- P_NSSCD
- P_MBSCD

However, in case of NFRegister service operation, the following validations are not supported:

- If UDR registers with `DataSetId` attribute with "***APPLICATION***", it is required to register with all ***A_****** enumerations in addition.
- If UDR registers with all the above defined ***A_****** enumerations, it should register with "***APPLICATION***" enumeration in addition.
- If UDR registers with `DataSetId` attribute with "***POLICY***", it is required to register with all ***P_****** enumerations in addition.
- If UDR registers with all the above defined ***P_****** enumerations, it should register with "***POLICY***" enumeration in addition.

For more information about the attribute, see *NRF Compliance Matrix*.

> **Note**
>
> The DataSetId parameter is ignored in the following scenarios for the NF Profiles (if the discovery query contains the DataSetId parameter):
>
> - supportedDataSets attribute is missing: If the UdrInfo profile does not contain the supportedDataSets attribute, the data-set parameter in the discovery request should be ignored.
> - UdrInfo JSON object is empty: If the UdrInfo JSON object is provided and it is empty, the data-set parameter should be ignored.
> - UdrInfo JSON object is missing: If the entire UdrInfo JSON object is not included in the request, the data-set parameter should be ignored.

B.4 Support for servingClientTypes in LmfInfo and GmlcInfo

As per 3GPP TS 29.510 v16.7, NRF supports `servingClientTypes` attribute in NF Profiles for Location Management Function (LMF) (`LmfInfo`) and Gateway Mobile Location Center (GMLC) (`GmlcInfo`). NRF supports `client-type` discovery query parameter to discover NFs based on `servingClientTypes` attribute. `servingclientTypes` attribute is supported for GMLC and LMF NF Types. If the above mentioned attribute is present in the NF Profile then the GMLC and LMF NFs are dedicated to serve the listed external client type. The listed external client types are mentioned in 3GPP TS 29.572 v16.7. If this attribute is not present in the NF Profile for the mentioned NFTypes then NFs are not dedicated to serve any external client type.

Possible values for `servingClientTypes` for both `LmfInfo` and `GmlcInfo` are as follows:

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table B-3 Possible Values for servingClientTypes

Enumeration value	Description
"EMERGENCY_SERVICES"	External client for emergency services
"VALUE_ADDED_SERVICES"	External client for value added services
"PLMN_OPERATOR_SERVICES"	External client for PLMN operator services
"LAWFUL_INTERCEPT_SERVICES"	External client for Lawful Intercept services
"PLMN_OPERATOR_BROADCAST_SERVICES"	External client for PLMN Operator Broadcast services
"PLMN_OPERATOR_OM"	External client for PLMN Operator O&M
"PLMN_OPERATOR_ANONYMOUS_STATISTICS"	External client for PLMN Operator anonymous statistics
"PLMN_OPERATOR_TARGET_MS_SERVICE_SUPPORT"	External client for PLMN Operator target MS service support

Note

If the `servingClientTypes` attribute is present in the respective `LmfInfo` or `GmlcInfo` attributes indicates NF are dedicated to serve the mentioned external client type. Hence absence of the respective `LmfInfo` or `GmlcInfo` attribute is also considered as NFs are not dedicated to serve any external client type.

If `servingClientTypes` attribute is present in `LmfInfo` or `GmlcInfo` attributes, it indicates that LMF or GMLC is dedicated to serve the listed external client type(s), For example, EMERGENCY_SERVICES.

If `servingClientTypes` attribute is not present in `LmfInfo` or `GmlcInfo` attributes or if `LmfInfo` and `GmlcInfo` attributes are absent, it indicates that the LMF or GMLC is not dedicated to serve the listed external client type.

For NFManagement service operations, NRF checks the `servingClientTypes` value matches with the supported values for `servingClientTypes` as mentioned in 3GPP TS 29.572 v16.7. If this value matches then NRF processes the requests. If the value does not match, NRF rejects the request.

For NFDiscover service operation, the `client-type` query parameter is received in the incoming NFDiscovery query. NRF checks the `client-type` value with the supported values for `servingClientTypes` as mentioned in 3GPP TS 29.572 v16.7.

- If the `servingClientTypes` value matches with the supported `servingClientTypes` values as mentioned in 3GPP TS 29.572 v16.7:
  - NRF checks the value of `client-type` query parameter with the values in the `servingClientTypes` list from the corresponding NfInfo of the registered NFProfiles.
    - If these values matches, that NFProfile is considered for the discovery response.
    - If the value does not match, NRF does not include that NFProfile in the response.
- If the `servingClientTypes` value is not matching with the supported values for `servingClientTypes` as mentioned in 3GPP TS 29.572 v16.7, NRF rejects the request.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Note

- In case, there are no NFProfiles of GMLC and LMF types registered in NRF with the matching `servingClientTypes`, it means none of the NF Profiles are dedicated to serve the `servingClientTypes`. Hence, NFProfiles which are not dedicated (where `servingClientTypes` attribute is absent or the respective NfInfo attribute is absent) are returned in the NFDiscovery response.
- In case, in any NRF deployment, `nfdiscovery.searchQueryIgnoreList` (Helm parameter) is configured with value client-type, `searchQueryIgnoreList` parameter shall not contain the `client-type`. So that NRF can process this attribute in the NFDiscover service operation.

Examples

- One instance of LMF is registered with NRF with `LMFInfo` present with `servingClientTypes` value as EMERGENCY_SERVICES and another instance of LMF is registered with NRF with `LmfInfo` present with `servingClientTypes` value as VALUE_ADDED_SERVICES.
  NFDiscover service operation query is received with`client-type` query parameter as EMERGENCY_SERVICES. In this scenario, first instance of LMF is returned by NRF in NFDiscover service operation response.
- One instance of LMF is registered with NRF with `LmfInfo` present with `servingClientTypes` value as EMERGENCY_SERVICES and another instance of LMF is registered with NRF with `LmfInfo` present with `servingClientTypes` value as VALUE_ADDED_SERVICES.
  NFDiscover service operation query is received with `client-type` query parameter as PLMN_OPERATOR_SERVICES. In this scenario, none of the LMF NF profiles are returned by NRF in NFDiscover service operation response.
- One instance of LMF is registered with NRF with `LmfInfo` present with `servingClientTypes` attribute is absent and another instance of LMF is registered with NRF with `LmfInfo` present with `servingClientTypes` value as VALUE_ADDED_SERVICES.
  NFDiscover service operation query is received with `client-type` query parameter as PLMN_OPERATOR_SERVICES. In this scenario, as there is no matching NFProfile with dedicated `servingClientTypes` value, hence NFProfile with non-dedicated `servingClientTypes`, that is the first instance of LMF is returned by NRF in NFDiscover service operation response.
- One instance of LMF is registered with NRF with `LmfInfo` present with `servingClientTypes` attribute is absent and another instance of LMF is registered with NRF with `LmfInfo` present with `servingClientTypes` value as PLMN_OPERATOR_SERVICES.
  NFDiscover service operation query is received with `client-type` query parameter as PLMN_OPERATOR_SERVICES. In this scenario, second instance of LMF is returned by NRF in NFDiscover service operation response.
- One instance of LMF is registered with NRF with `LmfInfo` is absent and another instance of LMF is registered with NRF with `LmfInfo` present with `servingClientTypes` value as VALUE_ADDED_SERVICES.
  NFDiscover service operation query is received with `client-type` query parameter as PLMN_OPERATOR_SERVICES. In this scenario, as there is no matching NFProfile with dedicated `servingClientTypes` value, hence NFProfile with non-dedicated

Copyright © 2019, 2025, Oracle and/or its affiliates.

servingClientTypes, that is the first instance of LMF is returned by NRF in NFDiscover service operation response.

- One instance of LMF is registered with NRFF with LmfInfo is absent and another instance of LMF is registered with NRF with LmfInfo present with servingClientTypes value as VALUE_ADDED_SERVICES.
  NFDiscover service operation query is received with client-type query parameter as VALUE_ADDED_SERVICES. In this scenario, second instance of LMF is returned by NRF in NFDiscover service operation response.

All these examples are also applicable to GMLC NF Type.

B.5 Enhancements for dnn NFProfile Attribute and Discovery Query Parameter

NRF supports dnn attribute that represents the Data Network Name (DNN). DNN comprises of the Network Identifier and Operator Identifier.

Until 24.2.x release, when dnn attribute was provided during NFManagement and NFDiscover service operations, NRF performed validations for null or empty values. NRF also supported exact matching of dnn query attribute value with the dnn attribute present in the registered NFProfile for NFDiscover service operations.

From 24.3.x releases, NRF performs the validations for the dnn attribute, as per 3GPP TS 29.571 v16.7. These validations are added for the NF Types SMF, UPF, PCF, and BSF (that is, smfInfo, smfInfoList, upfInfo, pcfInfo, and bsfInfo).

> **Note**
>
> The attributes for upfInfoList, pcfInfoList, and bsfInfoList are not supported by NRF.

- Maximum length of DNN is 100 Octets.
- DNN Structure = <Label>.<Label>.<Label>. Label should only contain (A-Z), (a-z), (0-9), '-'.
- Network Identifier structure is mandatory in DNN.
  Validations for Network Identifier:
  - maximum length can be 63 Octets
  - cannot start with {"rac","lac","sgsn","rnc"}
  - cannot end with {"gprs"}
  - cannot take value {"*"}
- Operator Identifier structure is optional in DNN.
  Validations for Operator Identifier:
  - The Access Point Name (APN) Operator Identifier is composed of three labels. The last label (= domain) shall be "gprs".
  - The first and second labels together identifies the PLMN.
    Format: "mnc<MNC>.mcc<MCC>.gprs", <MNC> of 3 digits , <MCC> of 3 digits.
    * The format should be strictly followed. Any change in the formatted value will result in considering the whole dnn value as Network Identifier.

Copyright © 2019, 2025, Oracle and/or its affiliates.

* Any other value of `mcc` and `mnc` comprising of anything else like alphabets, negative value, or hyphen(-) will be rejected.
  Consumer NF should send the MNC in 3 digits to NRF.

  Note: If there are only 2 significant digits in the MNC, one "0" digit must be added at the left side by Consumer NF to fill the 3 digits coding of MNC in the APN Operator Identifier.

  For example: If the APN Operator Identifier for MCC 345 and MNC 12, then Consumer NF sends MNC as 012 in the DNS "`mnc012.mcc345.gprs`" to NRF.

Examples:

- Following are the examples of valid `dnn` values:
  - `province1.mnc012.mcc345.gprs`
    Here, `province1` is Network Identifier and `mnc012.mcc345.gprs` is Operator Identifier.
  - `ggsn-cluster-A.provinceB.mnc012.mcc345.gprs`
    Here, `ggsn-cluster-A.provinceB` is Network Identifier and `mnc012.mcc345.gprs` is Operator Identifier.
- Following are the examples of invalid `dnn` values along with their rejection reasons:
  - `province_A, provinc*A.mnc012.mcc345.gprs`
    Here, these would be rejected as both the values contain a label having a character (_ and * respectively) that is not in permissible list of characters (A-Z), (a-z), (0-9), '-'.
  - `province1.gprs, province-A.gprs.mnc012.mcc345.gprs`
    Here, these would be rejected as all the values have their network identifier part ending with `gprs`.
  - `lac.province.mnc012.mcc345.gprs`, `race-province`, `rncabc.provinceA.mnc012.mcc345.gprs`, and `sgsn`
    Here, these would be rejected as all of the values have their network identifier starting with one of the following {"`rac`","`lac`","`sgsn`","`rnc`"}.

As per 3GPP TS 29.510 v16.7, the `dnn` attribute present in the discovery query is considered as a match with a `dnn` attribute present in the registered NFProfile during the following scenarios:

- Both the `dnn` attributes contain the same Network Identifier and Operator Identifier.
- Both the `dnn` attributes contain the same Network Identifier and none contains an Operator Identifier.
- The `dnn` attribute in the discovery query contains only the Network Identifier and the DNN value in the registered NF Profile contains both the Network Identifier and Operator Identifier, and both contain the same Network Identifier.
- The `dnn` attribute in the discovery query contains both the Network Identifier and Operator Identifier, the `dnn` value in the NF Profile contains the Network Identifier only, both contain the same Network Identifier and the Operator Identifier matches one PLMN of the NF (that is, plmnList of the NF Profile).

Copyright © 2019, 2025, Oracle and/or its affiliates.

Note

Prior to 24.3.0 release, NRF was not performing dnn validations. From 24.3.0 release, dnn validations are added for NFManagement and NFDiscover service operations as per 3GPP defined format. The following points should be considered while performing the validations:

- Prior upgrading NRF to 24.3.x, in case NRF database contains NFProfiles with dnn attribute and dnn attribute value is not as per 3GPP defined format, NFs should update their NFProfiles with correct dnn format (as defined in 3GPP TS 23.003 v16.8 Section 9A). NFProfiles can be updated using NFRegister or NFUpdate (PUT or PATCH) service operations.
- Post upgrading NRF to release 24.3.x, in case there are still NFProfiles with dnn attribute value not as per 3GPP defined format, then NFProfile processing is performed for NRF service operations as follows:
  - NFRegister (PUT): NFs are allowed to re-register with dnn attribute value as per 3GPP defined format.
  - NFUpdate (PUT): NFs are allowed to update the dnn attribute value as per 3GPP defined format.
  - NFUpdate (PATCH): NFs are not allowed to update the dnn attribute value. NRF rejects the request with status code as 400 bad request.
  - NFUpdate (Heartbeat): NFs are allowed to do heartbeat for such NFProfiles as per 3GPP specifications.
  - NFDiscover:
    * NFDiscover query with dnn attribute value if not as per 3GPP defined format, is rejected by NRF.
    * The NFProfiles in the database having invalid dnn value format will be ignored while filtering NFProfiles based on dnn query attribute.
    * The NFProfiles in the database having invalid dnn value format will be included in the NFDiscover response if dnn is not present in the discovery query if other attributes present in the query matches as per 3GPP NRF specifications.
    * The NFProfiles in the database having invalid dnn value format for Operator Identifier but Network Identifier value is in correct format. dnn attribute is present in the discovery query with matching network identifier and no operator identifier. In this case, such NFProfiles will be included in the NFDiscover response.
  - NFStatusNotify: NRF sends notifications for NFUpdate and NFRegister PUT service operations with the updated NFProfiles. As NFUpdate (Patch) request is rejected, NRF does not send notifications for the service operation.

B.6 Content-Encoding Header

NRF does not accept incoming request messages in gzipped format for NFManagement service operations such as NFRegister, NFUpdate (Complete/Patch Replacement, Heartbeat), and NFStatusSubscribe (both New and Update Subscriptions).

Copyright © 2019, 2025, Oracle and/or its affiliates.

If the `Content-Encoding` header is present and has an unsupported value, NRF rejects the HTTP request with 415 Unsupported Media Type error, NRF only accepts the `Content-Encoding` header value set to “identity”. In such error cases, NRF will include an `Accept-Encoding` header with the value set to “identity.”

If the `Content-Encoding` header is missing but the payload is still compressed, NRF rejects the HTTP request with a 400 Bad Request error due to the invalid message.

However, NRF supports gzipped encoding for response messages for JSON body in NFManagement service operations such as NFRegister, NFUpdate (Complete/Patch Replacement, Heartbeat), NFListRetrieval, and NFProfileRetrieval, provided that the client (NF) specifies the gzip format in the `Accept-Encoding` header when requesting response.

Copyright © 2019, 2025, Oracle and/or its affiliates.

C

Error Response Code Details

This section lists the available error response codes in NRF categorized according to the NRF microservices. For more information, on these error response codes with their details, see the following sections:

- NF Registration Microservice
- NF Subscription Microservice
- NF Discovery Microservice
- NF Access Token Microservice
- NRF Cache Data Microservice
- NRF Configuration Microservice
- NRF Auditor Microservice
- NRF Artisan Microservice
- Ingress Gateway Service
- Egress Gateway Service
- Alternate Route Service
- Common Configuration

C.1 NF Registration Microservice

The following table lists the error codes and its problem details for the NF Registration microservices.

Table C-1 REGN - NFRegister Service Operation

Error Code	Details
ONRF-REG-REGN-E0002	This error code indicates that an invalid message format received for NFRegister service operation message.
ONRF-REG-REGN-E0007	This error code indicates that a mandatory attribute is incorrect for NFRegister service operation message.
ONRF-REG-REGN-E0008	This error code indicates that an optional attribute is incorrect for NFRegister service operation message.
ONRF-REG-REGN-E0009	This error code indicates that a mandatory attribute is missing for NFRegister service operation message.
ONRF-REG-REGN-E0023	This error code indicates that NF profile size limit breached exception occurred while processing of NFRegister service operation message.
ONRF-REG-REGN-E0021	This error code indicates that multiple attributes are missing or incorrect in NFRegister service operation message.
ONRF-REG-REGN-E0100	This error code indicates that message failure occurred generic in nature while processing the NFRegister service operation message.
ONRF-REG-REGN-E0104	This error code indicates that NF FQDN Authentication failure occurred while processing the NFRegister service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-1 (Cont.) REGN - NFRegister Service Operation

Error Code	Details
ONRF-REG-REGN-E0108	This error code indicates that NRF PLMN is not configured while processing the NFRegister service operation message.
ONRF-REG-REGN-E0109	This error code indicates that NF Management Service is unavailable while processing the NFRegister service operation message.
ONRF-REG-REGN-E0110	This error code indicates that OC-XFCC-DNS header is not present while processing the NFRegister service operation message.
ONRF-REG-REGN-E0111	This error code indicates that NF Screening failed while processing the NFRegister service operation message.
ONRF-REG-REGN-E0112	This error code indicates that Foreign PLMN is not supported while processing the NFRegister service operation message.
ONRF-REG-REGN-E0801	This error code indicates that database access exception occurred while processing the NFRegister service operation message.
ONRF-REG-REGN-E0802	This error code indicates that pessimistic locking exception occurred while processing the NFRegister service operation message.
ONRF-REG-REGN-E0803	This error code indicates that database access resource failure exception occurred while processing the NFRegister service operation message.
ONRF-REG-REGN-E0804	This error code indicates that database insertion failure occurred while processing the NFRegister service operation message.
ONRF-REG-REGN-E0807	This error code indicates that database read failure occurred while processing the NFRegister service operation message.
ONRF-REG-REGN-E0808	This error code indicates that SQL exception occurred while processing the NFRegister service operation message.
ONRF-REG-REGN-E0899	This error code indicates that general database exception occurred while processing the NFRegister service operation message.
ONRF-REG-REGN-E0901	This error code indicates that an IO exception occurred while processing the NFRegister service operation message.
ONRF-REG-REGN-E0902	This error code indicates that JSON processing exception occurred while processing the NFRegister service operation message.
ONRF-REG-REGN-E0903	This error code indicates that an interrupted exception occurred while processing the NFRegister service operation message.
ONRF-REG-REGN-E0904	This error code indicates that class cast exception occurred while processing the NFRegister service operation message.
ONRF-REG-REGN-E0905	This error code indicates that decoding exception occurred while processing the NFRegister service operation message.
ONRF-REG-REGN-E0999	This error code indicates that any other exception occurred while processing the NFRegister service operation message.

C.1.1 NFUpdate with Full Replacement (PUT)

The following table lists the error codes and its problem details for the NF Registration microservices for the NFUpdate with Full Replacement (PUT) service operation.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-2 FULLUPD - NFUpdate with Full Replacement (PUT) Service Operation

Error Code	Details
ONRF-REG-FULLUPD-E0023	This error code indicates that NF profile size limit breached exception occurred while processing of NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0100	This error code indicates that message failure occurred generic in nature while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0108	This error code indicates that NRF PLMN is not configured while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0109	This error code indicates that NF Management Service is unavailable while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0110	This error code indicates that OC-XFCC-DNS header is not present while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0111	This error code indicates that NF Screening failed while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0801	This error code indicates that database access exception occurred while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0802	This error code indicates that when pessimistic locking exception occurred while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0803	This error code indicates that database access resource failure exception occurred while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0805	This error code indicates that database update exception occurred while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0807	This error code indicates that database read failure occurred while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0808	This error code indicates that SQL exception occurred while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0899	This error code indicates that general database exception occurred while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0901	This error code indicates that IO exception occurred while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0902	This error code indicates that JSON processing exception occurred while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0903	This error code indicates that an interrupted exception occurred while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0904	This error code indicates that class cast exception occurred while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0905	This error code indicates that decoding exception occurred while processing the NFUpdate Full replacement service operation message.
ONRF-REG-FULLUPD-E0999	This error code indicates that any other exception occurred while processing the NFUpdate Full replacement service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

C.1.2 NFUpdate with Partial Replacement (PATCH)

The following table lists the error codes and its problem details for the NF Registration microservices for the NFUpdate with Partial Replacement (PATCH) service operation.

Table C-3 PARTUPD - NFUpdate with Partial Replacement (PATCH) service operation

Error Code	Details
ONRF-REG-PARTUPD-E0002	This error code indicates an invalid message format for NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0007	This error code indicates that a mandatory attribute is incorrect for NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0008	This error code indicates that an optional attribute is incorrect for NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0009	This error code indicates that a mandatory attribute is missing for NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0010	The error code indicates that an unspecified message failure occurred during the processing of the NF update partial replacement service operation message.
ONRF-REG-PARTUPD-E0023	This error code indicates that NF profile size limit breached exception occurred while processing of NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0100	This error code indicates that message failure occurred generic in nature while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0104	This error code indicates that NF FQDN Authentication failure occurred while processing of NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0108	This error code indicates that NRF PLMN is not configured while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0109	This error code indicates that NF Management Service is unavailable while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0110	This error code indicates that OC-XFCC-DNS header is not present while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0111	This error code indicates that NF Screening failed while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0801	This error code indicates that database access exception occurred while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0802	This error code indicates that when pessimistic locking exception occurred while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0803	This error code indicates that database access resource failure exception occurred while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0805	This error code indicates that database update exception occurred while processing the NFUpdate Partial replacement service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-3 (Cont.) PARTUPD - NFUpdate with Partial Replacement (PATCH) service operation

Error Code	Details
ONRF-REG-PARTUPD-E0807	This error code indicates that database read failure occurred while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0808	This error code indicates that SQL exception occurred while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0899	This error code indicates that general database exception occurred while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0901	This error code indicates that IO exception occurred while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0902	This error code indicates that JSON processing exception occurred while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0903	This error code indicates that an interrupted exception occurred while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0904	This error code indicates that class cast exception occurred while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0905	This error code indicates that decoding exception occurred while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E0999	This error code indicates that any other exception occurred while processing the NFUpdate Partial replacement service operation message.
ONRF-REG-PARTUPD-E1004	This error code indicates that specific NF Instance ID record not found while processing the NFUpdate Partial service operation message.

C.1.3 NFHeartbeat Service Operation

The following table lists the error codes and its problem details for the NF Registration microservices for the NFHeartbeat service operation.

Table C-4 HB - NFHeartbeat Service Operation

Error Code	Details
ONRF-REG-HB-E0002	This error code indicates an invalid message format for the NFHeartbeat service operation message.
ONRF-REG-HB-E0109	This error code indicates that NF Management Service is unavailable while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E0110	This error code indicates that OC-XFCC-DNS header is not present while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E0801	This error code indicates that database access exception occurred while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E0802	This error code indicates that when pessimistic locking exception occurred while processing the NFHeartbeat service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-4 (Cont.) HB - NFHeartbeat Service Operation

Error Code	Details
ONRF-REG-HB-E0803	This error code indicates that database access resource failure exception occurred while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E0805	This error code indicates that database update exception occurred while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E0807	This error code indicates that database read failure occurred while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E0808	This error code indicates that SQL exception occurred while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E0899	This error code indicates that general database exception occurred while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E0901	This error code indicates that an IO exception occurred while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E0902	This error code indicates that JSON processing exception occurred while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E0903	This error code indicates that an interrupted exception occurred while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E0904	This error code indicates that class cast exception occurred while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E0905	This error code indicates that decoding exception occurred while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E0999	This error code indicates that any other exception occurred while processing the NFHeartbeat service operation message.
ONRF-REG-HB-E1004	This error code indicates that specific NF Instance ID record not found while processing the NFHeartbeat service operation message.

C.1.4 NFDeregister Service Operation

The following table lists the error codes and its problem details for the NF Registration microservices for the NF Deregistration service operation category.

Table C-5 DEREG - NFDeregister Service Operation

Error Code	Details
ONRF-REG-DEREG-E0002	This error code indicates an invalid message format for NFDeregister service operation message.
ONRF-REG-DEREG-E0104	This error code indicates that NF FQDN Authentication failure occurred while processing the NFDeregister service operation message.
ONRF-REG-DEREG-E0109	This error code indicates that NF Management Service is unavailable while processing the NFDeregister service operation message.
ONRF-REG-DEREG-E0801	This error code indicates that database access exception occurred while processing the NFDeregister service operation message.
ONRF-REG-DEREG-E0802	This error code indicates that pessimistic locking exception occurred while processing the NFDeregister service operation message.
ONRF-REG-DEREG-E0803	This error code indicates that database access resource failure exception occurred while processing the NFDeregister service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-5 (Cont.) DEREG - NFDeregister Service Operation

Error Code	Details
ONRF-REG-DEREG-E0806	This error code indicates that database delete failure occurred while processing the NFDeregister service operation message.
ONRF-REG-DEREG-E0807	This error code indicates that database read failure occurred while processing the NFDeregister service operation message.
ONRF-REG-DEREG-E0808	This error code indicates that SQL exception occurred while processing the NFDeregister service operation message.
ONRF-REG-DEREG-E0899	This error code indicates that general database exception occurred while processing the NFDeregister service operation message.
ONRF-REG-DEREG-E0901	This error code indicates that IO exception occurred while processing the NFDeregister service operation message.
ONRF-REG-DEREG-E0902	This error code indicates that JSON processing exception occurred while processing the NFDeregister service operation message.
ONRF-REG-DEREG-E0903	This error code indicates that an interrupted exception occurred while processing the NFDeregister service operation message.
ONRF-REG-DEREG-E0904	This error code indicates that class cast exception occurred while processing the NFDeregister service operation message.
ONRF-REG-DEREG-E0999	This error code indicates that any other exception occurred while processing the NFDeregister service operation message.
ONRF-REG-DEREG-E1004	This error code indicates that specific NF Instance ID record not found while processing the NFDeregister service operation message.

C.1.5 NFProfileRetrieval Service Operation

The following table lists the error codes and its problem details for the NF Registration microservices for the NFProfileRetrieval service operation.

Table C-6 NFPR - NFProfileRetrieval Service Operation

Error Code	Details
ONRF-REG-NFPR-E0002	This error code indicates that message format is invalid while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0003	This error code indicates that query parameter is invalid while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0104	This error code indicates that NF FQDN Authentication failure occurred while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0109	This error code indicates that NF Management Service is unavailable while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0110	This error code indicates that OC-XFCC-DNS header is not present while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0801	This error code indicates that database access exception occurred while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0802	This error code indicates that pessimistic locking exception occurred while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0803	This error code indicates that database access resource failure exception occurred while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0807	This error code indicates that database read failure occurred while processing the NFProfile retrieval service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-6 (Cont.) NFPR - NFProfileRetrieval Service Operation

Error Code	Details
ONRF-REG-NFPR-E0808	This error code indicates that SQL exception occurred while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0899	This error code indicates that database access exception occurred while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0901	This error code indicates that IO exception occurred while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0902	This error code indicates that JSON processing exception occurred while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0903	This error code indicates that an interrupted exception occurred while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0904	This error code indicates that class cast exception occurred while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0905	This error code indicates that decoding exception occurred while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E0999	This error code indicates that any other exception occurred while processing the NFProfile retrieval service operation message.
ONRF-REG-NFPR-E1004	This error code indicates that specific NF Instance ID record was not found while processing the NFProfile retrieval service operation message.

C.1.6 NFListRetrieval Service Operation

The following table lists the error codes and its problem details for the NF Registration microservices for the NF List Retrieval service operation category.

Table C-7 NFLR - NFListRetrieval Service Operation

Error Code	Details
ONRF-REG-NFLR-E0002	This error code indicates that message format is invalid while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0003	This error code indicates that query parameter is invalid while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0109	This error code indicates that NF Management Service is unavailable while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0110	This error code indicates that OC-XFCC-DNS header is not present while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0801	This error code indicates that database access exception occurred while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0802	This error code indicates that pessimistic locking exception occurred while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0803	This error code indicates that database access resource failure exception occurred while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0807	This error code indicates that database read failure occurred while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0808	This error code indicates that SQL exception occurred while processing the NFListRetrieval service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-7 (Cont.) NFLR - NFListRetrieval Service Operation

Error Code	Details
ONRF-REG-NFLR-E0899	This error code indicates that database access exception occurred while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0901	This error code indicates that IO exception occurred while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0902	This error code indicates that JSON processing exception occurred while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0903	This error code indicates that an interrupted exception occurred while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0904	This error code indicates that class cast exception occurred while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0905	This error code indicates that decoding exception occurred while processing the NFListRetrieval service operation message.
ONRF-REG-NFLR-E0999	This error code indicates that any other exception occurred while processing the NFListRetrieval service operation message.

C.1.7 NFProfileRetrieval Service Operation During Forwarding

The following table lists the error codes and its problem details for the NF Registration microservices for the NFProfileRetrieval service operation during forwarding.

Table C-8 NFPRFWD - NF Profile Retrieval Service Operation During Forwarding

Error Code	Details
ONRF-REG-NFPRFWD-E0103	This error code indicates that egress gateway error occurred while forwarding the NFProfile retrieval service operation message.
ONRF-REG-NFPRFWD-E0201	This error code indicates that NRF forwarding configuration is missing while forwarding the NFProfile retrieval service operation message.
ONRF-REG-NFPRFWD-E0202	This error code indicates that NRF is not reachable while forwarding the NFProfile retrieval service operation message.
ONRF-REG-NFPRFWD-E0203	This error code indicates that NRF forwarding loop is detected while forwarding the NFProfile retrieval service operation message.
ONRF-REG-NFPRFWD-E0204	This error code indicates that another NRF returns an incorrect response for the NFProfile retrieval service operation message.
ONRF-REG-NFPRFWD-E0299	This error code indicates that some unknown internal error occurred while forwarding the NFProfile retrieval service operation message.

C.1.8 UnSupported Media Type

The following table lists the error codes and its problem details for the NF Registration microservices for the NF UnSupported Media Type service operation category.

Table C-9 UNKSROP - UnSupported Media Type

Error Code	Details
ONRF-REG-UNKSROP-E0002	This error code indicates that an unknown service operation is received.
ONRF-REG-UNKSROP-E0902	This error code indicates that JSON processing error for unknown service operation is received.

Copyright © 2019, 2025, Oracle and/or its affiliates.

C.2 NF Subscription Microservice

This section lists the error codes and its problem details for the NF Subscription microservice.

C.2.1 NFStatusSubscribe Service Operation

The following table lists the error codes and its problem details for the NF Subscription microservices for the NFStatusSubscribe service operation.

Table C-10 SUBSCR-NFStatusSubscribe Service Operation

Error Code	Details
ONRF-SUB-SUBSCR-E0007	This error code indicates that mandatory attribute is incorrect while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0009	This error code indicates that a mandatory attribute is missing while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0100	This error code indicates that generic bad request occurred while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0104	This error code indicates that NF FQDN Authentication Failure occurred while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0108	This error code indicates that NRF PLMN is not configured while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0109	This error code indicates that NF Management Service is unavailable while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0110	This error code indicates that OC-XFCC-DNS header is not present while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0111	This error code indicates that NF screening rule fails while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0112	This error code indicates that Foreign PLMN is not supported while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0801	This error code indicates that database access exception occurred while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0802	This error code indicates that pessimistic locking exception occurred while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0803	This error code indicates that database access resource failure exception occurred while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0899	This error code indicates that generic database exception occurred while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0902	This error code indicates that json processing exception occurred while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0903	This error code indicates that an interrupted exception occurred while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0904	This error code indicates that class cast exception occurred while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E0999	This error code indicates that any generic exception occurred while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E2001	This error code indicates that create Subscription based on NfInstanceId condition, the relevant NFProfiles not found for NFInstanceId while processing the NFStatusSubscribe service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-10 (Cont.) SUBSCR-NFStatusSubscribe Service Operation

Error Code	Details
ONRF-SUB-SUBSCR-E2002	This error code indicates that Subscription is not authorized for NFInstanceId present in subscrCond while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E2003	This error code indicates that Subscription limit is breached and new subscriptions are not allowed while processing the NFStatusSubscribe service operation message.
ONRF-SUB-SUBSCR-E2004	This error code indicates that failure occurred while adding, deleting, updating records in NFSubscription table while processing the NFStatusSubscribe service operation message.

C.2.1.1 NFStatusSubscribe Update (PATCH) Service Operation

The following table lists the error codes and its problem details for the NF Subscription microservices for the NFStatusSubscribe Update (PATCH) service operation.

Table C-11 UPDSUBS - NFStatusSubscribe Update (PATCH) Service Operation

Error Code	Details
ONRF-SUB-UPDSUBS-E0007	This error code indicates that mandatory attribute is incorrect while processing the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPDSUBS-E0012	This error code indicates that specific subscription not found while processing the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPDSUBS-E0022	This error code indicates that unsupported media type comes while processing the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPDSUBS-E0100	This error code indicates that generic bad request occurred while processing the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPDSUBS-E0104	This error code indicates that NF FQDN Authentication Failure occurred while processing the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPDSUBS-E0108	This error code indicates that NRF PLMN is not configured while processing the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPDSUBS-E0109	This error code indicates that NF Management Service is unavailable while processing the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPDSUBS-E0110	This error code indicates that OC-XFCC-DNS header is not present while processing the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPDSUBS-E0801	This error code indicates that database access exception occurred while processing the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPDSUBS-E0802	This error code indicates that pessimistic locking exception occurred while processing the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPDSUBS-E0803	This error code indicates that database access resource failure exception occurred while processing the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPDSUBS-E0805	This error code indicates that update database exception occurred while processing the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPDSUBS-E0902	This error code indicates that json processing exception occurred while processing the NFStatusSubscribe Update service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-11 (Cont.) UPDSUBS - NFStatusSubscribe Update (PATCH) Service Operation

Error Code	Details
ONRF-SUB-UPDSUBS-E0999	This error code indicates that any generic exception occurred while processing the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPDSUBS-E2003	This error code indicates that subscription limit is breached and subscriptions are not allowed to update while processing the NFStatusSubscribe Update service operation message.

C.2.1.2 NFStatusSubscribe Service Operation During Forwarding

The following table lists the error codes and its problem details for the NF Subscription microservices for the NFStatusSubscribe service operation during forwarding.

Table C-12 SUBFWD-NFStatusSubscribe Service Operation During Forwarding

Error Code	Details
ONRF-SUB-SUBFWD-E0103	This error code indicates that egress gateway error occurred for NFStatusSubscribe service operation message targeted for forwarding to another NRF.
ONRF-SUB-SUBFWD-E0201	This error code indicates that NRF forwarding configuration is missing while forwarding the NFStatusSubscribe service operation message.
ONRF-SUB-SUBFWD-E0202	This error code indicates that NRF is not reachable while forwarding the NFStatusSubscribe service operation message.
ONRF-SUB-SUBFWD-E0203	This error code indicates that NRF forwarding loop is detected for NFStatusSubscribe service operation message being forwarded.
ONRF-SUB-SUBFWD-E0204	This error code indicates that another NRF returned incorrect response for NFStatusSubscribe service operation message.
ONRF-SUB-SUBFWD-E0205	This error code indicates that egress gateway is not reachable while trying to forward the NFStatusSubscribe service operation message to another NRF.
ONRF-SUB-SUBFWD-E0206	This error code indicates that some unexpected error occurred while forwarding the NFStatusSubscribe service operation message.
ONRF-SUB-SUBFWD-E0299	This error code indicates that some unknown internal error occurred while forwarding the NFStatusSubscribe service operation message.

C.2.1.3 NFStatusSubscribe Update (PATCH) Service Operation During Forwarding

The following table lists the error codes and its problem details for the NF Subscription microservices for the NFStatusSubscribe update (PATCH) service operation during forwarding.

Table C-13 UPSFWD - NFStatusSubscribe Update (PATCH) Service Operation During Forwarding

Error Code	Details
ONRF-SUB-UPSFWD-E0012	This error code indicates that subscription is not found for specific subscriptionId for NFStatusSubscribe Update service operation being forwarded.
ONRF-SUB-UPSFWD-E0103	This error code indicates that egress gateway error occurred for NFStatusSubscribe Update service operation message targeted for forwarding to another NRF.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-13 (Cont.) UPSFWD - NFStatusSubscribe Update (PATCH) Service Operation During Forwarding

Error Code	Details
ONRF-SUB-UPSFWD-E0201	This error code indicates that NRF forwarding configuration is missing while forwarding the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPSFWD-E0202	This error code indicates that NRF is not reachable while forwarding the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPSFWD-E0203	This error code indicates that NRF forwarding loop is detected for NFStatusSubscribe Update service operation message being forwarded.
ONRF-SUB-UPSFWD-E0204	This error code indicates that another NRF returned incorrect response for NFStatusSubscribe Update service operation message.
ONRF-SUB-UPSFWD-E0205	This error code indicates that egress gateway is not reachable while trying to forward the NFStatusSubscribe Update service operation message to another NRF.
ONRF-SUB-UPSFWD-E0206	This error code indicates that some unexpected error occurred while forwarding the NFStatusSubscribe Update service operation message.
ONRF-SUB-UPSFWD-E0299	This error code indicates that some unknown internal error occurred while forwarding the NFStatusSubscribe Update service operation message.

C.2.1.4 NFStatusSubscribe Update (PATCH) Service Operation During Roaming

The following table lists the error codes and its problem details for the NF Subscription microservices for the NFStatusSubscribe update (PATCH) service operation during roaming.

Table C-14 UPSROAM - NFStatusSubscribe Update (PATCH) Service Operation During Roaming Scenarios

Error Code	Details
ONRF-SUB-UPSROAM-E0301	This error code indicates that 5xx error response is received for roaming scenarios.
ONRF-SUB-UPSROAM-E0302	This error code indicates that 4xx error response is received for roaming scenarios.
ONRF-SUB-UPSROAM-E0303	This error code indicates that 3xx error response is received for roaming scenarios.
ONRF-SUB-UPSROAM-E0304	This error code indicates that 5xx error response is received from egress gateway for roaming scenarios.
ONRF-SUB-UPSROAM-E0305	This error code indicates that 4xx error response is received from egress gateway for roaming scenarios.
ONRF-SUB-UPSROAM-E0306	This error code indicates that 3xx error response is received from egress gateway for roaming scenarios.
ONRF-SUB-UPSROAM-E0308	This error code indicates that loop is detected during roaming scenario for roaming scenarios.
ONRF-SUB-UPSROAM-E0399	This error code indicates that generic internal error occurred for roaming scenarios.
ONRF-SUB-UPSROAM-E0205	This error code indicates that egress gateway is not reachable for roaming scenarios.
ONRF-SUB-UPSROAM-E0206	This error code indicates that some unexpected error occurred for roaming scenarios.

Copyright © 2019, 2025, Oracle and/or its affiliates.

C.2.2 NFStatusUnsubscribe Service Operation

The following table lists the error codes and its problem details for the NF Subscription microservices for the NFStatusUnsubscribe service operation.

Table C-15 UNSUBSC - NFStatusUnSubscribe Service Operation

Error Code	Details
ONRF-SUB-UNSUBSC-E0012	This error code indicates that specific subscription not found while processing the NFStatusUnSubscribe service operation message.
ONRF-SUB-UNSUBSC-E0104	This error code indicates that NF FQDN Authentication Failure occurred while processing the NFStatusUnSubscribe service operation message.
ONRF-SUB-UNSUBSC-E0108	This error code indicates that NRF PLMN is not configured is not found while processing the NFStatusUnSubscribe service operation message.
ONRF-SUB-UNSUBSC-E0109	This error code indicates that NF Management Service is unavailable while processing the NFStatusUnSubscribe service operation message.
ONRF-SUB-UNSUBSC-E0110	This error code indicates that OC-XFCC-DNS header is not present while processing the NFStatusUnSubscribe service operation message.
ONRF-SUB-UNSUBSC-E0801	This error code indicates that database access exception occurred while processing the NFStatusUnSubscribe service operation message.
ONRF-SUB-UNSUBSC-E0802	This error code indicates that pessimistic locking exception occurred while processing the NFStatusUnSubscribe service operation message.
ONRF-SUB-UNSUBSC-E0803	This error code indicates that database access resource failure exception occurred while processing the NFStatusUnSubscribe service operation message.
ONRF-SUB-UNSUBSC-E0805	This error code indicates that update database exception occurred while processing the NFStatusUnSubscribe service operation message.
ONRF-SUB-UNSUBSC-E0903	This error code indicates that interrupted exception occurred while processing the NFStatusUnSubscribe service operation message.
ONRF-SUB-UNSUBSC-E0904	This error code indicates that class cast exception occurred while processing the NFStatusUnSubscribe service operation message.
ONRF-SUB-UNSUBSC-E0999	This error code indicates that generic exception occurred while processing the NFStatusUnSubscribe service operation message.

C.2.2.1 NFStatusUnsubscribe Service Operation During Forwarding

The following table lists the error codes and its problem details for the NF Subscription microservices for the NFStatusUnsubscribe service operation during forwarding.

Table C-16 UNSFWD - NFStatusUnsubscribe Service Operation During Forwarding

Error Code	Details
ONRF-SUB-UNSFWD-E0012	This error code indicates that subscription is not found for specific subscriptionId for NFStatusUnsubscribe service operation being forwarded.
ONRF-SUB-UNSFWD-E0103	This error code indicates that egress gateway error occurred for NFStatusUnsubscribe service operation message. targeted for forwarding to another NRF.
ONRF-SUB-UNSFWD-E0201	This error code indicates that NRF forwarding configuration is missing while forwarding the NFStatusUnsubscribe service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-16 (Cont.) UNSFWD - NFStatusUnsubscribe Service Operation During Forwarding

Error Code	Details
ONRF-SUB-UNSFWD-E0202	This error code indicates that NRF is not reachable while forwarding the NFStatusUnsubscribe service operation message.
ONRF-SUB-UNSFWD-E0203	This error code indicates that NRF forwarding loop is detected for NFStatusUnsubscribe service operation message. being forwarded
ONRF-SUB-UNSFWD-E0204	This error code indicates that another NRF returned incorrect response for NFStatusUnsubscribe service operation message.
ONRF-SUB-UNSFWD-E0205	This error code indicates that egress gateway is not reachable while trying to forward the NFStatusUnsubscribe service operation message to another NRF.
ONRF-SUB-UNSFWD-E0206	This error code indicates that some unexpected error occurred while forwarding the NFStatusUnsubscribe service operation message.
ONRF-SUB-UNSFWD-E0299	This error code indicates that some unknown internal error occurred while forwarding the NFStatusUnsubscribe service operation message.

C.2.3 NFStatusSubscribe Service Operation During Roaming

The following table lists the error codes and its problem details for the NF Subscription microservices for the NFStatusSubscribe service operation during roaming.

Table C-17 SUBROAM - NFStatusSubscribe Service Operation During Roaming

Error Code	Details
ONRF-SUB-SUBROAM-E0301	This error code indicates that 5xx error response is received for roaming scenarios.
ONRF-SUB-SUBROAM-E0302	This error code indicates that 4xx error response is received for roaming scenarios.
ONRF-SUB-SUBROAM-E0303	This error code indicates that 3xx error response is received for roaming scenarios.
ONRF-SUB-SUBROAM-E0304	This error code indicates that 5xx error response is received from egress gateway for roaming scenarios.
ONRF-SUB-SUBROAM-E0305	This error code indicates that 4xx error response is received from egress gateway for roaming scenarios.
ONRF-SUB-SUBROAM-E0306	This error code indicates that 3xx error response is received from egress gateway for roaming scenarios.
ONRF-SUB-SUBROAM-E0308	This error code indicates that loop is detected during roaming scenario for roaming scenarios.
ONRF-SUB-SUBROAM-E0309	This error code indicates that target PLMN not matches with NRF PLMN for roaming scenarios.
ONRF-SUB-SUBROAM-E0399	This error code indicates that generic internal error occurred for roaming scenarios.
ONRF-SUB-SUBROAM-E2008	This error code indicates that not able to determine that NFStatusSubscribe service operation message is intended for HomeNRF or Visited NRF.
ONRF-SUB-SUBROAM-E2009	This error code indicates that nfStatusNotificationURI is not present in inter-plmn format for roaming scenarios.
ONRF-SUB-SUBROAM-E0205	This error code indicates that egress gateway is not reachable for roaming scenarios.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-17 (Cont.) SUBROAM - NFStatusSubscribe Service Operation During Roaming

Error Code	Details
ONRF-SUB-SUBROAM-E0206	This error code indicates that some unexpected error occurred for roaming scenarios.

C.2.4 NFStatusUnsubscribe Service Operation During Roaming

The following table lists the error codes and its problem details for the NF Subscription microservices for the NFStatusUnsubscribe service operation during roaming.

Table C-18 UNSROAM - NFStatusUnSubscribe Service Operation During Roaming

Error Code	Details
ONRF-SUB-UNSROAM-E0301	This error code indicates that 5xx error response is received for roaming scenarios.
ONRF-SUB-UNSROAM-E0302	This error code indicates that 4xx error response is received for roaming scenarios.
ONRF-SUB-UNSROAM-E0303	This error code indicates that 3xx error response is received for roaming scenarios.
ONRF-SUB-UNSROAM-E0304	This error code indicates that 5xx error response is received from egress gateway for roaming scenarios.
ONRF-SUB-UNSROAM-E0305	This error code indicates that 4xx error response is received from egress gateway for roaming scenarios.
ONRF-SUB-UNSROAM-E0306	This error code indicates that 3xx error response is received from egress gateway for roaming scenarios.
ONRF-SUB-UNSROAM-E0308	This error code indicates that loop is detected during roaming scenario for roaming scenarios.
ONRF-SUB-UNSROAM-E0399	This error code indicates that generic internal error occurred for roaming scenarios.
ONRF-SUB-UNSROAM-E0205	This error code indicates that egress gateway is not reachable for roaming scenarios.
ONRF-SUB-UNSROAM-E0206	This error code indicates that some unexpected error occurred for roaming scenarios.

C.2.5 NFStatusNotify Service Operation

The following table lists the error codes and its problem details for the NF Subscription microservices for the NFStatusNotify service operation.

Table C-19 SNOTIF - NFStatus Notify Service Operation

Error Code	Details
ONRF-SUB-SNOTIF-E0009	This error code indicates that mandatory attribute is missing while processing the NFStatusNotify service operation message trigger coming from NFRegistration microservice.
ONRF-SUB-SNOTIF-E0012	This error code indicates that subscription not found while processing the NFStatusNotify service operation message trigger coming from NFRegistration microservice.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-19 (Cont.) SNOTIF - NFStatus Notify Service Operation

Error Code	Details
ONRF-SUB-SNOTIF-E0100	This error code indicates that generic bad request received while processing the NFStatusNotify service operation message trigger coming from NFRegistration microservice.
ONRF-SUB-SNOTIF-E0803	This error code indicates that database resource failure exception occurred while processing the NFStatusNotify service operation message trigger coming from NFRegistration microservice.
ONRF-SUB-SNOTIF-E0999	This error code indicates that generic exception occurred while processing the NFStatusNotify service operation message trigger coming from NFRegistration microservice.
ONRF-SUB-SNOTIF-E2007	This error code indicates that targetNFtype is null while processing the NFStatusNotify service operation message trigger coming from NFRegistration microservice.

C.2.6 NFStatus UnSupported HTTP Method for Service Operation

The following table lists the error codes and its problem details for the NF Subscription microservices for the NFStatus UnSupported HTTP Method for service operation.

Table C-20 INVSROP - UnSupported HTTP Method for Service Operation

Error Code	Details
ONRF-SUB-INVSROP-E0910	This error code indicates that an invalid service operation is received.

C.2.7 NFStatus UnSupported Media Type

The following table lists the error codes and its problem details for the NF Subscription microservices for the NFStatus UnSupported media type.

Table C-21 UNKSROP - UnSupported Media Type

Error Code	Details
ONRF-SUB-UNKSROP-E0905	This error code indicates that an unknown service operation decoding error occurred.
ONRF-SUB-UNKSROP-E0999	This error code indicates that generic error occurred for service operation.

C.3 NF Discovery Microservice

The following table lists the error codes and its problem details for the NF Discovery microservices.

C.3.1 NFDiscover Service Operation

The following table lists the error codes and its problem details for the NF Discovery microservices for the NFDiscover service operation.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-22 DISC - NFDiscover Service Operation

Error Code	Details
ONRF-DIS-DISC-E0003	This error code indicates that the discovery query contains an invalid query parameter.
ONRF-DIS-DISC-E0004	This error code indicates that the discovery query contains an incorrect mandatory query parameter.
ONRF-DIS-DISC-E0005	This error code indicates that the discovery query contains an incorrect optional query parameter.
ONRF-DIS-DISC-E0006	This error code indicates that a mandatory query parameter is missing in the discovery query.
ONRF-DIS-DISC-E0007	This error code indicates that a part of the query parameter value is incorrect.
ONRF-DIS-DISC-E0010	The error code indicates that an unspecified message failure occurred during the processing of the discovery query.
ONRF-DIS-DISC-E0021	This error code indicates that multiple attributes are incorrect in the discovery query.
ONRF-DIS-DISC-E0100	This error code indicates that message failure occurred which is generic in nature while processing the discovery query.
ONRF-DIS-DISC-E0108	This error code indicates that NRF PLMN is not configured and there is need to match NRF PLMN to incoming PLMN in discovery query.
ONRF-DIS-DISC-E0902	This error code indicates that JSON processing exception occurred while processing the NFDiscover service operation message.
ONRF-DIS-DISC-E0903	This error code indicates that an interrupted exception occurred while processing the NFDiscover service operation message.
ONRF-DIS-DISC-E0999	This error code indicates that any generic error is occurred while processing the NFDiscover service operation message.
ONRF-DIS-DISC-E3003	This error code indicates that requester NF is not allowed to discover the specific NF producers while processing the NFDiscover service operation message.

C.3.2 NFDiscover Service Operation During Authorization

The following table lists the error codes and its problem details for the NF Discovery microservices for the NFDiscover service operation during authorization.

Table C-23 DISAUTH - NFDiscover Service Operation During Authorization

Error Code	Details
ONRF-DIS-DISAUTH-E0104	This error code indicates that NF FQDN authentication failure occurred while processing the NFDiscover service operation message.
ONRF-DIS-DISAUTH-E0899	This error code indicates that general error occurred for authentication while processing the NFDiscover service operation message.

C.3.3 NFDiscover Service Operation During Forwarding

The following table lists the error codes and its problem details for the NF Discovery microservices for the NFDiscover service operation during forwarding.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-24 DISCFWD - NFDiscover Service Operation During Forwarding

Error Code	Details
ONRF-DIS-DISCFWD-E0002	This error code indicates that message format is invalid while forwarding the NFDiscover service operation message to another NRF.
ONRF-DIS-DISCFWD-E0005	This error code indicates that an optional query parameter is incorrect while forwarding the NFDiscover service operation message to another NRF.
ONRF-DIS-DISCFWD-E0103	This error code indicates that egress gateway error occurred while forwarding the NFDiscover service operation message to another NRF.
ONRF-DIS-DISCFWD-E0201	This error code indicates that NRF forwarding configuration is missing while forwarding the NFDiscover service operation message to another NRF.
ONRF-DIS-DISCFWD-E0202	This error code indicates that NRF is not reachable while forwarding the NFDiscover service operation message to another NRF.
ONRF-DIS-DISCFWD-E0203	This error code indicates that NRF forwarding loop is detected while forwarding the NFDiscover service operation message to another NRF.
ONRF-DIS-DISCFWD-E0204	This error code indicates that parsing of problem details failed for the response received back from another NRF for forwarding scenarios.
ONRF-DIS-DISCFWD-E0205	This error code indicates that egress gateway is not reachable while trying to forward the NFDiscover Service Operation message to another NRF.
ONRF-DIS-DISCFWD-E0206	This error code indicates that some unexpected error occurred while forwarding the NFDiscover Service Operation message.
ONRF-DIS-DISCFWD-E0299	This error code indicates that NRF forwarding internal error is occurred while forwarding the NFDiscover service operation message to another NRF.
ONRF-DIS-DISCFWD-E0902	This error code indicates that JSON processing exception occurred while forwarding the NFDiscover service operation message to another NRF.

C.3.4 SLF for NFDiscover

The following table lists the error codes and its problem details for the NF Discovery microservices for SLF for NFDiscover.

Table C-25 DISCSLF - SLF for NFDiscover

Error Code	Details
ONRF-DIS-DISCSLF-E0103	This error code indicates that egress gateway error occurred for SLF query.
ONRF-DIS-DISCSLF-E0401	This error code indicates that mandatory attributes are missing for SLF query.
ONRF-DIS-DISCSLF-E0402	This error code indicates that SLF configuration is missing for SLF query.
ONRF-DIS-DISCSLF-E0403	This error code indicates that SLF/UDR is not reachable for SLF query.
ONRF-DIS-DISCSLF-E0404	This error code indicates that subscriber is not configured at UDR for specific subscriber details in SLF query.
ONRF-DIS-DISCSLF-E0406	This error code indicates that 4xx error response is received for SLF query.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-25 (Cont.) DISCSLF - SLF for NFDiscover

Error Code	Details
ONRF-DIS-DISCSLF-E0407	This error code indicates that 5xx error response is received for SLF query.
ONRF-DIS-DISCSLF-E0408	This error code indicates that SLF Oauth token error occurred for SLF query.
ONRF-DIS-DISCSLF-E0499	This error code indicates that any internal error occurred for SLF query.
ONRF-DIS-DISCSLF-E3004	This error code indicates that SLF discovered candidate list is empty for DISCOVERED_SLF_CONFIG_MODE.
ONRF-DIS-DISCSLF-E0205	This error code indicates that egress gateway is not reachable for SLF query.
ONRF-DIS-DISCSLF-E0206	This error code indicates that some unexpected error occurred for SLF query.

C.3.5 NFDiscover Service Operation During Roaming

The following table lists the error codes and its problem details for the NF Discovery microservices for the NFDiscover service operation during roaming.

Table C-26 DISROAM - NFDiscover Service Operation During Roaming

Error Code	Details
ONRF-DIS-DISROAM-E0002	This error code indicates that the message format is invalid for roaming scenarios.
ONRF-DIS-DISROAM-E0005	This error code indicates that an optional query parameter is incorrect for roaming scenarios.
ONRF-DIS-DISROAM-E0101	This error code indicates that a mandatory attribute missing for roaming scenarios.
ONRF-DIS-DISROAM-E0102	This error code indicates that the user agent header is missing for roaming scenarios.
ONRF-DIS-DISROAM-E0103	This error code indicates that egress gateway error occurred for roaming scenarios.
ONRF-DIS-DISROAM-E0205	This error code indicates that egress gateway is not reachable for roaming scenarios.
ONRF-DIS-DISROAM-E0206	This error code indicates that some unexpected error occurred for roaming scenarios.
ONRF-DIS-DISROAM-E0301	This error code indicates that 5xx error response is received for roaming scenarios.
ONRF-DIS-DISROAM-E0302	This error code indicates that 4xx error response is received for roaming scenarios.
ONRF-DIS-DISROAM-E0303	This error code indicates that 3xx error response is received for roaming scenarios.
ONRF-DIS-DISROAM-E0304	This error code indicates that 5xx error response is received for egress gateway for roaming scenarios.
ONRF-DIS-DISROAM-E0305	This error code indicates that 4xx error response is received for egress gateway for roaming scenarios.
ONRF-DIS-DISROAM-E0306	This error code indicates that 3xx error response is received for egress gateway for roaming scenarios.
ONRF-DIS-DISROAM-E0308	This error code indicates that loop is detected for roaming scenarios.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-26 (Cont.) DISROAM - NFDiscover Service Operation During Roaming

Error Code	Details
ONRF-DIS-DISROAM-E0309	This error code indicates that target not matches the NRF PLMN for roaming scenarios.
ONRF-DIS-DISROAM-E0399	This error code indicates that internal error occurred for roaming scenarios.

C.3.6 Unsupported HTTP Method for NFDiscover Service Operation

The following table lists the error codes and its problem details for the unsupported HTTP method for NFDiscover service operation.

Table C-27 INVSROP - UnSupported HTTP Method for Service Operation

Error Code	Details
ONRF-DIS-INVSROP-E0910	This error code indicates that an invalid service operation is received.

C.3.7 Unsupported Media Type for NFDiscover Service Operation

The following table lists the error codes and its problem details for the unsupported media type for NFDiscover service operation.

Table C-28 UNKSROP - UnSupported Media Type

Error Code	Details
ONRF-DIS-UNKSROP-E0999	This error code indicates that generic error occurred for the NFDiscover service operation.

C.4 NF Access Token Microservice

The following table lists the error codes and its problem details for the NF Access Token microservices.

C.4.1 NF Access Token Microservice

The following table lists the error codes and its problem details for the NF Access Token microservices for the NFAccessToken service operation category.

Table C-29 ACTOK - Access Token Service Operation

Error Code	Details
ONRF-ACC-ACTOK-E0002	This error code indicates that an invalid message format is received for Access Token service operation message.
ONRF-ACC-ACTOK-E0105	This error code indicates that requester NF is unauthorized for Access Token service operation message.
ONRF-ACC-ACTOK-E0106	This error code indicates that invalid key details are configured during Access Token service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-29 (Cont.) ACTOK - Access Token Service Operation

Error Code	Details
ONRF-ACC-ACTOK-E0107	This error code indicates that current key id is not configured during Access Token service operation message.
ONRF-ACC-ACTOK-E0802	This error code indicates that pessimistic locking exception occurred during Access Token service operation message.
ONRF-ACC-ACTOK-E0803	This error code indicates that data access resource failure exception occurred during Access Token service operation message.
ONRF-ACC-ACTOK-E0899	This error code indicates that generic DB exception occurred during Access Token service operation message.
ONRF-ACC-ACTOK-E0902	This error code indicates that JSON processing exception occurred during Access Token service operation message.
ONRF-ACC-ACTOK-E0903	This error code indicates that an interrupted exception occurred during Access Token service operation message.
ONRF-ACC-ACTOK-E0999	This error code indicates that general exception occurred during Access Token service operation message.
ONRF-ACC-ACTOK-E4001	This error code indicates that unable to deduce the targetNfType during Access Token service operation message.
ONRF-ACC-ACTOK-E4002	This error code indicates that OauthTokenIssuerId in TokenSigningDetails is not configured during Access Token service operation message.
ONRF-ACC-ACTOK-E4004	This error code indicates that attribute values are different from the registered NFProfile during Access Token service operation message.
ONRF-ACC-ACTOK-E4011	This error code indicates that attributes shall only be present for targetNfType during Access Token service operation message.
ONRF-ACC-ACTOK-E4015	This error code indicates that oAuth2 client is not registered with OCNRF during Access Token service operation message.
ONRF-ACC-ACTOK-E4101	This error code indicates that an invalid grant is received during Access Token service operation message.
ONRF-ACC-ACTOK-E4102	This error code indicates that an invalid scope is received during Access Token service operation message.
ONRF-ACC-ACTOK-E4103	This error code indicates that invalid request is received during Access Token service operation message.

C.4.2 Access Token Service Operation During Forwarding

The following table lists the error codes and its problem details for the NF Access Token microservices for the Access Token service operation during forwarding.

Table C-30 ACFWD - Access Token Service Operation During Forwarding

Error Code	Details
ONRF-ACC-ACFWD-E0002	This error code indicates that an invalid message format received while forwarding the Access Token service operation message.
ONRF-ACC-ACFWD-E0103	This error code indicates that egress gateway error occurred while forwarding the Access Token service operation message.
ONRF-ACC-ACFWD-E0201	This error code indicates that NRF forwarding configuration missing while forwarding the Access Token service operation message.
ONRF-ACC-ACFWD-E0202	This error code indicates that NRF is not reachable error occurred while forwarding the Access Token service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-30 (Cont.) ACFWD - Access Token Service Operation During Forwarding

Error Code	Details
ONRF-ACC-ACFWD-E0203	This error code indicates that NRF loop is detected while forwarding the Access Token service operation message.
ONRF-ACC-ACFWD-E0204	This error code indicates that parsing of problem details response received from another NRF while forwarding the Access Token service operation message.
ONRF-ACC-ACFWD-E0299	This error code indicates that forwarding internal error occurred while forwarding the Access Token service operation message.
ONRF-ACC-ACFWD-E4016	This error code indicates that NRF failed to forward the Access Token service operation message.

C.4.3 Access Token Service Operation During Roaming

The following table lists the error codes and its problem details for the NF Access Token microservices for the Access Token service operation during roaming.

Table C-31 ACROAM - Access Token Service Operation During Roaming

Error Codes	Details
ONRF-ACC-ACROAM-E0002	This error code indicates that invalid message format received during roaming scenario of Access Token service operation message.
ONRF-ACC-ACROAM-E0108	This error code indicates that NRF plmn is not configured during roaming scenario of Access Token service operation message.
ONRF-ACC-ACROAM-E0301	This error code indicates that NRF peer 5xx response received during roaming scenario of Access Token service operation message.
ONRF-ACC-ACROAM-E0302	This error code indicates that NRF peer 4xx response received during roaming scenario of Access Token service operation message.
ONRF-ACC-ACROAM-E0303	This error code indicates that NRF peer 3xx response received during roaming scenario of Access Token service operation message.
ONRF-ACC-ACROAM-E0304	This error code indicates that Egress gateway 5xx response received during roaming scenario of Access Token service operation message.
ONRF-ACC-ACROAM-E0305	This error code indicates that Egress gateway 4xx response received during roaming scenario of Access Token service operation message.
ONRF-ACC-ACROAM-E0306	This error code indicates that Egress gateway 3xx response received during roaming scenario of Access Token service operation message.
ONRF-ACC-ACROAM-E0308	This error code indicates that loop is detected during roaming scenario of Access Token service operation message.
ONRF-ACC-ACROAM-E0309	This error code indicates that target PLMN not matches NRF PLMN during roaming scenario of Access Token service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-31 (Cont.) ACROAM - Access Token Service Operation During Roaming

Error Codes	Details
ONRF-ACC-ACROAM-E0399	This error code indicates that internal error occurred for roaming scenarios of Access Token service operation message.
ONRF-ACC-ACROAM-E0807	This error code indicates that database read error occurred during roaming scenario of Access Token service operation message.
ONRF-ACC-ACROAM-E0902	This error code indicates that JSON processing error occurred during roaming scenario of Access Token service operation message.
ONRF-ACC-ACROAM-E4004	This error code indicates that attribute values are different from the registered NF profile during roaming scenario of Access Token service operation message.
ONRF-ACC-ACROAM-E4015	This error code indicates that oAuth2 client is not registered with OCNRF during roaming scenario of Access Token service operation message.
ONRF-ACC-ACROAM-E4017	This error code indicates that NRF failed to forward the roaming scenario of Access Token service operation message.

C.4.4 Authentication During Access Token Service Operation

The following table lists the error codes and its problem details for the NF Access Token microservices for the Authentication during Access Token service operation.

Table C-32 ACAUTHN - Authentication During Access Token Service Operation

Error Codes	Details
ONRF-ACC-ACAUTHN-E0018	This error code indicates that some system failure occurred during authentication scenario of Access Token service operation message.
ONRF-ACC-ACAUTHN-E0104	This error code indicates that NF FQDN authentication failure occurred during authentication scenario of Access Token service operation message.
ONRF-ACC-ACAUTHN-E0110	This error code indicates that OC-XFCC-DNS header is missing during authentication scenario of Access Token service operation message.
ONRF-ACC-ACAUTHN-E4003	This error code indicates that requester NF is not authorized to access targetNF services during authentication scenario of Access Token service operation message.
ONRF-ACC-ACAUTHN-E4004	This error code indicates that attribute values are different from the registered NF profile during authentication scenario of Access Token service operation message.
ONRF-ACC-ACAUTHN-E4006	This error code indicates that targetNFInstanceId does not match the configured NRF instance id during authentication scenario of Access Token service operation message.
ONRF-ACC-ACAUTHN-E4007	This error code indicates that target NF Type is not NRF during authentication scenario of Access Token service operation message.
ONRF-ACC-ACAUTHN-E4008	This error code indicates that no NFs found for the specific information during authentication scenario of Access Token service operation message.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-32 (Cont.) ACAUTHN - Authentication During Access Token Service Operation

Error Codes	Details
ONRF-ACC-ACAUTHN-E4009	This error code indicates that no NFs found for the requested scope during authentication scenario of Access Token service operation message.
ONRF-ACC-ACAUTHN-E4010	This error code indicates that NFs with specific attribute values are not allowed to access the requested NFProfile during authentication scenario of Access Token service operation message.
ONRF-ACC-ACAUTHN-E4012	This error code indicates that NF services in scope doesn't belong to scope of same NF type during authentication scenario of Access Token service operation message.
ONRF-ACC-ACAUTHN-E4013	This error code indicates that Either targetNfType or targetNfInstanceId needs to be present if token is requested for only custom service(s) during authentication scenario of Access Token service operation message.
ONRF-ACC-ACAUTHN-E4014	This error code indicates that no NfInstance is found with the requested targetNfInstanceId during authentication scenario of Access Token service operation message.
ONRF-ACC-ACAUTHN-E4015	This error code indicates that oAuth2 client is not registered with OCNRF during authentication scenario of Access Token service operation message.

C.4.5 Authorization During Access Token Service Operation

The following table lists the error codes and its problem details for the NF Access Token microservices for the Authorization during Access Token service operation.

Table C-33 ACAUTHZ - Authorization During Access Token Service Operation

Error Codes	Details
ONRF-ACC-ACAUTHZ-E0899	This error code indicates that general database exception occurred during authorization scenario of Access Token service operation message.
ONRF-ACC-ACAUTHZ-E0902	This error code indicates that JSON processing exception occurred during authorization scenario of Access Token service operation message.
ONRF-ACC-ACAUTHZ-E4015	This error code indicates that oAuth2 client is not registered with OCNRF during authorization scenario of Access Token service operation message.

C.5 NRF Cache Data Microservice

This section lists the error codes and its problem details for the NRF Cache Data microservice.

C.5.1 GET Subscription

The following table lists the error codes and its problem details for the NRF Cache Data microservice for the GET Subscription service operation.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-34 GTSUB - Get Subscription

Error Code	Details
ONRF-CDS-GTSUB-E5001	This error code indicates that cache fallback failed for getting subscriptions.
ONRF-CDS-GTSUB-E0999	This error code indicates that any other exception occurred while getting subscriptions from cache data.

C.5.2 GET Remote Discovery Instance and Remote Instance Data

This section lists the error codes and its problem details for the NRF Cache Data microservice for GET remote discovery instance and remote instance data.

Table C-35 NFINSTS - GET Remote Discovery Instance and Remote Instance Data

Error Code	Details
ONRF-CDS-NFINSTS-E5001	This error code indicates that cache fallback failed for remote instance discovery.
ONRF-CDS-NFINSTS-E0999	This error code indicates that any other exception occurred while remote instance discovery.

C.5.3 NFListRetrieval Service Operation

This section lists the error codes and its problem details for the NRF Cache Data microservice for NFListRetrieval service operation.

Table C-36 NFLR - NFListRetrieval Service Operation

Error Code	Details
ONRF-CDS-NFLR-E5001	This error code indicates that cache fallback failed for NF list retrieval.

C.5.4 NFProfileRetrieval Service Operation

This section lists the error codes and its problem details for the NRF Cache Data microservice for NFProfileRetrieval service operation.

Table C-37 NFPR - NFProfileRetrieval Service Operation

Error Code	Details
ONRF-CDS-NFLR-E5001	This error code indicates that cache fallback failed for NF list retrieval.

C.6 NRF Auditor Microservice

This section lists the error codes and its problem details for the NRF Auditor microservice.

Copyright © 2019, 2025, Oracle and/or its affiliates.

C.6.1 Heart Beat for NRF Auditor Service Operation

The following table lists the error codes and its problem details for the Heart Beat for NRF Auditor service operation.

Table C-38 HBAUD - Heart Beat for NRF Auditor Service Operation

Error Codes	Details
ONRF-AUD-HBAUD-E0899	This error code indicates that general database exception occurred while auditing heartbeat.
ONRF-AUD-HBAUD-E0901	This error code indicates that an IO exception occurred while auditing heartbeat.
ONRF-AUD-HBAUD-E0902	This error code indicates that JSON processing exception occurred while auditing heartbeat.
ONRF-AUD-HBAUD-E0999	This error code indicates that any other exception occurred auditing heartbeat.

C.6.2 Remote Heart Beat for NRF Auditor

The following table lists the error codes and its problem details for the Remote Heart Beat for NRF Auditor service operation.

Table C-39 HBRMAUD - Remote Heart Beat for NRF Auditor

Error	Details
ONRF-AUD-HBRMAUD-E0899	This error code indicates that general database exception occurred while remote auditing heartbeat.
ONRF-AUD-HBRMAUD-E0901	This error code indicates that an IO exception occurred while remote auditing heartbeat.
ONRF-AUD-HBRMAUD-E0902	This error code indicates that JSON processing exception occurred while remote auditing heartbeat.
ONRF-AUD-HBRMAUD-E0999	This error code indicates that any other exception occurred while remote auditing heartbeat.

C.6.3 NRF Auditor Service Operation for NF Profiles

The following table lists the error codes and its problem details for the NRF Auditor microservice for the NRF Auditor service operation for NF Profiles.

Table C-40 NFPRAUD - NRF Auditor Service Operation for NF Profiles

Error Codes	Details
ONRF-AUD-NFPRAUD-E0899	This error code indicates that general database exception occurred while auditing NF profile.
ONRF-AUD-NFPRAUD-E0901	This error code indicates that IO exception occurred while auditing NF profile.
ONRF-AUD-NFPRAUD-E0902	This error code indicates that JSON processing exception occurred while auditing NF profile.

Copyright © 2019, 2025, Oracle and/or its affiliates.

C.6.4 Fetching NRF Configuration

The following table lists the error codes and its problem details for the NRF Auditor microservice for Fetching NRF Configuration.

Table C-41 NRFCFG - Fetching NRF Configuration

Error Codes	Details
ONRF-AUD-NRFCFG-E0802	This error code indicates that pessimistic locking exception occurred while fetching NRF configurations.
ONRF-AUD-NRFCFG-E0803	This error code indicates that data access resource failure exception occurred while fetching NRF configurations.
ONRF-AUD-NRFCFG-E0899	This error code indicates that general database exception occurred while fetching NRF configurations.

C.6.5 NRF Auditor Service Operation for Reloading Site and Network Version Info

The following table lists the error codes and its problem details for the NRF Auditor microservice for NRF Auditor service operation for reloading site and network version information.

Table C-42 RELDVER - NRF Auditor Service Operation for Reloading Site and Network Version Info

Error Codes	Details
ONRF-AUD-RELDVER-E0901	This error code indicates that an IO exception occurred while reloading site and network version info.

C.6.6 NRF Auditor Service Operation for Remote Subscription

The following table lists the error codes and its problem details for the NRF Auditor microservice for NRF Auditor service operation for remote subscription.

Table C-43 SUBRAUD - NRF Auditor Service Operation for Remote Subscription

Error Codes	Details
ONRF-AUD-SUBRAUD-E0899	This error code indicates that general database exception occurred while remote auditing subscription.

C.6.7 NRF Auditor Service Operation for Subscription

The following table lists the error codes and its problem details for the NRF Auditor microservice for NRF Auditor service operation for subscription.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-44 SUBAUD - NRF Auditor Service Operation for Subscription

Error Codes	Details
ONRF-AUD-SUBAUD-E0899	This error code indicates that general database exception occurred while auditing subscriptions.

C.7 NRF Artisan Microservice

This section lists the error codes and its problem details for the NRF Artisan microservice.

C.7.1 NRF Artisan for NAPTR Registration

The following table lists the error codes and its problem details for the NRF Artisan microservice for NRF Artisan for NAPTR registration.

Table C-45 NPTRREG - NRF Artisan for NAPTR Registration

Error Codes	Details
ONRF-ART-NPTRREG-E0001	This error code indicates that an invalid API was used when sending NAPTR registration.
ONRF-ART-NPTRREG-E0009	This error code indicates that a mandatory attribute is missing when sending NAPTR registration.

C.7.2 Status API for DNS NAPTR Record

The following table lists the error codes and its problem details for the NRF Artisan microservice for Status API for DNS NAPTR Record.

Table C-46 NPTRREC - Status API for DNS NAPTR Record

Error Details	Details
ONRF-ART-NPTRREC-E0001	This error code indicates that an invalid API was used when returning AMF DNS NAPTR records from cache.

C.7.3 NRF Artisan for Notification Event to DNS NAPTR on Deleting AMF Profile

The following table lists the error codes and its problem details for the NRF Artisan microservice for Notification Event to DNS NAPTR on deleting AMF Profile.

Table C-47 NPTRDRG - NRF Artisan for Notification Event to DNS NAPTR on Deleting AMF Profile

Error Codes	Details
ONRF-ART-NPTRDRG-E0001	This error code indicates that an invalid API was used when notification event to DNS NAPTR on delete AMF profile.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-47 (Cont.) NPTRDRG - NRF Artisan for Notification Event to DNS NAPTR on Deleting AMF Profile

Error Codes	Details
ONRF-ART-NPTRDRG-E0009	This error code indicates that a mandatory attribute is missing when notification event to DNS NAPTR on delete AMF profile.

C.7.4 NRF Artisan for DNS NAPTR Update

This section lists the error codes and its problem details for the NRF Artisan microservice for NRF Artisan for DNS NAPTR update.

Table C-48 NPTRUPD - NRF Artisan for DNS NAPTR Update

Error Code	Details
ONRF-ART-NPTRUPD-E0001	This error code indicates that an invalid API was used when notification event to DNS NAPTR on changed AMF profile.

C.7.5 Manual Cache Update of DNS NAPTR Records

The following table lists the error codes and its problem details for the NRF Artisan microservice for manual cache update of DNS NAPTR records.

Table C-49 RTNPTR - Manual Cache Update of DNS NAPTR Records

Error Code	Details
ONRF-ART-RTNPTR-E0001	This error code indicates that an invalid API was used when retriggering DNS NAPTR record update.
ONRF-ART-RTNPTR-E0002	This error code indicates that an invalid message format received when retriggering DNS NAPTR record update.
ONRF-ART-RTNPTR-E0100	This error code indicates that request is invalid when retriggering DNS NAPTR record update.

C.7.6 UnSupported Media Type

The following table lists the error codes and its problem details for the NRF Artisan microservice for UnSupported Media Type.

Table C-50 UNKSROP - UnSupported Media Type

Error Details	Details
ONRF-ART-UNKSROP-E0902	This error code indicates that JSON processing exception occurred during an unknown service operation.

Copyright © 2019, 2025, Oracle and/or its affiliates.

C.8 NRF Configuration Microservice

This section lists the error codes and its problem details for the NRF Configuration microservice.

C.8.1 NRF Configuration for General Options

The following table lists the error codes and its problem details for the NRF Configuration service operation for General Options category.

Table C-51 GENOPT - NRF Configuration General Options

Error Code	Details
ONRF-CFG-GENOPT-E0007	This error code indicates that request had mandatory IE incorrect during updates of General Options.
ONRF-CFG-GENOPT-E0008	This error code indicates that request had optional IE incorrect during updates of General Options.
ONRF-CFG-GENOPT-E0009	This error code indicates that request had mandatory IE missing during updates of General Options.
ONRF-CFG-GENOPT-E0010	The error code indicates that an unspecified message failure occurred while updating the General Options.
ONRF-CFG-GENOPT-E0021	This error code indicates that request had multiple attributes missing or incorrect during updates of General Options.
ONRF-CFG-GENOPT-E0100	This error code indicates that request is invalid for updating the configuration of General Options.
ONRF-CFG-GENOPT-E0802	This error code indicates that database exception occurred during updates of General Options.
ONRF-CFG-GENOPT-E0803	This error code indicates that data access resource failure exception occurred during updates of General Options.
ONRF-CFG-GENOPT-E0805	This error code indicates that NRF is unable to update the required data to database.
ONRF-CFG-GENOPT-E0807	This error code indicates that NRF unable to fetch the required data from database.
ONRF-CFG-GENOPT-E0899	This error code indicates that a general database exception occurred during updates of General Options.
ONRF-CFG-GENOPT-E0903	This error code indicates that an interrupted exception occurred during updates of General Options
ONRF-CFG-GENOPT-E0904	This error code indicates that class cast exception occurred during updates of General Options.
ONRF-CFG-GENOPT-E0999	This error code indicates that a general exception occurred during updates of General Options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

C.8.2 Screening Options Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for Screening Options Configuration.

Table C-52 SCRNOPT - Screening Options Configuration

Error Code	Details
ONRF-CFG-SCRNOPT-E0008	This error code indicates that request had optional IE incorrect during updates of Screening Options.
ONRF-CFG-SCRNOPT-E0009	This error code indicates that request had mandatory IE missing during updates of Screening Options.
ONRF-CFG-SCRNOPT-E0010	The error code indicates that an unspecified message failure occurred while updating the Screening Options.
ONRF-CFG-SCRNOPT-E0021	This error code indicates that request had multiple attributes missing or incorrect during updates of Screening Options
ONRF-CFG-SCRNOPT-E0802	This error code indicates that a database exception occurred during updates of Screening Options.
ONRF-CFG-SCRNOPT-E0803	This error code indicates that data access resource failure exception occurred during updates of Screening Options.
ONRF-CFG-SCRNOPT-E0805	This error code indicates that NRF is unable to update the required data regarding Screening Options to database.
ONRF-CFG-SCRNOPT-E0807	This error code indicates that NRF is unable to fetch the required data regarding Screening Options from database.
ONRF-CFG-SCRNOPT-E0899	This error code indicates that a general database exception occurred during updates of Screening Options.
ONRF-CFG-SCRNOPT-E0903	This error code indicates that an interrupted exception occurred during updates of Screening Options.
ONRF-CFG-SCRNOPT-E0904	This error code indicates that class cast Exception occurred during updates of Screening Options.
ONRF-CFG-SCRNOPT-E0999	This error code indicates that general exception occurred during updates of Screening Options.

C.8.3 Management Options Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for Management Options Configuration.

Table C-53 MGMTOPT - Management Options Configuration

Error Codes	Details
ONRF-CFG-MGMTOPT-E0007	This error code indicates that request had mandatory IE incorrect during updates of Management Options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-53 (Cont.) MGMTOPT - Management Options Configuration

Error Codes	Details
ONRF-CFG-MGMTOPT-E0008	This error code indicates that request had optional IE incorrect during updates of Management Options.
ONRF-CFG-MGMTOPT-E0009	This error code indicates that request had mandatory IE missing during updates of Management Options.
ONRF-CFG-MGMTOPT-E0010	The error code indicates that an unspecified message failure occurred while updating the Management Options.
ONRF-CFG-MGMTOPT-E0021	This error code indicates that request had multiple attributes missing or incorrect during updates of Management Options
ONRF-CFG-MGMTOPT-E0100	This error code indicates that request is invalid during updates of Management Options.
ONRF-CFG-MGMTOPT-E0802	This error code indicates that database exception occurred during updates of Management Options.
ONRF-CFG-MGMTOPT-E0803	This error code indicates that data access resource failure exception occurred during updates of Management Options.
ONRF-CFG-MGMTOPT-E0805	This error code indicates that NRF is unable to update the required data regarding Management Options to database.
ONRF-CFG-MGMTOPT-E0807	This error code indicates that NRF is unable to fetch the required data regarding Management Options from database.
ONRF-CFG-MGMTOPT-E0899	This error code indicates that general database exception occurred during updates of Management Options.
ONRF-CFG-MGMTOPT-E0903	This error code indicates that an interrupted exception occurred during updates of Management Options.
ONRF-CFG-MGMTOPT-E0904	This error code indicates that class cast exception occurred during updates of Management Options.
ONRF-CFG-MGMTOPT-E0999	This error code indicates that a general exception occurred during updates of Management Options.

C.8.4 Discovery Options Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for Discovery Options Configuration.

Table C-54 DISCOPT-Discovery Options Configuration

Error Code	Details
ONRF-CFG-DISCOPT-E0007	This error code indicates that request had mandatory IE incorrect during updates of Discovery Options.
ONRF-CFG-DISCOPT-E0008	This error code indicates that request had optional IE incorrect during updates of Discovery Options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-54 (Cont.) DISCOPT-Discovery Options Configuration

Error Code	Details
ONRF-CFG-DISCOPT-E0009	This error code indicates that request had mandatory IE missing during updates of Discovery Options.
ONRF-CFG-DISCOPT-E0010	The error code indicates that an unspecified message failure occurred while updating the Discovery Options.
ONRF-CFG-DISCOPT-E0021	This error code indicates that request had multiple attributes missing or incorrect during updates of Discovery Options
ONRF-CFG-DISCOPT-E0100	This error code indicates that request is invalid for updating the configuration of Discovery Options.
ONRF-CFG-DISCOPT-E0802	This error code indicates that database exception occurred during updates of Discovery Options.
ONRF-CFG-DISCOPT-E0803	This error code indicates that data access resource failure exception occurred during updates of Discovery Options.
ONRF-CFG-DISCOPT-E0805	This error code indicates that NRF is unable to update the required data regarding Discovery Options to database.
ONRF-CFG-DISCOPT-E0807	This error code indicates that NRF is unable to fetch the required data regarding Discovery Options from database.
ONRF-CFG-DISCOPT-E0899	This error code indicates that a general database exception occurred during updates of Discovery Options.
ONRF-CFG-DISCOPT-E0903	This error code indicates that an interrupted exception occurred during updates of Discovery Options.
ONRF-CFG-DISCOPT-E0904	This error code indicates that class cast exception occurred during updates of Discovery Options.
ONRF-CFG-DISCOPT-E0999	This error code indicates that general exception occurred during updates of Discovery Options.

C.8.5 Access Token Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for Access Token Configuration.

Table C-55 ACCOPT- Access Token Configuration

Error Code	Details
ONRF-CFG-ACCOPT-E0007	This error code indicates that request had mandatory IE incorrect during updates of AccessToken Options.
ONRF-CFG-ACCOPT-E0008	This error code indicates that request had optional IE incorrect during updates of AccessToken Options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-55 (Cont.) ACCOPT- Access Token Configuration

Error Code	Details
ONRF-CFG-ACCOPT-E0009	This error code indicates that request had mandatory IE missing during updates of AccessToken Options.
ONRF-CFG-ACCOPT-E0010	The error code indicates that an unspecified message failure occurred while updating the AccessToken Options.
ONRF-CFG-ACCOPT-E0021	This error code indicates that request had multiple attributes missing or incorrect during updates of AccessToken Options.
ONRF-CFG-ACCOPT-E0100	This error code indicates that request is invalid for updating the configuration of AccessToken Options.
ONRF-CFG-ACCOPT-E0802	This error code indicates that database exception occurred during updates of AccessToken Options.
ONRF-CFG-ACCOPT-E0803	This error code indicates that data access resource failure exception occurred during updates of AccessToken Options.
ONRF-CFG-ACCOPT-E0807	This error code indicates that NRF is unable to fetch the required data regarding AccessToken Options to database.
ONRF-CFG-ACCOPT-E0899	This error code indicates that general database exception occurred during updates of AccessToken Options.
ONRF-CFG-ACCOPT-E0903	This error code indicates that interrupted exception occurred during updates of AccessToken Options.
ONRF-CFG-ACCOPT-E0904	This error code indicates that class cast exception occurred during updates of AccessToken Options.
ONRF-CFG-ACCOPT-E0999	This error code indicates that general exception occurred during updates of AccessToken Options.

C.8.6 Forwarding Options Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for Forwarding Options Configuration.

Table C-56 FWDOPT - Forwarding Options Configuration

Error Code	Details
ONRF-CFG-FWDOPT-E0007	This error code indicates that request had mandatory IE incorrect during updates of Forwarding Options.
ONRF-CFG-FWDOPT-E0008	This error code indicates that request had optional IE incorrect during updates of Forwarding Options.
ONRF-CFG-FWDOPT-E0009	This error code indicates that request had mandatory IE missing during updates of Forwarding Options.
ONRF-CFG-FWDOPT-E0010	The error code indicates that an unspecified message failure occurred while processing the Forwarding Options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-56 (Cont.) FWDOPT - Forwarding Options Configuration

Error Code	Details
ONRF-CFG-FWDOPT-E0021	This error code indicates that request had multiple attributes missing or incorrect during updates of Forwarding Options.
ONRF-CFG-FWDOPT-E0100	This error code indicates that request is invalid for updating the configuration of Forwarding Options.
ONRF-CFG-FWDOPT-E0802	This error code indicates that database exception occurred during updates of Forwarding Options.
ONRF-CFG-FWDOPT-E0803	This error code indicates that data access resource failure exception occurred during updates of Forwarding Options.
ONRF-CFG-FWDOPT-E0805	This error code indicates that NRF is unable to update the required data regarding Forwarding Options to database.
ONRF-CFG-FWDOPT-E0807	This error code indicates that NRF is unable to fetch the required data regarding Forwarding Options from database.
ONRF-CFG-FWDOPT-E0899	This error code indicates that a general database exception occurred during updates of Forwarding Options.
ONRF-CFG-FWDOPT-E0903	This error code indicates that an interrupted exception occurred during updates of Forwarding Options.
ONRF-CFG-FWDOPT-E0904	This error code indicates that class cast exception occurred during updates of Forwarding Options.
ONRF-CFG-FWDOPT-E0999	This error code indicates that a general exception occurred during updates of Forwarding Options.

C.8.7 SLF Options Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for SLF Options Configuration.

Table C-57 SLFOPT - SLF Options Configuration

Error Code	Details
ONRF-CFG-SLFOPT-E0007	This error code indicates that request had mandatory IE incorrect during updates of SLF Options.
ONRF-CFG-SLFOPT-E0008	This error code indicates that request had optional IE incorrect during updates of SLF Options.
ONRF-CFG-SLFOPT-E0009	This error code indicates that request had mandatory IE missing during updates of SLF Options.
ONRF-CFG-SLFOPT-E0010	The error code indicates that an unspecified message failure occurred while processing the SLF Options.
ONRF-CFG-SLFOPT-E0021	This error code indicates that request had multiple attributes missing or incorrect during updates of SLF Options

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-57 (Cont.) SLFOPT - SLF Options Configuration

Error Code	Details
ONRF-CFG-SLFOPT-E0100	This error code indicates that request is invalid for updating the configuration of SLF Options.
ONRF-CFG-SLFOPT-E0802	This error code indicates that database exception occurred during updates of SLF Options.
ONRF-CFG-SLFOPT-E0803	This error code indicates that data access resource failure exception occurred during updates of SLF Options.
ONRF-CFG-SLFOPT-E0805	This error code indicates that NRF is unable to update the required data regarding SLF Options to database.
ONRF-CFG-SLFOPT-E0807	This error code indicates that NRF is unable to fetch the required data regarding SLF Options from database.
ONRF-CFG-SLFOPT-E0899	This error code indicates that a general database exception occurred during updates of SLF Options.
ONRF-CFG-SLFOPT-E0903	This error code indicates that an interrupted exception occurred during updates of SLF Options.
ONRF-CFG-SLFOPT-E0904	This error code indicates that class cast exception occurred during updates of SLF Options.
ONRF-CFG-SLFOPT-E0999	This error code indicates that a general exception occurred during updates of SLF Options.

C.8.8 GeoRedundancy Options Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for GeoRedundancy options configuration.

Table C-58 GROPT-GeoRedundancy Options Configuration

Error code	Details
ONRF-CFG-GROPT-E0007	This error code indicates that request had mandatory IE incorrect during updates of GeoRedundancy Options.
ONRF-CFG-GROPT-E0008	This error code indicates that request had optional IE incorrect during updates of GeoRedundancy Options.
ONRF-CFG-GROPT-E0009	This error code indicates that request had mandatory IE missing during updates of GeoRedundancy Options.
ONRF-CFG-GROPT-E0010	The error code indicates that an unspecified message failure occurred while updating the GeoRedundancy Options.
ONRF-CFG-GROPT-E0021	This error code indicates that request had multiple attributes missing or incorrect during updates of GeoRedundancy Options
ONRF-CFG-GROPT-E0100	This error code indicates that request is invalid for updating the configuration of GeoRedundancy Options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-58 (Cont.) GROPT-GeoRedundancy Options Configuration

Error code	Details
ONRF-CFG-GROPT-E0802	This error code indicates that database exception occurred during updates of GeoRedundancy Options.
ONRF-CFG-GROPT-E0803	This error code indicates that data access resource failure Exception occurred during updates of GeoRedundancy Options.
ONRF-CFG-GROPT-E0805	This error code indicates that NRF is unable to update the required data regarding GeoRedundancy Options to database.
ONRF-CFG-GROPT-E0807	This error code indicates that NRF is unable to fetch the required data regarding GeoRedundancy Options from database.
ONRF-CFG-GROPT-E0899	This error code indicates that general database exception occurred during updates of GeoRedundancy Options.
ONRF-CFG-GROPT-E0903	This error code indicates that an interrupted exception occurred during updates of GeoRedundancy Options.
ONRF-CFG-GROPT-E0904	This error code indicates that class cast exception occurred during updates of GeoRedundancy Options.
ONRF-CFG-GROPT-E0999	This error code indicates that a general exception occurred during updates of GeoRedundancy Options.

C.8.9 NAPTR Options Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for NAPTR Options configuration.

Table C-59 NPTROPT - NAPTR Options Configuration

Error Code	Details
ONRF-CFG-NPTROPT-E0007	This error code indicates that request had mandatory IE incorrect during updates of NAPTR Options.
ONRF-CFG-NPTROPT-E0008	This error code indicates that request had optional IE incorrect during updates of NAPTR Options.
ONRF-CFG-NPTROPT-E0009	This error code indicates that request had mandatory IE missing during updates of NAPTR Options.
ONRF-CFG-NPTROPT-E0010	The error code indicates that an unspecified message failure occurred while updating the NAPTR Options.
ONRF-CFG-NPTROPT-E0021	This error code indicates that request had multiple attributes missing or incorrect during updates of NAPTR Options.
ONRF-CFG-NPTROPT-E0100	This error code indicates that request is invalid for updating the configuration of NAPTR Options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-59 (Cont.) NPTROPT - NAPTR Options Configuration

Error Code	Details
ONRF-CFG-NPTROPT-E0802	This error code indicates that a database exception occurred during updates of NAPTR Options.
ONRF-CFG-NPTROPT-E0803	This error code indicates that data access resource failure exception occurred during updates of NAPTR Options.
ONRF-CFG-NPTROPT-E0805	This error code indicates that NRF is unable to update the required data regarding NAPTR Options to the database.
ONRF-CFG-NPTROPT-E0807	This error code indicates that NRF unable to fetch the required data regarding NAPTR Options from the database.
ONRF-CFG-NPTROPT-E0899	This error code indicates that a general database exception occurred during updates of NAPTR Options.
ONRF-CFG-NPTROPT-E0903	This error code indicates that an interrupted exception occurred during updates of NAPTR Options.
ONRF-CFG-NPTROPT-E0904	This error code indicates that class cast exception occurred during updates of NAPTR Options.
ONRF-CFG-NPTROPT-E0999	This error code indicates that a general exception occurred during updates of NAPTR Options.

C.8.10 Pod Protection Options Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for Pod Protection Options Configuration.

Table C-60 POPROPT - Pod Protection Options Configuration

Error Code	Details
ONRF-CFG-POPROPT-E0007	This error code indicates that request had mandatory IE incorrect during updates of pod protection.
ONRF-CFG-POPROPT-E0008	This error code indicates that request had optional IE incorrect during updates of pod protection.
ONRF-CFG-POPROPT-E0009	This error code indicates that request had mandatory IE missing during updates of pod protection.
ONRF-CFG-POPROPT-E0010	The error code indicates that an unspecified message failure occurred while updating the pod protection options.
ONRF-CFG-POPROPT-E0021	This error code indicates that request had multiple attributes missing or incorrect during updates of pod protection options.
ONRF-CFG-POPROPT-E0100	This error code indicates that request is invalid for updating the configuration of pod protection options.
ONRF-CFG-POPROPT-E0802	This error code indicates that database exception occurred during updates of pod protection options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-60 (Cont.) POPROPT - Pod Protection Options Configuration

Error Code	Details
ONRF-CFG-POPROPT-E0803	This error code indicates that data access resource failure exception occurred during updates of pod protection options.
ONRF-CFG-POPROPT-E0805	This error code indicates that NRF is unable to update the required data regarding pod protection options to database.
ONRF-CFG-POPROPT-E0807	This error code indicates that NRF is unable to fetch the required data regarding pod protection options from database.
ONRF-CFG-POPROPT-E0899	This error code indicates that a general database exception occurred during updates of pod protection options.
ONRF-CFG-POPROPT-E0903	This error code indicates that an interrupted exception occurred during updates of pod protection options.
ONRF-CFG-POPROPT-E0904	This error code indicates that class cast exception occurred during updates of pod protection options.
ONRF-CFG-POPROPT-E0999	This error code indicates that a general exception occurred during updates of pod protection options.

C.8.11 Authentication Options Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for Authentication Options Configuration.

Table C-61 AUTHOPT - Authentication Options Configuration

Error Code	Details
ONRF-CFG-AUTHOPT-E0007	This error code indicates that request had mandatory IE incorrect during updates of Authentication Options.
ONRF-CFG-AUTHOPT-E0008	This error code indicates that request had optional IE incorrect during updates of Authentication Options.
ONRF-CFG-AUTHOPT-E0009	This error code indicates that request had mandatory IE missing during updates of Authentication Options.
ONRF-CFG-AUTHOPT-E0010	The error code indicates that an unspecified message failure occurred while updating the Authentication Options.
ONRF-CFG-AUTHOPT-E0021	This error code indicates that request had multiple attributes missing or incorrect during updates of Authentication Options
ONRF-CFG-AUTHOPT-E0100	This error code indicates that request is invalid for updating the configuration of Authentication Options.
ONRF-CFG-AUTHOPT-E0802	This error code indicates that database exception occurred during updates of Authentication Options

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-61 (Cont.) AUTHOPT - Authentication Options Configuration

Error Code	Details
ONRF-CFG-AUTHOPT-E0803	This error code indicates that data access resource failure Exception occurred during updates of Authentication Options.
ONRF-CFG-AUTHOPT-E0805	This error code indicates that NRF is unable to update the required data regarding. Authentication Options to database.
ONRF-CFG-AUTHOPT-E0807	This error code indicates that NRF unable to fetch the required data regarding Authentication Options from database.
ONRF-CFG-AUTHOPT-E0899	This error code indicates that general database exception occurred during updates of Authentication Options.
ONRF-CFG-AUTHOPT-E0903	This error code indicates that Interrupted exception occurred during updates of Authentication Options.
ONRF-CFG-AUTHOPT-E0904	This error code indicates that class cast exception occurred during updates of Authentication Options.
ONRF-CFG-AUTHOPT-E0999	This error code indicates that general exception occurred during updates of Authentication Options.

C.8.12 Logging Options Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for Logging Options Configuration.

Table C-62 LOGOPT - Logging Options Configuration

Error Codes	Details
ONRF-CFG-LOGOPT-E0007	This error code indicates that request had mandatory IE incorrect during updates of Logging Options.
ONRF-CFG-LOGOPT-E0008	This error code indicates that request had optional IE incorrect during updates of Logging Options.
ONRF-CFG-LOGOPT-E0009	This error code indicates that request had mandatory IE missing during updates of Logging Options.
ONRF-CFG-LOGOPT-E0010	The error code indicates that an unspecified message failure occurred while updating Logging Options.
ONRF-CFG-LOGOPT-E0021	This error code indicates that request had multiple attributes missing or incorrect during updates of Logging Options.
ONRF-CFG-LOGOPT-E0100	This error code indicates that request is invalid for updating the configuration of Logging Options.
ONRF-CFG-LOGOPT-E0802	This error code indicates that database exception occurred during updates of Logging Options.
ONRF-CFG-LOGOPT-E0803	This error code indicates that data access resource failure Exception occurred during updates of Logging Options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-62 (Cont.) LOGOPT - Logging Options Configuration

Error Codes	Details
ONRF-CFG-LOGOPT-E0805	This error code indicates that NRF is unable to update the required data regarding Logging Options to database.
ONRF-CFG-LOGOPT-E0807	This error code indicates that NRF is unable to fetch the required data regarding Logging Options from database.
ONRF-CFG-LOGOPT-E0899	This error code indicates that general database exception occurred during updates of Logging Options.
ONRF-CFG-LOGOPT-E0902	This error code indicates that JSON processing exception occurred during updates of Logging Options.
ONRF-CFG-LOGOPT-E0903	This error code indicates that an interrupted exception occurred during updates of Logging Options.
ONRF-CFG-LOGOPT-E0904	This error code indicates that class cast exception occurred during updates of Logging Options.
ONRF-CFG-LOGOPT-E0999	This error code indicates that general exception occurred during updates of Logging Options.

C.8.13 Control Shutdown Options Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for Control Shutdown Options Configuration.

Table C-63 CSHTOPT - Control Shutdown Options Configuration

Error Code	Details
ONRF-CFG-CSHTOPT-E0007	This error code indicates that mandatory attribute is incorrect in request for configuring the controlled shutdown options.
ONRF-CFG-CSHTOPT-E0008	This error code indicates that optional attribute is incorrect in request for configuring the controlled shutdown options.
ONRF-CFG-CSHTOPT-E0009	This error code indicates that mandatory attribute is missing in request for configuring the controlled shutdown options.
ONRF-CFG-CSHTOPT-E0010	The error code indicates that an unspecified message failure occurred while updating the controlled shutdown options.
ONRF-CFG-CSHTOPT-E0020	This error code indicates that request timed out while configuring the controlled shutdown options.
ONRF-CFG-CSHTOPT-E0021	This error code indicates that request had multiple attributes missing or incorrect for configuring the controlled shutdown options.
ONRF-CFG-CSHTOPT-E0100	This error code indicates that request is invalid for configuring the controlled shutdown options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-63 (Cont.) CSHTOPT - Control Shutdown Options Configuration

Error Code	Details
ONRF-CFG-CSHTOPT-E0802	This error code indicates that pessimistic locking exception occurred while configuring the controlled shutdown options.
ONRF-CFG-CSHTOPT-E0803	This error code indicates that data access resource failure exception occurred while configuring the controlled shutdown options.
ONRF-CFG-CSHTOPT-E0805	This error code indicates that NRF is unable to update the required data for updating configuring the controlled shutdown options.
ONRF-CFG-CSHTOPT-E0807	This error code indicates that NRF unable to fetch the required data while fetching the controlled shutdown options.
ONRF-CFG-CSHTOPT-E0899	This error code indicates that general database exception occurred while accessing the controlled shutdown options.
ONRF-CFG-CSHTOPT-E0903	This error code indicates that an interrupted exception occurred while accessing the controlled shutdown options.
ONRF-CFG-CSHTOPT-E0904	This error code indicates that class cast exception occurred while accessing the controlled shutdown options.
ONRF-CFG-CSHTOPT-E0999	This error code indicates that a general exception occurred while accessing the controlled shutdown options.

C.8.14 Roaming Options Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for Roaming Options Configuration.

Table C-64 ROAMOPT - Roaming Options Configuration

Error Code	Details
ONRF-CFG-ROAMOPT-E0007	This error code indicates that mandatory IE is incorrect in request for configuring the roaming options.
ONRF-CFG-ROAMOPT-E0008	This error code indicates that an optional IE is incorrect in request for configuring the roaming options.
ONRF-CFG-ROAMOPT-E0009	This error code indicates that mandatory IE is missing in request for configuring the roaming options.
ONRF-CFG-ROAMOPT-E0010	The error code indicates that an unspecified message failure occurred while configuring the roaming options.
ONRF-CFG-ROAMOPT-E0020	This error code indicates that request timed out while configuring the roaming options.
ONRF-CFG-ROAMOPT-E0021	This error code indicates that request had multiple attributes missing or incorrect for configuring the roaming options.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-64 (Cont.) ROAMOPT - Roaming Options Configuration

Error Code	Details
ONRF-CFG-ROAMOPT-E0100	This error code indicates that request is invalid for configuring the roaming options.
ONRF-CFG-ROAMOPT-E0802	This error code indicates that pessimistic locking exception occurred while configuring the roaming options.
ONRF-CFG-ROAMOPT-E0803	This error code indicates that data access resource failure exception occurred while configuring the roaming options.
ONRF-CFG-ROAMOPT-E0805	This error code indicates that NRF is unable to update the required data for updating the roaming options.
ONRF-CFG-ROAMOPT-E0807	This error code indicates that NRF unable to fetch the required data while fetching the roaming options.
ONRF-CFG-ROAMOPT-E0899	This error code indicates that while general database exception occurred while accessing the roaming options.
ONRF-CFG-ROAMOPT-E0903	This error code indicates that an interrupted exception occurred while accessing the roaming options.
ONRF-CFG-ROAMOPT-E0904	This error code indicates that class cast exception occurred while accessing the roaming options.
ONRF-CFG-ROAMOPT-E0999	This error code indicates that General Exception occurred while accessing the roaming options.

C.8.15 Screening Rules Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for Screening Rules Configuration.

Table C-65 SCRNRUL - Screening Rules Configuration

Error Code	Details
ONRF-CFG-SCRNRUL-E0007	This error code indicates that request has mandatory IE Incorrect.
ONRF-CFG-SCRNRUL-E0008	This error code indicates that request has optional IE Incorrect.
ONRF-CFG-SCRNRUL-E0009	This error code indicates that request has mandatory IE missing.
ONRF-CFG-SCRNRUL-E0100	This error code indicates that request is invalid for updating the screening rules.
ONRF-CFG-SCRNRUL-E0802	This error code indicates that pessimistic locking exception occurred while updating the screening rules.
ONRF-CFG-SCRNRUL-E0803	This error code indicates that data access resource failure exception occurred for fetching the screening rules.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-65 (Cont.) SCRNRUL - Screening Rules Configuration

Error Code	Details
ONRF-CFG-SCRNRUL-E0805	This error code indicates that NRF is unable to update the required data for updating the screening rules.
ONRF-CFG-SCRNRUL-E0807	This error code indicates that NRF unable to fetch the required data fetching the screening rules.
ONRF-CFG-SCRNRUL-E0899	This error code indicates that general database exception occurred during accessing the screening rules.
ONRF-CFG-SCRNRUL-E0902	This error code indicates that an interrupted exception occurred while accessing the screening rules.
ONRF-CFG-SCRNRUL-E0903	This error code indicates that an interrupted exception occurred while accessing the screening rules.
ONRF-CFG-SCRNRUL-E0904	This error code indicates that class cast exception occurred during accessing the screening rules.
ONRF-CFG-SCRNRUL-E0999	This error code indicates that general exception occurred during accessing the screening rules.

C.8.16 Growth Options Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for Growth Options Configuration.

Table C-66 GWTHOPT - Growth Options Configuration

Error Code	Details
ONRF-CFG-GWTHOPT-E0007	This error code indicates that a mandatory attribute is incorrect for updating the Growth Options.
ONRF-CFG-GWTHOPT-E0008	This error code indicates that an optional attribute is incorrect for updating the Growth Options.
ONRF-CFG-GWTHOPT-E0009	This error code indicates that a mandatory attribute is missing for updating the Growth Options.
ONRF-CFG-GWTHOPT-E0010	The error code indicates that an unspecified message failure occurred while updating the Growth Options.
ONRF-CFG-GWTHOPT-E0021	This error code indicates that request had multiple attributes missing or incorrect for updating the Growth Options.
ONRF-CFG-GWTHOPT-E0100	This error code indicates that request is invalid for updating the Growth Options.

C.8.17 Growth Configuration for Forwarding

The following table lists the error codes and its problem details for the NRF Configuration service operation for Growth configuration for Forwarding.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-67 GWTHFOR - Growth Configuration for Forwarding

Error Code	Details
ONRF-CFG-GWTHFOR-E0007	This error code indicates that a mandatory attribute is incorrect for updating the Growth forwarding.
ONRF-CFG-GWTHFOR-E0008	This error code indicates that an optional attribute is incorrect for updating the Growth forwarding.
ONRF-CFG-GWTHFOR-E0009	This error code indicates that a mandatory attribute is missing for updating the Growth forwarding.
ONRF-CFG-GWTHFOR-E0010	The error code indicates that an unspecified message failure occurred while updating the Growth forwarding.
ONRF-CFG-GWTHFOR-E0021	This error code indicates that request had multiple attributes missing or incorrect for updating the Growth forwarding.
ONRF-CFG-GWTHFOR-E0100	This error code indicates that request is invalid for updating the Growth forwarding.

C.8.18 Fetching NRF Configuration

The following table lists the error codes and its problem details for fetching NRF Configuration.

Table C-68 NRFCFG - Fetching NRF Configuration

Error Code	Details
ONRF-CFG-NRFCFG-E0802	This error code indicates that pessimistic locking exception occurred while fetching the NRF Configurations.
ONRF-CFG-NRFCFG-E0803	This error code indicates that data access resource failure exception occurred while fetching the NRF Configurations.
ONRF-CFG-NRFCFG-E0807	This error code indicates that NRF is unable to fetch the required data while fetching the NRF Configurations.
ONRF-CFG-NRFCFG-E0899	This error code indicates that general database exception occurred while fetching the NRF Configurations.
ONRF-CFG-NRFCFG-E0903	This error code indicates that an interrupted exception occurred while fetching the NRF Configurations.
ONRF-CFG-NRFCFG-E0904	This error code indicates that class cast exception occurred while fetching the NRF Configurations.
ONRF-CFG-NRFCFG-E0999	This error code indicates that general exception occurred while fetching the NRF Configurations.

C.8.19 Fetching NF State Data

The following table lists the error codes and its problem details for the NRF Configuration service operation for Fetching NF State Data.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-69 NFDTLS - Fetching NF State Data

Error Code	Details
ONRF-CFG-NFDTLS-E0021	This error code indicates that request had multiple attributes missing or incorrect for fetching the NF details.
ONRF-CFG-NFDTLS-E0100	This error code indicates that request is invalid for fetching the NF details.
ONRF-CFG-NFDTLS-E0807	This error code indicates that NRF is unable to fetch the required data for the fetching NF details.
ONRF-CFG-NFDTLS-E0999	This error code indicates that general exception occurred while fetching the NF details.

C.8.20 Retrieve Subscription Details Based on Query Attributes for Configuring the Non-Signaling API

The following table lists the error codes and its problem details for retrieving subscription details based on query attributes for configuring the non-signaling API.

Table C-70 SUBDTLS - Retrieve Subscription Details Based on Query Attributes for Configuring the Non-Signaling API

Error Code	Details
ONRF-CFG-SUBDTLS-E0021	This error code indicates that request had multiple attributes missing or incorrect for fetching subscription details.
ONRF-CFG-SUBDTLS-E0807	This error code indicates that NRF is unable to fetch the required data for fetching subscription details.
ONRF-CFG-SUBDTLS-E0999	This error code indicates that general exception occurred during fetching subscription details.

C.8.21 Status API for DNS NAPTR Record

The following table lists the error codes and its problem details for the Status API for DNS NAPTR record.

Table C-71 NPTRREC- Status API for DNS NAPTR Record

Error Code	Details
ONRF-CFG-NPTRREC-E0807	This error code indicates that NRF is unable to fetch the required data from cache.

C.8.22 Manual Cache Update of DNS NAPTR Records

The following table lists the error codes and its problem details for the NRF Configuration Manual cache update of DNS NAPTR records.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-72 RTNPTR - Manual Cache Update of DNS NAPTR Records

Error Details	Details
ONRF-CFG-RTNPTR-E7000	This error code indicates that there was failure to trigger DNS NAPTR Update records.
ONRF-CFG-RTNPTR-E0902	This error code indicates that JSON processing exception occurred while triggering DNS NAPTR Update records.
ONRF-CFG-RTNPTR-E0100	This error code indicates that request is invalid for triggering DNS NAPTR Update records.
ONRF-CFG-RTNPTR-E0021	This error code indicates that request had multiple attributes missing or incorrect for triggering DNS NAPTR Update records.

C.8.23 NRF Operation State Configuration

The following table lists the error codes and its problem details for the NRF Operation State Configuration.

Table C-73 NRFOPER - NRF Operation State Configuration

Error Code	Details
ONRF-CFG-NRFOPER-E0020	This error code indicates that the request timed out when fetching NRF Configuration NRF operation state.
ONRF-CFG-NRFOPER-E0802	This error code indicates that pessimistic locking exception occurred when fetching NRF Configuration NRF operation state.
ONRF-CFG-NRFOPER-E0803	This error code indicates that data access resource failure exception occurred while fetching NRF Configuration NRF operation state.
ONRF-CFG-NRFOPER-E0805	This error code indicates that NRF is unable to update the required data for NRF Configuration NRF operation state.
ONRF-CFG-NRFOPER-E0807	This error code indicates that NRF is unable to fetch the required data while fetching NRF Configuration NRF operation state.
ONRF-CFG-NRFOPER-E0899	This error code indicates that general database exception occurred while fetching NRF Configuration NRF operation state.
ONRF-CFG-NRFOPER-E0903	This error code indicates that interrupted exception occurred while fetching NRF Configuration NRF operation state.
ONRF-CFG-NRFOPER-E0904	This error code indicates that class cast exception occurred while fetching NRF Configuration NRF operation state.
ONRF-CFG-NRFOPER-E0999	This error code indicates that general exception occurred while fetching NRF Configuration NRF operation state.

Copyright © 2019, 2025, Oracle and/or its affiliates.

C.8.24 CNCC Controller Configuration

The following table lists the error codes and its problem details for the NRF Configuration service operation for CNCC Controller Configuration

Table C-74 CNCCDTN - CNCC Controller Configuration

Error Code	Details
ONRF-CFG-CNCCDTN-E7001	This error code indicates that there is an error in CNCC controller.

C.8.25 UnSupported Media Type

The following table lists the error codes and its problem details for the NRF Configuration microservice for UnSupported Media Type.

Table C-75 UNKSROP - UnSupported Media Type

Error Details	Details
ONRF-CFG-UNKSROP-E0902	This error code indicates that JSON processing exception occurred during an unknown service operation.

C.9 Ingress Gateway Service

The following table lists the error codes and its problem details for the Ingress Gateway service.

Table C-76 Ingress Gateway Service Error Codes

Error Code	Description
ONRF-IGW-E001	This error code indicates that Ingress Gateway is unable to establish the connection with the backend service in the configured connectionTimeout period.
ONRF-IGW-E002	This error code indicates that request processing time is more than the configured request timeout at Ingress Gateway.
ONRF-IGW-E003	This error code indicates that Ingress Gateway is unable to resolve the configured backend service host name.
ONRF-IGW-E004	This error code indicates that Ingress Gateway is unable to establish the connection with the backend service.
ONRF-IGW-E005	This error code indicates that Blocklist IP exception has occurred while establishing a new connection with the blocklisted IP at Ingress Gateway. IP is blocklisted if the TCP connection to this IP has failed more than 10 times.
ONRF-IGW-E006	This error code indicates that closed channel exception has occurred because the incoming TCP connection was abruptly terminated before Ingress Gateway sent back the response.
ONRF-IGW-E051	This error code indicates that RejectedExecutionException has occurred at Ingress Gateway as the number of incoming requests is more than the queueCapacity configured in applicationThreadPoolConfig.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-76 (Cont.) Ingress Gateway Service Error Codes

Error Code	Description
ONRF-IGW-E052	This error code indicates that Ingress Gateway received the request path with no endpoint implemented in the backend service.
ONRF-IGW-E053	This error indicates that SocketException has occurred as Ingress Gateway was unable to create a TCP connection.
ONRF-IGW-E054	This error code indicates some unknown internal error occurred at Ingress Gateway while processing the request.
ONRF-IGW-E102	This error code indicates that late arrival exception occurred at Ingress Gateway as the request has already expired as part of the SbiTimer feature.
ONRF-IGW-E112	This error code indicates validation of the user-agent header failed at Ingress Gateway.
ONRF-IGW-E122	This error code indicates requests received at Ingress Gateway were rejected gracefully during the Ingress Gateway shutdown.
ONRF-IGW-E132	This error code indicates requests are being rejected after exceeding the burstCapacity when the global ratelimit feature is enabled.
ONRF-IGW-E152	This error code indicates that requests are being rejected after exceeding the burstCapacity when the rss ratelimit feature is enabled.
ONRF-IGW-E162	This error code indicates that validation of the XFCC header failed at Ingress Gateway.
ONRF-IGW-E163	This error code indicates that a request received at Ingress Gateway has no XFCC header or an empty XFCC header.
ONRF-IGW-E164	This error code indicates that the XFCC header validation failed due to XFCC certificate count mismatch in Ingress Gateway.
ONRF-IGW-E165	This error code indicates that an invalid XFCC header is received at Ingress Gateway.
ONRF-IGW-E171	This error code indicates that validation of the OAuth token failed at Ingress Gateway.
ONRF-IGW-E172	This error code indicates that the OAuth cert that has expired is received at Ingress Gateway.
ONRF-IGW-E173	This error code indicates that OAuth mismatch occurred in KID at Ingress Gateway.
ONRF-IGW-E174	This error code indicates that producer scope is absent in the OAuth token received at Ingress Gateway.
ONRF-IGW-E175	This error code indicates that the producer scope present in the OAuth token mismatches with the producer scope configured at Ingress Gateway.
ONRF-IGW-E176	This error code indicates that the NRF instance ID present in the OAuth token mismatches with the NRF instance ID configured at Ingress Gateway.
ONRF-IGW-E177	This error code indicates that the received OAuth token mismatches with the producer PLMN configured at Ingress Gateway.
ONRF-IGW-E178	This error code indicates that the received OAuth token does not contain an audience or an invalid audience at Ingress Gateway.
ONRF-IGW-E179	This error code indicates that an invalid OAuth token is received at Ingress Gateway.
ONRF-IGW-E180	This error code indicates that a request was received at Ingress Gateway with no OAuth token or an empty OAuth token.
ONRF-IGW-E182	This error code indicates that overload control exception occurred at Ingress Gateway.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-76 (Cont.) Ingress Gateway Service Error Codes

Error Code	Description
ONRF-IGW-E183	This error code indicates that ResponseStatusException was created by Spring boot as Ingress Gateway exception handler was unable to handle an exception.
ONRF-IGW-E191	This error code indicates that validation of the CCA header failed at Ingress Gateway.
ONRF-IGW-E192	This error code indicates that requests are being discarded partially at Ingress Gateway when the control shutdown feature is enabled.
ONRF-IGW-E193	This error code indicates that requests are being discarded completely at Ingress Gateway when the control shutdown feature is enabled.
ONRF-IGW-E200	This error code indicates all requests will be rejected at Ingress Gateway when the traffic reject feature is enabled.
ONRF-IGW-E201	This error code indicates that requests are being rejected when the load level matches the configured percentage in the discard policy mapping of the overload control feature.
ONRF-IGW-E202	This error code indicates that requests with sbi-priority header with value greater than the configured value of discarded load level matches discard policy mapping of the overload control feature.
ONRF-IGW-E211	This error code indicates that requests are being rejected when the priority level matches the configured priority in the discard policy mapping of the overload control feature.
ONRF-IGW-E220	This error code indicates that requests are being rejected after exceeding the burstCapacity when the PLMN Egress Rate Limit feature is enabled.
ONRF-IGW-E221	This error code indicates that an invalid route exception has occurred as invalidRouteFilter was configured in the routesConfiguration.

C.10 Egress Gateway Service

The following table lists the error codes and its problem details for the Egress Gateway service.

Table C-77 Egress Gateway Service Error Codes

Error Code	Description
ONRF-EGW-E001	This error code indicates that Egress Gateway is unable to establish the connection with peer NF in the configured connectionTimeout period.
ONRF-EGW-E002	This error code indicates that request processing time is more than the configured request timeout at Egress Gateway.
ONRF-EGW-E003	This error code indicates that Egress Gateway is unable to resolve the peer NF service host name.
ONRF-EGW-E004	This error code indicates that Egress Gateway is unable to establish the connection with peer NF service.
ONRF-EGW-E005	This error code indicates an exception occurred because HTTP1.1 is disabled, but the oc-http-version header has a value http1.
ONRF-EGW-E006	This error code indicates that a closed channel exception has occurred as the incoming TCP connection was abruptly terminated before Egress Gateway sent the response.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-77 (Cont.) Egress Gateway Service Error Codes

Error Code	Description
ONRF-EGW-E007	This error code indicates the SSL handshake exception occurred in Egress Gateway.
ONRF-EGW-E051	This error code indicates that RejectedExecutionException has occurred at Egress Gateway as the number of incoming requests is more than the queueCapacity configured in applicationThreadPoolConfig.
ONRF-EGW-E052	This error code indicates that Egress Gateway received a request path with an endpoint that is not implemented in the peer NF service.
ONRF-EGW-E054	This error code indicates that some unknown internal error occurred in Egress Gateway while processing the request.
ONRF-EGW-E102	This error code indicates that late arrival exception has occurred at Egress Gateway as the request has already expired as part of the SbiTimer feature.
ONRF-EGW-E112	This error code indicates that requests are being rejected after exceeding the bucketCapacity when the notificationRateLimit feature is enabled.
ONRF-EGW-E122	This error code indicates that the retry-after feature is enabled and producer FQDN is blocklisted, therefore the request cannot be fulfilled.
ONRF-EGW-E127	This error code indicates that the service shutdown is in progress.
ONRF-EGW-E132	This error code indicates that the DNS resolution of virtual FQDN failed.
ONRF-EGW-E152	This error code indicates the exception occurred because of a bad AccessToken request received.
ONRF-EGW-E153	This error code indicates that Egress Gateway is unable to send an AccessToken request to NRF.
ONRF-EGW-E154	This error code indicates an error in retrieving a token from NRF.
ONRF-EGW-E155	This error code indicates response failure from NRF(invalid/bad response) while trying to fetch oauth token.
ONRF-EGW-E172	This error code indicates that requests are being rejected after exceeding the bucketCapacity when the egressRateLimiting feature is enabled.
ONRF-EGW-E182	This error code indicates that custom exception occurred in some feature at Egress Gateway because some conditions failed to fulfill at runtime, such as SbiRouting, SeppProxyFilter, PeerBlackListHelper, EgressGatewayFilter, and so on.
ONRF-EGW-E183	This error code indicates that ResponseStatusException was created by Spring boot as Egress Gateway exception handler was unable to handle an exception.
ONRF-EGW-E192	This error code indicates that requests are being discarded partially at Egress Gateway when the control shutdown feature is enabled.
ONRF-EGW-E193	This error code indicates that requests are being discarded completely at Egress Gateway when the control shutdown feature is enabled.
ONRF-EGW-E202	This error code indicates that an invalid route exception has occurred as invalidRouteFilter was configured in routesConfiguration.

C.11 Alternate Route Service

The following table lists the error codes and its problem details for the Alternate Route Service.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table C-78 Alternate Route Service Error Codes

Error Code	Description
ARS-E101	This error code indicates that Alternate Route Service experienced an exception while trying to connect/send request to upstream DNS for the DNS NAPTR feature.
ARS-E102	This error code indicates FQDN or scheme parsing exception has occurred at Alternate Route Service.
ARS-E103	This error code indicates a custom exception has occurred at Alternate Route Service.

C.12 Common Configuration

The following table lists the error codes and its problem details for the Common Configuration.

Table C-79 Common Configuration Error Codes

Error Code	Description
CFG-E101	This error code indicates that an invalid method argument exception has occurred at Common Configuration Service.
CFG-E102	This error code indicates that the missing request parameter exception has occurred at Common Configuration Service.
CFG-E103	This error code indicates that no suitable converter was found for Common Configuration Service.
CFG-E105	This error code indicates that the triggered HTTP method is not supported for the URI/API end point.
CFG-E106	This error code indicates that exception has occurred as the payload was not in the proper format to read.
CFG-E107	This error code indicates that configuration related exception has occurred at Common Configuration Service.
CFG-E108	This error code indicates that HTTP error has occurred at Common Configuration Service.
CFG-E109	This error code indicates that custom exception has occurred at Common Configuration Service.

Copyright © 2019, 2025, Oracle and/or its affiliates.

D
Server Header Propagation

D.1 Adding Filters for Server Header Propagation

For service-mesh based deployment the below filter needs to be applied for the server header to be propagated so that the sidecar container will propagate the server header sent by the application container. Otherwise, it sends the server header as envoy.

The filter should be applied in each namespace for the server header propagation. Use the following command to apply this filter in each namespace:

```
kubectl apply -f server-header-envoy-filter.yaml --namespace $NAMESPACE
```

Where,

<namespace> is the namespace.

The content of the `server-header-envoy-filter.yaml` file is given below:

```
apiVersion: networking.istio.io/v1alpha3
kind: EnvoyFilter
metadata:
  name: serverheaderfilter
spec:
  configPatches:
  - applyTo: NETWORK_FILTER
    match:
      listener:
        filterChain:
          filter:
            name: envoy.filters.network.http_connection_manager
    patch:
      operation: MERGE
      value:
        typed_config:
          '@type': type.googleapis.com/
envoy.extensions.filters.network.http_connection_manager.v3.HttpConnectionMana
ger
          server_header_transformation: PASS_THROUGH
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

E

Use Cases for Allow Duplicate Subscriptions

This section describes the behavior of `allowDuplicateSubscriptions` parameter for different use cases. For more information, about configuring `allowDuplicateSubscriptions` parameter, see "NF Management Options" section in *Oracle Communications Cloud Native Core, Network Repository Function REST API Guide*.

Use Case 1: If the value of `allowDuplicateSubscriptions` is set as false and same subscription request than existing ones

NRF checks for duplicate subscription by matching the current subscription request with every subscription present in NRF except the `validityTime` attribute.

Existing subscription with NRF:

Subscription-1

```
{
"nfStatusNotificationUri":http://notification-url.oracle.com/subscribe/
notification,
"subscriptionId":"54d05ee9d3c042b0bf1499463cf53fe0",
"reqNotifEvents":["NF_REGISTERED","NF_PROFILE_CHANGED"],
"subscrCond":{"nfType" : "UDM"},
"validityTime":"2025-01-18T08:56:23.382Z"
}
```

New NFStatusSubscribe Request:

Subscription-2

```
{
"nfStatusNotificationUri":http://notification-url.oracle.com/subscribe/
notification,
"reqNotifEvents":["NF_REGISTERED","NF_PROFILE_CHANGED"],
"subscrCond":{"nfType":"UDM"},
"validityTime":"2025-06-12T08:56:23.382Z"
}
```

As per the above example, Subscription-2 matches exactly with Subscription-1 except with the `validityTime`, the subscription response contains the same `subscriptionId` and `validityTime` of Subscription-1.

Use Case 2: If the value of `allowDuplicateSubscriptions` value is set to false and different subscription request than existing ones.

NRF checks for duplicate subscription by matching the current subscription request with every subscription present in NRF except the `validityTime` attribute.

Existing Subscription with NRF:

Copyright © 2019, 2025, Oracle and/or its affiliates.

Subscription-1

```
{
"nfStatusNotificationUri":http://notification-url.oracle.com/subscribe/
notification,
"subscriptionId":"54d05ee9d3c042b0bf1499463cf53fe0",
"reqNotifEvents":["NF_REGISTERED","NF_PROFILE_CHANGED"],
"subscrCond":{"nfType" : "UDM"},
"validityTime":"2025-01-18T08:56:23.382Z"
}
```

New NFStatusSubscribe Request:

Subscription-3

```
{
"nfStatusNotificationUri":http://notification-url.oracle.com/subscribe/
notification-new,
"reqNotifEvents":["NF_REGISTERED","NF_PROFILE_CHANGED"],
"subscrCond":{"nfType" : "UDM"},
"validityTime":"2025-06-12T08:56:23.382Z"
}
```

As per the above example, Subscription-3 does not match with Subscription-1 as nfStatusNotificationUri is different. In this case, NRF creates a new subscription. Hence, subscription response returns the new `subscriptionId`.

Use Case 3:- If the value of `allowDuplicateSubscriptions` is set to false and different subscription request than existing ones

NRF checks for duplicate subscription by matching the current subscription request with every subscription present in NRF except the `validityTime` attribute.

Existing Subscription with NRF:

Subscription-1

```
{
"nfStatusNotificationUri":http://notification-url.oracle.com/subscribe/
notification,
"subscriptionId":"54d05ee9d3c042b0bf1499463cf53fe0",
"reqNotifEvents":["NF_REGISTERED","NF_PROFILE_CHANGED"],
"subscrCond":{"nfType" : "UDM"},
"validityTime":"2025-01-18T08:56:23.382Z"
}
```

New NFStatusSubscribe Request:

Subscription-3

```
{
"nfStatusNotificationUri":http://notification-url.oracle.com/subscribe/
```


Copyright © 2019, 2025, Oracle and/or its affiliates.

```
notification,
"reqNotifEvents":["NF_REGISTERED","NF_PROFILE_CHANGED"],
"subscrCond":{"nfType" : "AMF"},
"validityTime":"2025-06-12T08:56:23.382Z"
}
```

Subscription-3 does not match with Subscription-1 as the value of `subscrCond` is different. NRF creates a new subscription. Hence, subscription response returns the new `subscriptionId`.

Use Case-4:- When the value of `allowDuplicateSubscriptions` value is set to true and same subscription request than existing ones

NRF does not check for duplicate subscription and creates a new subscription.

Existing Subscription with NRF:

Subscription-1

```
{
"nfStatusNotificationUri":http://notification-url.oracle.com/subscribe/
notification,
"subscriptionId":"54d05ee9d3c042b0bf1499463cf53fe0",
"reqNotifEvents":["NF_REGISTERED","NF_PROFILE_CHANGED"],
"subscrCond":{"nfType" : "UDM"},
"validityTime":"2025-01-18T08:56:23.382Z"
}
```

New NFStatusSubscribe Request:-

Subscription-2

```
{
"nfStatusNotificationUri":http://notification-url.oracle.com/subscribe/
notification,
"subscriptionId":"54d05ee9d3c042b0bf1499463cf53fe0",
"reqNotifEvents":["NF_REGISTERED","NF_PROFILE_CHANGED"],
"subscrCond":{"nfType" : "UDM"},
"validityTime":"2025-06-12T08:56:23.382Z"
}
```

Subscription-2 matches exactly with Subscription-1 except `validityTime`, NRF creates a new subscription. Hence, subscription response returns the new `subscriptionId`.

Copyright © 2019, 2025, Oracle and/or its affiliates.

F

NRF Recommended Features

This section details the recommended features that are to be enabled mandatorily during the deployment of NRF at the production-level.

Table F-1 NRF Recommended Features

Feature Name	Sub Features	Overview
Limiting Number of NFProfiles in NFDiscover Response	NA	This feature controls the count of NFProfiles in the NFDiscover service operation response. For more information, see Limiting Number of NFProfiles in NFDiscover Response.
Network Policies	NA	This feature allows to specify how a pod is allowed to communicate with various network entities. For more information, see Network Policies. **Note**: Should be enabled.
Pod Protection Support for NRF Subscription Microservice	NA	NRF Subscription microservice is responsible for the NfStatusSubscribe, NfStatusUnsubscribe, and NfStatusNotify service operations. During processing of these service operations, Subscription microservice may get into congested or overload conditions. This feature protects the Subscription microservice from overload situations and maintains the overall health of the microservice. For more information, see Pod Protection Support for NRF Subscription Microservice.
Pre and Post Install/Upgrade Validations	NRF Infrastructure Validation	This feature helps in validating the infrastructure specific details such as NRF version, Kubernetes version, cnDBTier version, replication status, and alerts during NRF install and upgrade procedure. For more information, see NRF Infrastructure Validation.
Overload Control Based on Percentage Discards	Pod Level Traffic Rejections	NRF supports overload control to protect the nfregistration, nfdiscovery, nfsubscription, and nfaccesstoken services from overload situations and maintain the overall health of the services. This feature helps you to protect, mitigate, avoid entering into an overload condition, detect overload conditions, and take necessary actions to recover from overload. For more information, see Overload Control.

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table F-1 (Cont.) NRF Recommended Features

Feature Name	Sub Features	Overview
Subscription Limit	NA	NRF restricts the maximum number of allowed subscriptions to avoid the overload condition at NRF subscription microservice. NRF regulates the total number of allowed subscriptions using a configurable Global Subscription Limit. In case of georedundant NRFs, the limit is applied across all of the mated sites. For more information, see Subscription Limit.
Error Log Messages Enhancements	NA	NRF adds additional information to the existing ERR0R log messages. This additional information can provide more details about the error which can help to identify the problem details, error generating entity, and subscriber information. For more information, see Error Log Messages Enhancement.
User-Agent Header for Outgoing Requests	NA	NRF supports the addition of User-Agent header for outgoing messages such as NFStatusNotify and SLF query requests. If the User-Agent header value is configured, then NRF adds the User-Agent header with the configured value to the mentioned outgoing messages. For more information, see User-Agent Header for Outgoing Requests.
NRF Georedundancy	NA	NRF supports georedundancy to ensure high availability and redundancy. It offers a two, three, or four-sites georedundancy to ensure service availability when one of the NRF site is down. When NRF is deployed as georedundant site, all the sites work in an active state and the same data is available at all the sites. For more information, see NRF Georedundancy. **Note**: If the Georedundancy feature is enabled, NRF Last Known Response feature must be enabled. NRF Last Known Response: When NRF detects that the replication channel with its georedundant NRF(s) is down, it stops considering the NFs registered in the georedundant NRF(s) for any service operations. For more information, see NRF Last Known Response.
Controlled Shutdown of NRF	NA	NRF supports isolating an NRF from the current network at a particular site. This isolation helps to perform any maintenance activities or recovery procedures as required without uninstalling the NRF at the particular site. For more information, see Controlled Shutdown of NRF .

Copyright © 2019, 2025, Oracle and/or its affiliates.

Table F-1 (Cont.) NRF Recommended Features

Feature Name	Sub Features	Overview
Ingress Gateway Pod Protection Using Rate Limiting	NA	With this feature, the Ingress Gateway pods are protected by restricting the pods to process only a defined number of requests. The Ingress Gateway pods calculate the rate of the incoming requests and evaluate if it has breached the defined threshold. If breached, the pods reject the additional requests with the defined error code. If the rate of incoming requests has not breached the defined threshold, the pod allows the requests for further processing. For more information, see Ingress Gateway Pod Protection Using Rate Limiting.
Egress Gateway Pod Protection Using Rate Limiting	NA	With this feature, the Egress Gateway pods are protected by restricting the pods to process only a defined number of requests. The Egress Gateway pods calculate the rate of the incoming requests and evaluate if it has breached the defined threshold. If breached, the pods reject the additional requests with the defined error code. If the rate of incoming requests has not breached the defined threshold, the pod allows the requests for further processing. For more information, see Egress Gateway Pod Protection Using Rate Limiting.

Copyright © 2019, 2025, Oracle and/or its affiliates.