Oracle® Communications

Cloud Native Core, Network Repository Function Troubleshooting Guide

Release 24.2.6
F99732-07
October 2025

Oracle Communications Cloud Native Core, Network Repository Function Troubleshooting Guide, Release 24.2.6

F99732-07

Contents

Preface

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.

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.

Acronyms

The following table provides information about the acronyms and the terminology used in the document.

Table Acronyms

Term	Definition
3GPP	3rd Generation Partnership Project
5G-AN	5G Access Network
5GC	5G Core Network
5G System	3GPP system consisting of 5G Access Network (AN), 5G Core Network and UE
AMF	Access and Mobility Management Function
API Gateway	An API gateway is programming that sits in front of an API (Application Programming Interface) and is the single-entry point for a defined group of microservices
CNE	Cloud Native Environment
DD	Data Director
Dimension	Dimension is a tag of Metric. For example, "ocnrf_nfRegister_rx_requests_total {{ OriginatorNfType }} {{NrfLevel }} {{NfInstanceId }}" where the dimensions are: OriginatorNfType, NrfLevel, and NfInstanceId.
DNS	Domain Name System
FQDN	Fully Qualified Domain Name
K8s	Kubernetes
KPI	Key Performance Indicator
MMI	Machine Machine Interface
MPS	Messages Per Second
NDB	Network Database
NF	Network Function A functional building block within a network infrastructure, which has well defined external interfaces and defined functional behavior. In practical terms, a network function is often a network node or physical appliance.
Network Slice	A logical network that provides specific network capabilities and network characteristics.
Network Slice instance	A set of Network Function instances and the required resources. For example, compute, storage, and networking resources that form a deployed Network Slice.
NF Consumer	A generic way to refer to an NF which consumes services provided by another NF. For example, an AMF is referred to as a Consumer when it consumes AMPolicy services provided by the PCF.
NF Instance	A specific instance of a network function type.
NF Producer or NF Provider	A generic way to refer to an NF which provides services that can be consumed by another NF. For example, a PCF is a provider NF and provides AMPolicy Services

Table (Cont.) Acronyms

Term	Definition
NRF	Network Repository Function
PCF	Policy Control Function
PLMN	Public Land Mobile Network
Resiliency	The ability of the NFV framework to limit disruption and return to normal or at a minimum acceptable service delivery level in the fame of a fault, failure, or an event that disrupts normal operation.
SASL	Simple Authentication and Security Layer
Scaling	Ability to dynamically extend or reduce resources granted to the Virtual Network Function (VNF) as needed. This includes scaling out/in or scaling up/down.
Scaling Out/In/ Horizontally	The ability to scale by adding or removing resource instances, for example, VMs. It is also known as scaling Horizontally.
Scaling Up/Down/ Vertically	The ability to scale by changing allocated resources, for example, increase or decrease memory, CPU capacity, or storage size.
SCP	Service Communication Proxy
SEPP	Security Edge Protection Proxy
SLF	Subscriber Location Function
SMF	Session Management Function
URI	Uniform Resource Identifier

What's New in This Guide

This section lists the documentation updates for release 24.2.x.

Release 24.2.6 - F99732-07, October 2025

There are no changes made to this document in this release.

Release 24.2.5 - F99732-06, July 2025

There are no changes made to this document in this release.

Release 24.2.4 - F99732-05, April 2025

- Added the following alerts in the System Level Alerts section:
  - OcnrfTotalEgressTrafficRateAboveCriticalThreshold
  - OcnrfTotalForwardingTrafficRateAboveCriticalThreshold
  - OcnrfTotalSLFRateAboveCriticalThreshold
  - OcnrfTotalDiscoveryRateAboveCriticalThreshold
- Added the following alerts in the Service Level Alerts section:
  - OcnrfAccessTokenRequestsAboveThreshold
  - OcnrfNfUpdateRequestsAboveThreshold
  - OcnrfNfHeartBeatRequestsAboveThreshold
  - OcnrfRegisteredNfCountAboveThreshold
  - OcnrfNfProfileSizeAboveThreshold
  - OcnrfDiscoveryResponseSizeAboveThreshold
  - OcnrfTotalSubscriptionsAboveThreshold
  - OcnrfDiscoveryRequestsForUDRAboveThreshold
  - OcnrfDiscoveryRequestsForUDMAboveThreshold
  - OcnrfDiscoveryRequestsForAMFAboveThreshold
  - OcnrfDiscoveryRequestsForSMFAboveThreshold
- Updated the critical configured threshold for the OcnrfTotalIngressTrafficRateAboveCriticalThreshold alert.
- Added the following alerts for the Egress Gateway Pod Throttling" feature in the Egress Gateway Pod Throttling section:
  - OcnrfEgressPerPodDiscardRateAboveMajorThreshold
  - OcnrfEgressPerPodDiscardRateAboveCriticalThreshold

Release 24.2.3 - F99732-04, January 2025

There are no updates made to this document in this release.

Release 24.2.2 - F99732-03, October 2024

- Added the troubleshooting scenarios for Egress Connection Failure due to cipher mismatch and the expired certificates in the TLS Connection Failure section.

- Updated the Incorrect image name in ocnrf-custom-values files section as `--purge` command is not supported in Helm3 deployment.

Release 24.2.1 - F99732-02, September 2024

Added the description for the alert severity types in the NRF Alerts section.

Release 24.2.0 - F99732-01, July 2024

- Added the following alerts for the **Support for TLS 1.3** feature.
  - OcnrfTLSCertificateExpireMinor
  - OcnrfTLSCertificateExpireMajor
  - OcnrfTLSCertificateExpireCritical
- Added the TLS Connection Failure section for the **Support for TLS 1.3** feature.
- Updated the Understanding Logs section with the following for the **Error Log Messages Enhancements** feature:
  - Added the errorStatus, errorTitle, errorDetails, errorCause, sender, receiver, and subscriberId attributes to the Log Attributes Details table.
  - Added the sample ERROR log statement for the following microservices:
    - nfregistration
    - nfdiscovery
    - nfsubscription
    - nfaccesstoken
    - nrfconfiguration
- Removed the ProblemDetails and httpStatusCode attributes from the Understanding Logs section.
- Updated the description of the OcnrfMaxSlfAttemptsExhausted alert for **Rerouting SLF Requests using Alternate SCP and Alternate SLF** feature.

1 Introduction

This document provides information about troubleshooting Oracle Communications Cloud Native Core, Network Repository Function (NRF).

1.1 Overview

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.

This guide provides extensive information about resolving problems you might experience while installing and configuring NRF. It also contains information about tools available to help you collect and analyze diagnostic data.

> **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.

1.2 Reference

Following are the reference documents:

- *Oracle Communications Cloud Native Core, Cloud Native Environment Installation and Upgrade Guide*
- *Oracle Communications Cloud Native Core, cnDBTier Disaster Recovery Guide*
- *Oracle Communications Cloud Native Core, Network Repository Function User Guide*
- *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*
- *Oracle Communications Cloud Native Core, Data Collector Guide*

2
Logs

This chapter explains the process to retrieve the logs and status that can be used for effective troubleshooting. The Oracle Communications Cloud Native Core, Network Repository Function (NRF) provides various sources of information that may be helpful in the troubleshooting process.

2.1 Log Levels

Logs register system events along with their date and time of occurrence. They also provide important details about a chain of events that could have led to an error or problem.

Supported Log Levels

For NRF, the log level for a microservice can be set to any of the following valid values:

- **TRACE**: A log level describing events showing step by step execution of your code that can be ignored during the standard operation, but may be useful during extended debugging sessions.
- **DEBUG**: A log level used for events considered to be useful during software debugging when more granular information is needed.
- **INFO**: The standard log level indicating that something happened, the application entered a certain state, etc.
- **WARN**: Indicates that something unexpected happened in the application, a problem, or a situation that might disturb one of the processes. But that doesn't mean that the application failed. The WARN level should be used in situations that are unexpected, but the code can continue the work.
- **ERROR**: The log level that should be used when the application hits an issue preventing one or more functionalities from properly functioning.

Configuring Log Levels

To view logging configurations and update logging levels, use the Logging Level page under Logging Level Options on the Cloud Native Core (CNC) Console. For more information, see the section "Logging Level Options" in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

2.2 Collecting Logs

This section describes steps to collect the logs from PODs or containers.

1. Run the following command to get the POD details:

   ```
   $ kubectl -n <namespace_name> get pods
   ```

2. Collect the logs from the specific pods or containers:

   ```
   $ kubectl logs <podname> -n <namespace>
   ```

Example:

```
$ kubectl logs ocnrf-nfaccesstoken-xxxxxxxxxx-xxxxx -n ocnrf
```

3. Store the log in a file using the following command:

```
$ kubectl logs <podname> -n <namespace>  > <filename>
```

Example:

```
$ kubectl logs ocnrf-nfaccesstoken-xxxxxxxxxx-xxxxx -n ocnrf > logs.txt
```

4. (Optional) You can also use the following commands for the log stream with file redirection starting with last 100 lines of log:

```
$ kubectl logs <podname> -n <namespace> -f --tail <number of lines> >
<filename>
```

Example:

```
$ kubectl logs ocnrf-nfaccesstoken-xxxxxxxxxx-xxxxx -n ocnrf -f --tail 100
> logs.txt
```

For more information on how to collect the logs, see *Oracle Communications Cloud Native Core Data Collector Guide*.

2.3 Understanding Logs

This chapter explains the logs you need to look into, to handle different NRF debugging issues.

This section provides log level attribute details for following services:

- nfregistration
- nfdiscovery

Sample Logs

Sample log statement nfregistration:

```
{"instant":
{"epochSecond":1604654859,"nanoOfSecond":278610000},"thread":"XNIO-1
task-1","level":"WARN","loggerName":"com.oracle.cgbu.cne.nrf.service.NfMgmtSer
viceImpl","message":"NF status changed for nfInstanceId 8faf1bbc-5324-4454-
a507-a14ef8e1bc2c, whose previous status was SUSPENDED and current status is
REGISTERED.","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.Abs
tractLogger","nfInstanceID": "8faf1bbc-5324-4454-a507-
a14ef8e1bc2c","threadId":49,"threadPriority":5,"messageTimestamp":"2020-11-06T
09:27:39.278+0000","configuredLevel":"INFO","serviceOperation":"nfRegister","r
equesterNfType": "UDM","nfFqdn": "UNKNOWN","processId":"1","nrfTxId":"nrf-
tx-226718429","ocLogId":"1604654852868_56_ocnrf-ingressgateway-59548646fc-
g7qtk:1604654859244_49_ocnrf-nfregistration-65b999468b-gchg8"}
```

Sample ERROR log statement for nfregistration when Error Log Messages Enhancement feature is enabled:


```
{"instant":
{"epochSecond":1717082050,"nanoOfSecond":263756204},"thread":"XNIO-1

task-2","level":"ERROR","loggerName":"com.oracle.cgbu.cne.nrf.rest.NFManagemen
tController","message":"Response
                sent: 400 Bad Request for uri :
                http://ocnrf-ingressgateway.nrf1-ns/nnrf-nfm/v1/nf-instances/
fcff26ea-6c8d-4502-968b-982532f48dcf
                with the problem cause : MANDATORY_IE_INCORRECT and problem
details :
                NRF-d5g.oracle.com: Nnrf_NFManagement: Bad Request:
                ONRF-REG-REGN-
E0007","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.AbstractL
ogger","threadId":76,"threadPriority":5,"messageTimestamp":"2024-05-30T15:14:1
0.263+0000","configuredLevel":"WARN","processId":"1","nrfTxId":"nrf-
tx-1868521809","ocLogId":"1717082050214_77_ocnrf-
ingressgateway-854464d548-426bz:1717082050236_76_ocnrf-nfregistration-
d5d7bb9ff-
gdlct","xRequestId":"","nfInstanceID":"fcff26ea-6c8d-4502-968b-982532f48dcf","
requesterNfType":"AMF","nfFqdn":"UNKNOWN","httpMethod":"PUT","serviceOperation
":"nfRegister","featureStatus":"","requestUrl":"","hostname":"ocnrf-
nfregistration-d5d7bb9ff-
gdlct","subsystem":"","errorStatus":"400","errorTitle":"Bad
                Request","errorDetails":"NRF-d5g.oracle.com:
Nnrf_NFManagement: Bad Request:
                ONRF-REG-REGN-
E0007","errorCause":"MANDATORY_IE_INCORRECT","sender":"NRF-6faf1bbc-6e4a-4454-
a507-a14ef8e1bc5c","subscriberId":"imsi-345012123123125"}
```


Sample log statement nfdiscovery:


```
{"instant":
{"epochSecond":1604655402,"nanoOfSecond":946649000},"thread":"XNIO-1
task-1","level":"INFO","loggerName":"com.oracle.cgbu.cne.nrf.rest.NFDiscoveryC
ontroller","message":"Request received with uri http://10.75.226.148:30547/
nnrf-disc/v1/nf-instances and http-method
GET"}","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.AbstractL
ogger","threadId":47,"threadPriority":5,"messageTimestamp":"2020-11-06T09:36:4
2.946+0000","configuredLevel":"INFO","serviceOperation":"NFDiscover","processI
d":"1","nrfTxId":"nrf-tx-1782432001","ocLogId":"1604655396595_56_ocnrf-
ingressgateway-59548646fc-g7qtk:1604655402932_47_ocnrf-nfdiscovery-6df9dbb6bb-
kzbgl"}
```


Sample ERROR log statement for nfdiscovery when Error Log Messages Enhancement feature is enabled:


```
{"instant":
{"epochSecond":1717082579,"nanoOfSecond":276265496},"thread":"discovery-h2c-
epoll-2","level":"ERROR","loggerName":"com.oracle.cgbu.cne.nrf.rest.NFDiscover
yController","message":"Response
      sent: 400 Bad Request for uri : /nnrf-disc/v1/nf-instances?{target-nf-
type=[AMF],
```

```
      requester-nf-type=[ABC]} with the problem cause :
MANDATORY_QUERY_PARAM_INCORRECT and problem
      details : NRF-d5g.oracle.com: Nnrf_NFDiscovery: requesterNfType should
be a valid value:
      ONRF-DIS-DISC-
E0004","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.AbstractL
ogger","threadId":5286,"threadPriority":5,"messageTimestamp":"2024-05-30T15:22
:59.276+0000","configuredLevel":"WARN","processId":"1","nrfTxId":"nrf-
tx--1469395440","ocLogId":"1717082579244_250_ocnrf-
ingressgateway-854464d548-426bz:1717082579263_5286_ocnrf-
nfdiscovery-76f946759c-
gg7lg","serviceOperation":"NFDiscover","xRequestId":"","requesterNfType":"ABC"
,"targetNfType":"AMF","discoveryQuery":"/nnrf-disc/v1/nf-instances?{target-nf-
type=[AMF],
      requester-nf-type=[ABC]}","hostname":"ocnrf-nfdiscovery-76f946759c-
gg7lg","subsystem":"","errorStatus":"400","errorTitle":"Bad
      Request","errorDetails":"NRF-d5g.oracle.com: Nnrf_NFDiscovery:
requesterNfType should be a
      valid value:
      ONRF-DIS-DISC-
E0004","errorCause":"MANDATORY_QUERY_PARAM_INCORRECT","sender":"NRF-6faf1bbc-6
e4a-4454-a507-a14ef8e1bc5c","subscriberId":"imsi-345012123123126"}
```


Table 2-1 Log Attribute Details

Log Attribute	Details	Sample Value	Data Type
instant	Epoch time **Note**: It is group of two values epochSecond and nanoOfSecond	{"epochSecond":1604655402,"nanoOfSecond":946649000}	Object
thread	Logging Thread Name	"XNIO-1 task-1"	String
level	Log Level of the log printed	"WARN"	String
loggerName	Class or Module which printed the log	"com.oracle.cgbu.cne.nrf.rest.NFDiscoveryController"	String
message	Message related to the log providing brief details **Note**: Indicates that no NFProfiles found for mentioned search query	"{Discovery Query=target-nf-type=AMF&requester-nf-type=UDM, logMsg=No NFProfiles found for query}"	String
endOfBatch	Log4j2 Internal Default from log4j2: false	false	boolean
loggerFqcn	Log4j2 Internal Fully Qualified class name of logger module	org.apache.logging.log4j.spi.AbstractLogger	String
threadId	Thread Id generated internally by Log4j2	47	Integer
threadPriority	Thread Priority set internally by Log4j2	5	Integer
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ	"2020-11-06T09:36:42.946+0000"	String
configuredLevel	Log Level configured	"INFO"	String

Table 2-1 (Cont.) Log Attribute Details

Log Attribute	Details	Sample Value	Data Type
serviceOperation	Indicates the service operation	"NFDiscover"	String
processId	Process ID internally assigned	"1"	String
nrfTxId	Internal unique number for each log message to correlate the logs inside microservice	"nrf-tx-1782432001"	String
ocLogId	End to End Log Identifier across the NRF microservices **Note**: Helps to correlate the logs across the microservices in NRF application	"ocLogId":"1604655396595_56_ocnrf-ingressgateway-59548646fc-g7qtk :1604655402932_47_ocnrf-nfdiscovery-6df9dbb6bb-kzbgl"	String
nfInstanceID	Provide the NF Instance Id details		String
requesterNfType	For Registration request, this indicates the message originator fetched from NF Type in the User Agent HeaderFor Discovery query, this indicates the Requester NF Type		String
targetNfType	Indicates the Discovery Query Target NF Type		String
nfFqdn	Value of NF FQDN. This is extracted from XFCC header.		String
httpMethod	Indicates the httpMethod. For example: PUT, GET, PATCH		String
requestUrl	Indicates the URL of the message		String
discoveryQuery	This is populated in the log for response message in case of response for NFDiscover service operation.		String
errorStatus	Indicates the status sent or received by NRF in ProblemDetails of HTTP response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	400	String
errorTitle	Indicates the title sent or received by NRF in ProblemDetails of HTTP response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	Bad Request	String

Table 2-1 (Cont.) Log Attribute Details

Log Attribute	Details	Sample Value	Data Type
errorDetails	Indicates the detail sent or received by NRF in ProblemDetails of HTTP response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	NRF-d5g.oracle.com: Nnrf_NFDiscovery: requesterNfType should be a valid value: ONRF-DIS-DISC-E0004	String
errorCause	Indicates the cause sent or received by NRF in ProblemDetails of HTTP response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	MANDATORY_QUERY_PARAM_INCORRECT	String
sender	Indicates the NRF-NRF Instance Id or Server header value as received in the error response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	NRF-6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c	String
receiver	Indicates the NRF-NRF Instance Id **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.		String
subscriberId	Indicates the • 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. **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	imsi-345012123123126	String

This section provides log level attribute details for following services:

- nfsubscription
- nfaccesstoken

Sample Logs

Sample log statement nfsubscription:


```
{"instant":
{"epochSecond":1604654864,"nanoOfSecond":237018000},"thread":"XNIO-1
task-1","level":"ERROR","loggerName":"com.oracle.cgbu.cne.nrf.service.NfSubsSe
rviceImpl","message":"{logMsg=No subscriptions found matching the
notification for Profile, nfProfile={\"nfInstanceId\":\"dd94d4b5-0ce6-4571-
a661-3ff7c21a79a7\",\"nfType\":\"CUSTOM_SPF\",\"nfStatus\":\"REGISTERED\",\"he
artBeatTimer\":30,\"fqdn\":\"UPF.d5g.oracle.com\",\"interPlmnFqdn\":\"UPF-
d5g.oracle.com\",\"ipv4Addresses\":
[\"192.168.2.100\",\"192.168.3.100\",\"192.168.2.110\",\"192.168.3.110\"],\"ip
v6Addresses\":
[\"2001:0db8:85a3:0000:0000:8a2e:0370:7334\"],\"additionalAttributes\":
{\"recordCreator\":\"6faf1bbc-6e4a-4454-a507-
a14ef8e1bc5c\",\"creationTimestamp\":1604654861984000,\"requesterNfFqdn\":null
,\"lastUpdateTimestamp\":1604654861984000},\"customInfo\":
{\"key1\":\"value1\",\"key2\":\"value2\"}},
eventType=NF_DEREGISTERED}","endOfBatch":false,"loggerFqcn":"org.apache.loggin
g.log4j.spi.AbstractLogger","threadId":49,"threadPriority":5,"messageTimestamp
":"2020-11-06T09:27:44.237+0000","configuredLevel":"INFO","subsystem":"notifyP
rofileDeregistration","processId":"1","nrfTxId":"nrf-
tx-2125219011","ocLogId":"1604654857899_56_ocnrf-ingressgateway-59548646fc-
g7qtk:1604654864210_49_ocnrf-nfregistration-65b999468b-gchg8"}
```


Sample ERROR log statement for nfsubscription when Error Log Messages Enhancement feature is enabled:


```
{"instant":
{"epochSecond":1717077558,"nanoOfSecond":256932930},"thread":"boundedElastic-8
","level":"ERROR","loggerName":"com.oracle.cgbu.cne.nrf.routes.SubscriptionHan
dler","message":"Response
                sent: 500 Subscription global limit breached for uri :
                http://ocnrf-ingressgateway.madhu-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.AbstractL
ogger","threadId":39547,"threadPriority":5,"messageTimestamp":"2024-05-30T13:5
9:18.256+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"}
```


Sample log statement nfaccesstoken:


```
{"instant":
{"epochSecond":1604512030,"nanoOfSecond":714594000},"thread":"XNIO-1
task-1","level":"INFO","loggerName":"com.oracle.cgbu.cne.nrf.service.AccessTok
enServiceImpl","message":"{logMsg=Successfully generated
token}","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.Abstract
Logger","threadId":53,"threadPriority":5,"messageTimeStamp":"20-11-0417:47:10.
714+0000","configuredlevel":"INFO","subsystem":"accessToken","processId":"1","
nrfTxId":"nrf-tx-1390815064","ocLogId":"1604512030507_73_ocnrf-
ingressgateway-7cbcff6b47-9qw9b:1604512030525_53_ocnrf-
nfaccesstoken-6bddf44c66-86lmf"}
```


Sample ERROR log statement for nfaccesstoken when Error Log Messages Enhancement feature is enabled:


```
{"instant":
{"epochSecond":1717078943,"nanoOfSecond":335260003},"thread":"XNIO-1

task-2","level":"ERROR","loggerName":"com.oracle.cgbu.cne.nrf.rest.NFAccessTok
enController","message":"Response
                sent: 400 Bad Request for uri :   http://ocnrf-
ingressgateway.madhu-ns/oauth2/token
                with the problem cause :   Bad Request and problem details :
Bad

Request","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.Abstrac
tLogger","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"}
```


Table 2-2 Log Attribute Details

Log Attribute	Details	Sample Value	Data Type
instant	Epoch time **Note**: It is group of two values epochSecond and nanoOfSecond	{"epochSecond":1604655402," nanoOfSecond":946649000}	Object
thread	Logging Thread Name	"XNIO-1 task-1"	String
level	Log Level of the log printed	"WARN"	String

Table 2-2 (Cont.) Log Attribute Details

Log Attribute	Details	Sample Value	Data Type
loggerName	Class or Module which printed the log	"com.oracle.cgbu.cne.nrf.rest.NFDiscoveryController"	String
message	Message related to the log providing brief details **Note**: Indicates that no NFProfiles found for mentioned search query	"{Discovery Query=target-nf-type=AMF&requester-nf-type=UDM, logMsg=No NFProfiles found for query}"	String
endOfBatch	Log4j2 Internal Default from log4j2: false	false	boolean
loggerFqcn	Log4j2 Internal Fully Qualified class name of logger module	org.apache.logging.log4j.spi.AbstractLogger	String
threadId	Thread Id generated internally by Log4j2	47	Integer
threadPriority	Thread Priority set internally by Log4j2	5	Integer
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ	"2020-11-06T09:36:42.946+0000"	String
configuredLevel	Log Level configured	"INFO"	String
subsystem	Subsystem inside microservice internal to microservice	discoveryNfInstances	String
processId	Process ID internally assigned	"1"	String
nrfTxId	Internal unique number for each log message to correlate the logs inside microservice	"nrf-tx-1782432001"	String
ocLogId	End to End Log Identifier across the NRF microservices **Note**: Helps to correlate the logs across the microservices in NRF application	"ocLogId":"1604655396595_56_ocnrf-ingressgateway-59548646fc-g7qtk :1604655402932_47_ocnrf-nfdiscovery-6df9dbb6bb-kzbgl"	String
errorStatus	Indicates the status sent or received by NRF in ProblemDetails of HTTP response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	400	String
errorTitle	Indicates the title sent or received by NRF in ProblemDetails of HTTP response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	Bad Request	String

Table 2-2 (Cont.) Log Attribute Details

Log Attribute	Details	Sample Value	Data Type
errorDetails	Indicates the detail sent or received by NRF in ProblemDetails of HTTP response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	NRF-d5g.oracle.com: Nnrf_AccessToken: NfType in oAuth request is different from the registered profile: ONRF-ACC-ACTOK-E4004	String
errorCause	Indicates the cause sent or received by NRF in ProblemDetails of HTTP response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	unauthorized_client	String
sender	Indicates the NRF-NRF Instance Id or Server header value as received in the error response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	NRF-6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c	String
receiver	Indicates the NRF-NRF Instance Id **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.		String
subscriberId	Indicates the • 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. **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	imsi-345012123123129	String

This section provides log level attribute details for following services:

- nrfauditor
- nrfconfiguration

Sample log statement nrfauditor:


```
{"instant":
{"epochSecond":1604654345,"nanoOfSecond":125124000},"thread":"main","level":"I
NFO","loggerName":"com.oracle.cgbu.cne.nrf.audit.AuditManager","message":"{log
Msg=Started NF Profile Auditor Thread, Id=25, Name=NF Profile
Auditor}","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.Abstra
ctLogger","threadId":1,"threadPriority":5,"messageTimestamp":"2020-11-06T09:19
:05.125+0000","configuredLevel":"INFO","subsystem":"AuditManager","processId":
"1","nrfTxId":"nrf-tx-1930793990"}
```


Sample log statement nrfconfiguration:


```
{"instant":
{"epochSecond":1604654450,"nanoOfSecond":117406000},"thread":"XNIO-1
task-1","level":"INFO","loggerName":"com.oracle.cgbu.cne.nrf.rest.NrfConfigura
tionController","message":"{logMsg=Update GeneralOptions request received
with generalOptions,
generalOptions={\"additionalAttributes\":null,\"nrfPlmnList\":
[{\"mcc\":\"310\",\"mnc\":\"14\"}],\"enableF3\":null,\"enableF5\":null,\"maxim
umHopCount\":null,\"defaultLoad\":null,\"defaultPriority\":null,\"defaultPrior
ityAssignment\":null,\"defaultLoadAssignment\":null,\"ocnrfHost\":null,\"ocnrf
Port\":null}}","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.A
bstractLogger","threadId":34,"threadPriority":5,"messageTimestamp":"2020-11-06
T09:20:50.117+0000","configuredLevel":"INFO","subsystem":"updateNrfGeneralOpti
ons","processId":"1","nrfTxId":"nrf-tx-1469282724"}
```


Sample ERROR log statement for nrfconfiguration when Error Log Messages Enhancement feature is enabled:


```
{"instant":
{"epochSecond":1717082305,"nanoOfSecond":790424465},"thread":"XNIO-1

task-3","level":"ERROR","loggerName":"com.oracle.cgbu.cne.nrf.rest.NrfConfigur
ationController","message":"Response
      sent: 400 Bad Request for uri :
      http://ocnrf-nrfconfiguration:8080/nrf-configuration/v1/generalOptions
with the problem cause
      : MANDATORY_IE_INCORRECT and problem details : NRF-d5g.oracle.com:
Nnrf_Internal_Config:
      Multiple attributes are missing or incorrect:
      ONRF-CFG-GENOPT-
E0021","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.AbstractL
ogger","threadId":68,"threadPriority":5,"messageTimestamp":"2024-05-30T15:18:2
5.790+0000","configuredLevel":"WARN","subsystem":"updateNrfGeneralOptions","pr
ocessId":"1","nrfTxId":"nrf-tx-2145435457","xRequestId":"","hostname":"ocnrf-
nrfconfiguration-6956dd5454-27vh2","errorStatus":"400","errorTitle":"Bad
      Request","errorDetails":"NRF-d5g.oracle.com: Nnrf_Internal_Config:
Multiple attributes are
      missing or incorrect:
      ONRF-CFG-GENOPT-
```

E0021","errorCause":"MANDATORY_IE_INCORRECT","sender":"NRF-6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"}

Table 2-3 Log Attribute Details for nrfauditor and nrfconfiguration

Log Attribute	Details	Sample Value	Data Type	Notes
instant	Epoch time	{"epochSecond":1604654450,"nanoOfSecond":117406000}	Object	It is group of two values epochSecond and nanoOfSecond.
thread	Logging Thread Name	"XNIO-1 task-1"	String	
level	Log Level of the log printed	"INFO"	String	
loggerName	Class or Module which printed the log	"com.oracle.cgbu.cne.nrf.rest.NrfConfigurationController"	String	
message	Message related to the log providing brief details	"{logMsg=Update GeneralOptions request received with generalOptions, generalOptions={\"additionalAttributes\":null, \"nrfPlmnList\": [{\"mcc\":\"310\",\"mnc\":\"14\"}], \"enableF3\":null,\"enableF5\":null,\"maximumHopCount\":null,\"defaultLoad\":null, \"defaultPriority\":null,\"defaultPriorityAssignment\":null,\"defaultLoadAssignment\":null, \"ocnrfHost\":null,\"ocnrfPort\":null}}"	String	It states that a request is received to update the generaloptions. generaloptions received is also printed in the message.
endOfBatch	Log4j2 Internal Default from log4j2: false	false	Boolean	
loggerFqcn	Log4j2 Internal Fully Qualified class name of logger module	org.apache.logging.log4j.spi.AbstractLogger	String	
threadId	Thread Id generated internally by Log4j2	34	Integer	
threadPriority	Thread Priority set internally by Log4j2	5	Integer	
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ	"2020-11-06T09:20:50.117+0000"	String	
configuredLevel	Log Level configured	"INFO"	String	
subsystem	Subsystem inside microservice internal to microservice	"updateNrfGeneralOptions"	String	

Table 2-3 (Cont.) Log Attribute Details for nrfauditor and nrfconfiguration

Log Attribute	Details	Sample Value	Data Type	Notes
processId	Process ID internally assigned	"1"	String	
nrfTxId	Internal unique number for each log message to correlate the logs inside microservice	"nrf-tx-1469282724"}	String	
errorStatus	Indicates the status sent or received by NRF in ProblemDetails of HTTP response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	400	String	
errorTitle	Indicates the title sent or received by NRF in ProblemDetails of HTTP response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	Bad Request	String	
errorDetails	Indicates the detail sent or received by NRF in ProblemDetails of HTTP response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	NRF-d5g.oracle.com: Nnrf_Internal_Config: Multiple attributes are missing or incorrect: ONRF-CFG-GENOPT-E0021	String	
errorCause	Indicates the cause sent or received by NRF in ProblemDetails of HTTP response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	MANDATORY_IE_INCORRECT	String	
sender	Indicates the NRF-NRF Instance Id or Server header value as received in the error response **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.	NRF-6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c	String	

Table 2-3 (Cont.) Log Attribute Details for nrfauditor and nrfconfiguration

Log Attribute	Details	Sample Value	Data Type	Notes
receiver	Indicates the NRF-NRF Instance Id **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.		String	
subscriberId	Indicates the • 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. **Note:**This attribute will be added only for the ERROR logs when the Error Log Enhancement feature is enabled.		String	

This section provides log level attribute details for the ingressgateway service:
Sample log statement ingressgateway:


```
{"thread":"ingress-h2c-
epoll-3","level":"DEBUG","loggerName":"ocpm.cne.gateway.filters.PreGatewayFilt
er","message":"Exiting
PreGatewayFilter","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.sp
i.AbstractLogger","instant":
{"epochSecond":1604650229,"nanoOfSecond":4993000},"contextMap":
{"hostname":"ocnrf-ingressgateway-69f6544b8d-cdbgx","ingressTxId":"ingress-
tx-1087436877","ocLogId":"1604650229002_72_ocnrf-ingressgateway-69f6544b8d-
cdbgx"},"threadId":72,"threadPriority":5,"messageTimestamp":"2020-11-06
08:10:29.004","ocLogId":"1604650229002_72_ocnrf-ingressgateway-69f6544b8d-
cdbgx","pod":"ocnrf-ingressgateway-69f6544b8d-
cdbgx","processId":"1","instanceType":"prod","ingressTxId":"ingress-
tx-1087436877"}
```

Table 2-4 Log Attribute Details for ingressgateway

Log Attribute	Details	Sample Value	Data Type	Notes
thread	Logging Thread Name	"ingress-h2c-epoll-3"	String	
level	Log Level of the log printed	"DEBUG"	String	
loggerName	Class or Module which printed the log	"ocpm.cne.gateway.filters.PreGatewayFilter"	String	
message	Message related to the log providing brief details	"Exiting PreGatewayFilter"	String	Indicates that the method PreGatewayFilter is being exited.
endOfBatch	Log4j2 Internal Default from log4j2: false	false	boolean	
loggerFqcn	Log4j2 Internal Fully Qualified class name of logger module	org.apache.logging.log4j.spi.AbstractLogger	String	
instant	Epoch timestamp	{"epochSecond":1604650229,"nanoOfSecond":4993000}	Object	It is group of two values epochSecond and nanoOfSecond
contextMap	contents of log4j ThreadContext map	{"hostname":"ocnrf-ingressgateway-69f6544b8d-cdbgx", "ingressTxId":"ingress-tx-1087436877", "ocLogId":"1604650229002_72_ocnrf-ingressgateway-69f6544b8d-cdbgx"}	Object	
threadId	Thread Id generated internally by Log4j2	72	Integer	
threadPriority	Thread Priority set internally by Log4j2	5	Integer	
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ	"2020-11-06 08:10:29.004"	String	
ocLogId	End to End Log Identifier across the NRF microservices.	"1604650229002_72_ocnrf-ingressgateway-69f6544b8d-cdbgx"	String	Helps to correlate the logs across the microservices in OCNRF application
pod	Pod Name	"ocnrf-ingressgateway-69f6544b8d-cdbgx"	String	
processId	Process ID internally assigned	"1"	String	
instanceType	Instance type	"prod"	String	
ingressTxId	Transaction id that is added to log4j ThreadContext map and is unique to every transaction	"ingress-tx-1087436877"	String	

This section provides log level attribute details for the egressgateway service:
Sample log statement egressgateway:

```
{"thread":"main","level":"DEBUG","loggerName":"ocpm.cne.gateway.config.ScpDyna
micBeanConfiguration","message":"Property name: server.port and value:
8080","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.AbstractLo
gger","instant":
{"epochSecond":1604564777,"nanoOfSecond":135977000},"contextMap":
{},"threadId":1,"threadPriority":5,"messageTimestamp":"2020-11-05
08:26:17.135","ocLogId":"1604650229002_72_ocnrf-ingressgateway-69f6544b8d-
cdbgx","pod":"ocnrf-egressgateway-69f6544b8d-
cdbgx","processId":"1","instanceType":"prod","egressTxId":"egress-
tx-1087436877"}
```

Table 2-5 Log Attribute Details for egressgateway

Log Attribute	Details	Sample Value	Data Type	Notes
thread	Logging Thread Name	"main"	String	
level	Log Level of the log printed	"DEBUG"	String	
loggerName	Class or Module which printed the log	"ocpm.cne.gateway.config.ScpDynamicBeanConfiguration"	String	
message	Message related to the log providing brief details	"Property name: server.port and value: 8080"	String	
endOfBatch	Log4j2 Internal Default from log4j2: false	false	boolean	
loggerFqcn	Log4j2 Internal Fully Qualified class name of logger module	org.apache.logging.log4j.spi.AbstractLogger	String	
instant	Epoch timestamp	{"epochSecond":1604564777,"nanoOfSecond":135977000	Object	It is group of two values epochSecond and nanoOfSecond
contextMap	Elements in log4j ThreadContext map	{}	Object	
threadId	Thread Id generated internally by Log4j2	1	Integer	
threadPriority	Thread Priority set internally by Log4j2	5	Integer	
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ	"2020-11-05 08:26:17.135"	String	
ocLogId	End to End Log Identifier across the NRF microservices	"1604650229002_72_ocnrf-ingressgateway-69f6544b8d-cdbgx"	String	Helps to correlate the logs across the microservices in OCNRF application
pod	Name of the egress pod	"ocnrf-egressgateway-69f6544b8d-cdbgx"	String	

Table 2-5 (Cont.) Log Attribute Details for egressgateway

Log Attribute	Details	Sample Value	Data Type	Notes
processId	Process ID internally assigned	"1"	String	
instanceType	Instance type	"prod"	String	
egressTxId	Transaction id that is added to log4j ThreadContext map and is unique to every transaction	"egress-tx-1087436877"	String	

This section provides log level attribute details for the appinfo service:
Sample log statement appinfo:

```
{"name": "gunicorn.access","message": "127.0.0.1 - - [29/Oct/2020:18:39:35
+0000]\"GET /v1/liveness HTTP/1.1\" 200 16 \"-\"\"kube-probe/1.17+\"",
"level": "INFO","filename":"glogging.py","lineno": 344,"module":
"glogging","func":"access","thread":"MainThread","created":
"2020-10-29T18:39:35.799448"}
```

Table 2-6 Log Attribute Details for appinfo

Log Attribute	Details	Sample Value	Data Type
name	Logger name	"gunicorn.access"	String
message	Message related to the log providing brief details	"message": "127.0.0.1 - - [29/Oct/2020:18:39:35 +0000] \"GET /v1/liveness HTTP/1.1\" 200 16 \"-\" \"kube-probe/1.17+\""	String
level	Log Level of the log printed	"INFO"	String
filename	File name which generates the log	"glogging.py"	String
lineno	Line number	344	Integer
module	Python module name	"glogging"	String
func	Function Name	"func"	String
thread	Thread Name	"MainThread"	String
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ	"2020-10-29T18:39:35.799448"	String

Common Useful log attributes

The below log attributes will be available only through Kibana. These attribute names are part of Kubernetes labels which are added in each NRF POD.

Table 2-7 Common useful log attributes

Log Attribute	Details	Sample Value	Data Type	Notes
application	NF Type	"ocnrf"	String	complete attribute name in Kibana: kubernetes.labels.application
microservice	Kubernetes service name Format: <Helm-Release-Name>-<Microservice-Name>	(assuming the <Helm-Release-Name> is "nrfnorth" "nrfnorth-nfregistration" "nrfnorth-nfdiscovery" "nrfnorth-nfsubscription" "nrfnorth-nfaccesstoken" "nrfnorth-appinfo" "nrfnorth-nrfauditor" "nrfnorth-nrfconfiguration" "nrfnorth-ingressgateway" "nrfnorth-egressgateway"	String	
engVersion	Engineering version	"1.8.0"	String	
mktgVersion	Marketing version	"1.8.0.0.0"	String	
vendor	Vendor Name	"Oracle"	Integer	

3 Debug Tool

Overview

The Debug Tool provides third-party troubleshooting tools for debugging the runtime issues for the lab environment. Following are the available tools:

- tcpdump
- ip
- netstat
- curl
- ping
- nmap
- dig

3.1 Preconfiguration Steps

This section explains the preconfiguration steps for using the debug tool:

> **Note**
>
> - For CNE 23.2.0 and later versions, follow Step a of Configuration in CNE.
> - For CNE versions prior to 23.2.0, follow Step b of Configuration in CNE.

1. ***Configuration in CNE***
   The following configurations must be performed in the Bastion Host.
   a. When NRF is installed on CNE version 23.2.0 or above:

> **Note**
>
> - In CNE version 23.2.0 or above, the default CNE 23.2.0 Kyverno policy, disallow-capabilities, do not allow NET_ADMIN and NET_RAW capabilities that are required for debug tool.
> - To run Debug tool on CNE 23.2.0 and above, the user must modify the existing Kyverno policy, disallow-capabilities, as below.

Adding a Namespace to an Empty Resource

i. Run the following command to verify if the current disallow-capabilities cluster policy has namespace in it.

Example:

```
$ kubectl get clusterpolicies disallow-capabilities -oyaml
```

Sample output:

```
apiVersion: kyverno.io/v1
kind: ClusterPolicy
...
...
spec:
  rules:
  -exclude:
      any:
      -resources:{}
```

ii. If there are no namespaces, then patch the policy using the following command to add <namespace> under resources:

```
$ kubectl patch clusterpolicy disallow-capabilities --type=json \
  -p='[{"op": "add", "path": "/spec/rules/0/exclude/any/0/
resources", "value": {"namespaces":["<namespace>"]} }]'
```

Example:

```
$ kubectl patch clusterpolicy disallow-capabilities --type=json \
  -p='[{"op": "add", "path": "/spec/rules/0/exclude/any/0/
resources", "value": {"namespaces":["ocnrf"]} }]'
```

Sample output:

```
apiVersion: kyverno.io/v1
kind: ClusterPolicy
...
...
spec:
  rules:
  -exclude:
      resources:
        namespaces:
        -ocnrf
```

iii. If in case it is needed to remove the namespace added in the above step, use the following command:

```
$ kubectl patch clusterpolicy disallow-capabilities --type=json \
  -p='[{"op": "replace", "path": "/spec/rules/0/exclude/any/0/
resources", "value": {} }]'
```

Sample output:

```
apiVersion: kyverno.io/v1
kind: ClusterPolicy
```

```
...
...
spec:
  rules:
  -exclude:
      any:
      -resources:{}
```

Adding a Namespace to an Existing Namespace List

i. Run the following command to verify if the current disallow-capabilities cluster policy has namespaces in it.
Example:

```
$ kubectl get clusterpolicies disallow-capabilities -oyaml
```

Sample output:

```
apiVersion: kyverno.io/v1
kind: ClusterPolicy
...
...
spec:
  rules:
  -exclude:
      any:
      -resources:
          namespaces:
          -namespace1
          -namespace2
          -namespace3
```

ii. If there are namespaces already added, then patch the policy using the following command to add <namespace> to the existing list:

```
$ kubectl patch clusterpolicy disallow-capabilities --type=json \
  -p='[{"op": "add", "path": "/spec/rules/0/exclude/any/0/resources/
namespaces/-", "value": "<namespace>" }]'
```

Example:

```
$ kubectl patch clusterpolicy disallow-capabilities --type=json \
  -p='[{"op": "add", "path": "/spec/rules/0/exclude/any/0/resources/
namespaces/-", "value": "ocnrf" }]'
```

Sample output:

```
apiVersion: kyverno.io/v1
kind: ClusterPolicy
...
...
spec:
  rules:
  -exclude:
```

```
    resources:
      namespaces:
      -namespace1
      -namespace2
      -namespace3
      -ocnrf
```

iii. If in case it is needed to remove the namespace added in the above step, use the following command:

```
$ kubectl patch clusterpolicy disallow-capabilities --type=json \
  -p='[{"op": "remove", "path": "/spec/rules/0/exclude/any/0/
resources/namespaces/<index>"}]'
```

Example:

```
$ kubectl patch clusterpolicy disallow-capabilities --type=json \
  -p='[{"op": "remove", "path": "/spec/rules/0/exclude/any/0/
resources/namespaces/3"}]'
```

Sample output:

```
apiVersion: kyverno.io/v1
kind: ClusterPolicy
...
...
spec:
  rules:
  -exclude:
      resources:
        namespaces:
        -namespace1
        -namespace2
        -namespace3
```

> **Note**
>
> While removing the namespace, provide the index value for namespace within the array. The index starts from '0'.

b. When NRF is installed on CNE version prior to 23.2.0
PodSecurityPolicy (PSP) Creation

Create a PSP by running the following command from the bastion host. The parameters **readOnlyRootFileSystem**, **allowPrivilegeEscalation**, and **allowedCapabilities** are required by debug container.

> **Note**
>
> Other parameters are mandatory for PSP creation and can be customized as per the CNE environment. The default values are recommended.

```
$ kubectl apply -f - <<EOF

apiVersion: policy/v1beta1
kind: PodSecurityPolicy
metadata:
  name: debug-tool-psp
spec:
  readOnlyRootFilesystem: false
  allowPrivilegeEscalation: true
  allowedCapabilities:
  - NET_ADMIN
  - NET_RAW
  fsGroup:
    ranges:
    - max: 65535
      min: 1
    rule: MustRunAs
  runAsUser:
    rule: MustRunAsNonRoot
  seLinux:
    rule: RunAsAny
  supplementalGroups:
    rule: RunAsAny
  volumes:
  - configMap
  - downwardAPI
  - emptyDir
  - persistentVolumeClaim
  - projected
  - secret
EOF
```

Role Creation

Run the following command to create a role for the PSP:

```
kubectl apply -f - <<EOF
apiVersion: rbac.authorization.k8s.io/v1
kind: Role
metadata:
  name: debug-tool-role
  namespace: cncc
rules:
- apiGroups:
  - policy
  resources:
  - podsecuritypolicies
  verbs:
  - use
```

```
  resourceNames:
  - debug-tool-psp
EOF
```

RoleBinding Creation

Run the following command to associate the service account for the NRF namespace with the role created for the PSP:

```
$ kubectl apply -f - <<EOF
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: debug-tool-rolebinding
  namespace: ocnrf
roleRef:
  apiGroup: rbac.authorization.k8s.io
  kind: Role
  name: debug-tool-role
subjects:
- kind: Group
  apiGroup: rbac.authorization.k8s.io
  name: system:serviceaccounts
EOF
```

2. ***Configuration in NF specific Helm***
   Following updates must be performed in custom_values.yaml file.
   a. Log in to the NF server.
   b. Open the custom_values file:

```
$ vim <custom_values file>
```

   c. Under global configuration, add the following:

```
# Allowed Values: DISABLED, ENABLED
# Preference is to set  "resources" request and limit to same values to
avoid HPA issues.
extraContainers: DISABLED
debugToolContainerMemoryLimit: 4Gi
extraContainersVolumesTpl: |
  - name: debug-tools-dir
    emptyDir:
      medium: Memory
      sizeLimit: {{ .Values.global.debugToolContainerMemoryLimit |
quote }}
extraContainersTpl: |
    - command:
        - /bin/sleep
        - infinity
      image: <image-name>:<image-tag>
      imagePullPolicy: Always
      name: tools
      resources:
        requests:
          ephemeral-storage: "512Mi"
```

```
          cpu: "0.5"
          memory: {{ .Values.global.debugToolContainerMemoryLimit |
quote }}
        limits:
          ephemeral-storage: "512Mi"
          cpu: "0.5"
          memory: {{ .Values.global.debugToolContainerMemoryLimit |
quote }}
      securityContext:
        allowPrivilegeEscalation: true
        capabilities:
          drop:
          - ALL
          add:
          - NET_RAW
          - NET_ADMIN
        runAsUser: <user-id>
      volumeMounts:
      - mountPath: /tmp/tools
        name: debug-tools-dir
```

> **Note**
>
> - Debug Tool Container comes up with the default user ID - 7000. If you want to override this default value, use the `runAsUser` field, or else, you can skip the field.
>
>   Default value: uid=7000(debugtool) gid=7000(debugtool) groups=7000(debugtool)
> - In case you want to customize the container name, replace the `name` field in the above values.yaml with the following:
>
>   ```
>   name: {{ printf "%s-tools-%s" (include "getprefix" .)
>   (include "getsuffix" .) | trunc 63 | trimPrefix "-" |
>   trimSuffix "-"  }}
>   ```
>
>   This will ensure that the container name is prefixed and suffixed with the necessary values.

d. Under service specific configurations for which debugging is required, add the following:

```
# Allowed Values: DISABLED, ENABLED, USE_GLOBAL_VALUE
extraContainers: USE_GLOBAL_VALUE
```

> **Note**
>
> - At the global level, `extraContainers` flag can be used to enable or disable injecting extra containers globally. This ensures that all the services that use this global value have extra containers enabled or disabled using a single flag.
> - At the service level, `extraContainers` flag determines whether to use the extra container configuration from the global level or enable or disable injecting extra containers for the specific service.

3.2 Deploy Debug Tool

Following is the procedure to run Debug Tool.

1. Run the following command to retrieve the POD details:

   ```
   $ kubectl get pods -n <k8s namespace>
   ```

   Example:

   ```
   $ kubectl get pods -n ocnrf
   ```

   Sample output:

   ```
   NAME                                      READY   STATUS    RESTARTS   AGE
   ocnrf-egressgateway-d6567bbdb-9jrsx       3/3     Running   0          30h
   ocnrf-egressgateway-d6567bbdb-ntn2v       3/3     Running   0          30h
   ocnrf-ingressgateway-754d645984-h9vzq     3/3     Running   0          30h
   ocnrf-ingressgateway-754d645984-njz4w     3/3     Running   0          30h
   ocnrf-nfaccesstoken-59fb96494c-k8w9p      3/3     Running   0          30h
   ocnrf-nfaccesstoken-49fb96494c-k8w9q      3/3     Running   0          30h
   ocnrf-nfdiscovery-84965d4fb9-rjxg2        2/2     Running   0          30h
   ocnrf-nfdiscovery-94965d4fb9-rjxg3        2/2     Running   0          30h
   ocnrf-nfregistration-64f4d8f5d5-6q92j     2/2     Running   0          30h
   ocnrf-nfregistration-44f4d8f5d5-6q92i     2/2     Running   0          30h
   ocnrf-nfsubscription-5b6db965b9-gcvpf     2/2     Running   0          30h
   ocnrf-nfsubscription-4b6db965b9-gcvpe     2/2     Running   0          30h
   ocnrf-nrfauditor-67b676dd87-xktbm         2/2     Running   0          30h
   ocnrf-nrfconfiguration-678fddc5f5-c5htj   2/2     Running   0          30h
   ocnrf-appinfo-8b7879cdb-jds4r             2/2     Running   0          30h
   ```

2. Run the following command to enter Debug Tool Container:

   ```
   $ kubectl exec -it <pod name> -c <debug_container name> -n <namespace> bash
   ```

   Example:

   ```
   $ kubectl exec -it ocnrf-nfaccesstoken-49fb96494c-k8w9q -c tools -n ocnrf
   bash
   ```

3. Run the debug tools:

   ```
   bash -4.2$ <debug_tools>
   ```

   Example:

   ```
   bash -4.2$ tcpdump
   ```

4. Copy the output files from container to host:

   ```
   $ kubectl cp -c <debug_container name> <pod name>:<file location in
   container> -n <namespace> <destination location>
   ```

   Example:

   ```
   $ kubectl cp -c tools ocnrf-nfaccesstoken-49fb96494c-k8w9q:/tmp/
   capture.pcap -n ocnrf /tmp/
   ```

3.3 Tools Tested in Debug Container

Following is the list of debugging tools that are tested.

tcpdump

The following table describes the options that are testing using `tcpdump` tool.

Table 3-1 tcpdump

Options Tested	Description	Output	Capabilities
-D	Print the list of the network interfaces available on the system and on which *tcpdump* can capture packets.	`tcpdump -D` 1. eth02. 2. nflog (Linux netfilter log (NFLOG) interface) 3. nfqueue (Linux netfilter queue (NFQUEUE) interface) 4. any (Pseudo-device that captures on all interfaces) 5. lo [Loopback]	NET_ADMIN, NET_RAW
-i	Listen on *interface*	`tcpdump -i eth0` tcpdump: verbose output suppressed, use -v or -vv for full protocol decodelistening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes12:10:37.381199 IP cncc-core-ingress-gateway-7ffc49bb7f-2kkhc.46519 > kubernetes.default.svc.cluster.local.https: Flags [P.], seq 1986927241:1986927276, ack 1334332290, win 626, options [nop,nop,TS val 849591834 ecr 849561833], length 3512:10:37.381952 IP cncc-core-ingress-gateway-7ffc49bb7f-2kkhc.45868 > kube-dns.kube-system.svc.cluster.local.domain: 62870+ PTR? 1.0.96.10.in-addr.arpa. (40)	NET_ADMIN, NET_RAW

Table 3-1 (Cont.) tcpdump

Options Tested	Description	Output	Capabilities
-w	Write the raw packets to file rather than parsing and printing them out.	`tcpdump -w capture.pcap -i eth0`	NET_ADMIN, NET_RAW
-r	Read packets from *file* (which was created with the **-w** option).	`tcpdump -r capture.pcap` reading from file /tmp/capture.pcap, link-type EN10MB (Ethernet)12:13:07.381019 IP cncc-core-ingress-gateway-7ffc49bb7f-2kkhc.46519 > kubernetes.default.svc.cluster.local.https: Flags [P.], seq 1986927416:1986927451, ack 1334332445, win 626, options [nop,nop,TS val 849741834 ecr 849711834], length 3512:13:07.381194 IP kubernetes.default.svc.cluster.local.https > cncc-core-ingress-gateway-7ffc49bb7f-2kkhc.46519: Flags [P.], seq 1:32, ack 35, win 247, options [nop,nop,TS val 849741834 ecr 849741834], length 3112:13:07.381207 IP cncc-core-ingress-gateway-7ffc49bb7f-2kkhc.46519 > kubernetes.default.svc.cluster.local.https: Flags [.], ack 32, win 626, options [nop,nop,TS val 849741834 ecr 849741834], length 0	NET_ADMIN, NET_RAW

ip

The following table describes the options that are testing using `ip` tool.

Table 3-2 ip

Options Tested	Description	Output	Capabilities
addr show	Look at protocol addresses.	`ip addr show` 1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group defaultlink/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00inet 127.0.0.1/8 scope host lovalid_lft forever preferred_lft forever2: tunl0@NONE: <NOARP> mtu 1480 qdisc noop state DOWN group defaultlink/ipip 0.0.0.0 brd 0.0.0.04: eth0@if190: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1440 qdisc noqueue state UP group defaultlink/ether aa:5a:27:8d:74:6f brd ff:ff:ff:ff:ff:ff link-netnsid 0inet 192.168.219.112/32 scope global eth0valid_lft forever preferred_lft forever	--
route show	List routes	`ip route show` default via 169.254.1.1 dev eth0 169.254.1.1 dev eth0 scope link	--

Table 3-2 (Cont.) ip

Options Tested	Description	Output	Capabilities
addrlabel list	List address labels	`ip addrlabel list` prefix ::1/128 label 0 prefix ::/96 label 3 prefix ::ffff:0.0.0.0/96 label 4 prefix 2001::/32 label 6 prefix 2001:10::/28 label 7 prefix 3ffe::/16 label 12 prefix 2002::/16 label 2 prefix fec0::/10 label 11 prefix fc00::/7 label 5 prefix ::/0 label 1	--

netstat

The following table describes the options that are testing using `netstat` tool.

Table 3-3 netstat

Options Tested	Description	Output	Capabilities
-a	Show both listening and non-listening (for TCP, this means established connections) sockets.	`netstat -a` Active Internet connections (servers and established)Proto Recv-Q Send-Q Local Address Foreign Address Statetcp 0 0 0.0.0.0:tproxy 0.0.0.0:* LISTENtcp 0 0 0.0.0.0:websm 0.0.0.0:* LISTENtcp 0 0 cncc-core-ingress:websm 10-178-254-194.ku:47292 TIME_WAITtcp 0 0 cncc-core-ingress:46519 kubernetes.defaul:https ESTABLISHEDtcp 0 0 cncc-core-ingress:websm 10-178-254-194.ku:47240 TIME_WAITtcp 0 0 cncc-core-ingress:websm 10-178-254-194.ku:47347 TIME_WAITudp 0 0 localhost:59351 localhost:ambit-lm ESTABLISHEDActive UNIX domain sockets (servers and established)Proto RefCnt Flags Type State I-Node Pathunix 2 [ ] STREAM CONNECTED 576064861	--
-l	Show only listening sockets.	`netstat -l` Active Internet connections (only servers)Proto Recv-Q Send-Q Local Address Foreign Address Statetcp 0 0 0.0.0.0:tproxy 0.0.0.0:* LISTENtcp 0 0 0.0.0.0:websm 0.0.0.0:* LISTENActive UNIX domain sockets (only servers)Proto RefCnt Flags Type State I-Node Path	--
-s	Display summary statistics for each protocol.	`netstat -s` Ip:4070 total packets received0 forwarded0 incoming packets discarded4070 incoming packets delivered4315 requests sent outIcmp:0 ICMP messages received0 input ICMP message failed.ICMP input histogram:2 ICMP messages sent0 ICMP messages failedICMP output histogram:destination unreachable: 2	--

Table 3-3 (Cont.) netstat

Options Tested	Description	Output	Capabilities
-i	Display a table of all network interfaces.	`netstat -i` Kernel Interface tableIface MTU RX-OK RX-ERR RX-DRP RX-OVR TX-OK TX-ERR TX-DRP TX-OVR Flgeth0 1440 4131 0 0 0 4355 0 0 0 BMRUlo 65536 0 0 0 0 0 0 0 0 LRU	--

curl

The following table describes the options that are testing using `curl` tool.

Table 3-4 curl

Options Tested	Description	Output	Capabilities
-o	Write output to <file> instead of stdout.	`curl -o file.txt http://abc.com/file.txt`	--
-x	Use the specified HTTP proxy.	`curl -x proxy.com:8080 -o http://abc.com/file.txt`	--

ping

The following table describes the options that are testing using `ping` tool.

Table 3-5 ping

Options Tested	Description	Output	Capabilities
<ip>	Run a ping test to see whether the target host is reachable or not.	`ping 10.178.254.194`	NET_ADMIN, NET_RAW
-c	Stop after sending 'c' number of ECHO_REQUEST packets.	`ping -c 5 10.178.254.194`	NET_ADMIN, NET_RAW
-f (with non zero interval)	Flood ping. For every ECHO_REQUEST sent, a period "." is printed, while for every ECHO_REPLY received a backspace is printed.	`ping -f -i 2 10.178.254.194`	NET_ADMIN, NET_RAW

nmap

The following table describes the options that are testing using `nmap` tool.

Table 3-6 nmap

Options Tested	Description	Output	Capabilities

<ip>	Scan for Live hosts, Operating systems, packet filters and open ports running on remote hosts.	`nmap 10.178.254.194`
		
		`Starting Nmap 6.40 ( http://nmap.org ) at 2020-09-29 05:54 UTCNmap scan report for`
		
		`10-178-254-194.kubernetes.default.svc.cluster .local (10.178.254.194)Host is up (0.00046s latency).Not shown: 995 closed portsPORT STATE SERVICE22/tcp open ssh179/tcp open bgp6666/tcp open irc6667/tcp open irc30000/tcp open unknownNmap done: 1 IP address (1 host up) scanned in 0.04 seconds`
			--

Table 3-6 (Cont.) nmap

Options Tested	Description	Output	Capabilities

-v	Increase verbosity level	`nmap -v 10.178.254.194`
		
		`Starting Nmap 6.40 ( http://nmap.org ) at 2020-09-29 05:55 UTC`
		`Initiating Ping Scan at 05:55`
		`Scanning 10.178.254.194 [2 ports]`
		`Completed Ping Scan at 05:55, 0.00s elapsed (1 total hosts)`
		`Initiating Parallel DNS resolution of 1 host. at 05:55`
		`Completed Parallel DNS resolution of 1 host. at 05:55, 0.00s elapsed`
		`Initiating Connect Scan at 05:55`
		`Scanning 10-178-254-194.kubernetes.default.svc.cluster.local (10.178.254.194) [1000 ports]`
		`Discovered open port 22/tcp on 10.178.254.194`
		`Discovered open port 30000/tcp on 10.178.254.194`
		`Discovered open port 6667/tcp on 10.178.254.194`
		`Discovered open port 6666/tcp on 10.178.254.194`
		`Discovered open port 179/tcp on 10.178.254.194`
		`Completed Connect Scan at 05:55, 0.02s elapsed (1000 total ports)`
		`Nmap scan report for 10-178-254-194.kubernetes.default.svc.cluster.local (10.178.254.194)`
		`Host is up (0.00039s latency).`
		`Not shown: 995 closed ports`
		`PORT STATE SERVICE`
		`22/tcp open ssh`
		`179/tcp open bgp`
		`6666/tcp open irc`
		`6667/tcp open irc`
		`30000/tcp open unknown`
		
		`Read data files from: /usr/bin/../share/nmap`
		`Nmap done: 1 IP address (1 host up) scanned in 0.04 seconds`
			--

Table 3-6 (Cont.) nmap

Options Tested	Description	Output	Capabilities

-iL	Scan all the listed IP addresses in a file. Sample file	`nmap -iL sample.txt`
		
		`Starting Nmap 6.40 ( http://nmap.org ) at 2020-09-29 05:57 UTC`
		`Nmap scan report for localhost (127.0.0.1)`
		`Host is up (0.00036s latency).`
		`Other addresses for localhost (not scanned): 127.0.0.1`
		`Not shown: 998 closed ports`
		`PORT STATE SERVICE`
		`8081/tcp open blackice-icecap`
		`9090/tcp open zeus-admin`
		
		`Nmap scan report for 10-178-254-194.kubernetes.default.svc.cluster.local (10.178.254.194)`
		`Host is up (0.00040s latency).`
		`Not shown: 995 closed ports`
		`PORT STATE SERVICE`
		`22/tcp open ssh`
		`179/tcp open bgp`
		`6666/tcp open irc`
		`6667/tcp open irc`
		`30000/tcp open unknown`
		
		`Nmap done: 2 IP addresses (2 hosts up) scanned in 0.06 seconds`
			--

dig

The following table describes the options that are testing using `dig` tool.

Table 3-7 dig

Options Tested	Description	Output	Capabilities
<ip>	It performs DNS lookups and displays the answers that are returned from the name server(s) that were queried.	`dig 10.178.254.194` **Note**: The IP should be reachable from inside the container.	--
-x	Query DNS Reverse lookup.	`dig -x 10.178.254.194`	--

3.4 Debug Tool Configuration Parameters

Following are the parameters used to configure debug tool.

CNE Parameters

Table 3-8 CNE Parameters

Parameter	Description
apiVersion	APIVersion defines the version schema of this representation of an object.
kind	Kind is a string value representing the REST resource this object represents.
metadata	Standard object's metadata.
metadata.name	Name must be unique within a namespace.
spec	This defines the policy enforced.
spec.allowPrivilegeEscalation	Gates whether or not a user is allowed to set the security context of a container to allowPrivilegeEscalation=true.
spec.allowedCapabilities	Provides a list of capabilities that are allowed to be added to a container.
spec.fsGroup	Controls the supplemental group applied to some volumes. RunAsAny allows any fsGroup ID to be specified.
spec.runAsUser	Controls which user ID the containers are run with. RunAsAny allows any runAsUser to be specified.
spec.seLinux	RunAsAny allows any seLinuxOptions to be specified.
spec.supplementalGroups	Controls which group IDs containers add. RunAsAny allows any supplementalGroups to be specified.
spec.volumes	Provides a list of allowed volume types. The allowable values correspond to the volume sources that are defined when creating a volume.

Role Creation Parameters

Table 3-9 Role Creation

Parameter	Description
apiVersion	APIVersion defines the schema version of this representation of an object.
kind	Kind is a string value representing the REST resource this object represents.
metadata	Standard object's metadata.
metadata.name	Name must be unique within a namespace.
metadata.namespace	Namespace defines the space within which each name must be unique.
rules	Rules holds all the PolicyRules for this Role
apiGroups	APIGroups is the name of the APIGroup that contains the resources.
rules.resources	Resources is a list of resources this rule applies to.
rules.verbs	Verbs is a list of Verbs that apply to ALL the ResourceKinds and AttributeRestrictions contained in this rule.
rules.resourceNames	ResourceNames is an optional allowed list of names that the rule applies to.

Table 3-10 Role Binding Creation

Parameter	Description
apiVersion	APIVersion defines the version of schema of this representation of an object.
kind	Kind is a string value representing the REST resource this object represents.
metadata	Standard object's metadata.
metadata.name	Name must be unique within a namespace.
metadata.namespace	Namespace defines the space within which each name must be unique.
roleRef	RoleRef can reference a Role in the current namespace or a ClusterRole in the global namespace.
roleRef.apiGroup	APIGroup is the group for the resource being referenced
roleRef.kind	Kind is the type of resource being referenced
roleRef.name	Name is the name of resource being referenced
subjects	Subjects holds references to the objects the role applies to.
subjects.kind	Kind of object being referenced. Values defined by this API group are "User", "Group", and "ServiceAccount".
subjects.apiGroup	APIGroup holds the API group of the referenced subject.
subjects.name	Name of the object being referenced.

Debug Tool Configuration Parameters

Table 3-11 Debug Tool Configuration Parameters

Parameter	Description
extraContainers	Specifies the spawns debug container along with application container in the pod.
debugToolContainerMemoryLimit	Indicates the memory assigned for the debug tool container.
extraContainersVolumesTpl	Specifies the extra container template for the debug tool volume.
extraContainersVolumesTpl.name	Indicates the name of the volume for debug tool logs storage.
extraContainersVolumesTpl.emptyDir.medium	Indicates the location where `emptyDir` volume is stored.
extraContainersVolumesTpl.emptyDir.sizeLimit	Indicates the `emptyDir` volume size.
command	String array used for container command.
image	Docker image name
imagePullPolicy	Image Pull Policy
name	Name of the container
resources	Compute Resources required by this container
resources.limits	Limits describes the maximum amount of compute resources allowed
resources.requests	Requests describes the minimum amount of compute resources required
resources.limits.cpu	CPU limits
resources.limits.memory	Memory limits
resources.limits.ephemeral-storage	Ephemeral Storage limits

Table 3-11 (Cont.) Debug Tool Configuration Parameters

Parameter	Description
resources.requests.cpu	CPU requests
resources.requests.memory	Memory requests
resources.requests.ephemeral-storage	Ephemeral Storage requests
securityContext	Security options the container should run with.
securityContext.allowPrivilegeEscalation	AllowPrivilegeEscalation controls whether a process can gain more privileges than its parent process. This directly controls if the no_new_privs flag will be set on the container process
secuirtyContext.readOnlyRootFilesyste m	Whether this container has a read-only root filesystem. Default is false.
securityContext.capabilities	The capabilities to add or drop when running containers. Defaults to the default set of capabilities granted by the container runtime.
securityContext.capabilities.drop	Removed capabilities
secuirtyContext.capabilities.add	Added capabilities
securityContext.runAsUser	The UID to run the entry point of the container process.
volumeMounts.mountPath	Indicates the path for volume mount.
volumeMounts.name	Indicates the name of the directory for debug tool logs storage.

4 Troubleshooting NRF

This chapter provides information to troubleshoot the common errors that can be encountered during the preinstall, installation, upgrade, and rollback procedures of Oracle Communications Cloud Native Core, Network Repository Function (NRF).

Following are the troubleshooting procedures:

- Helm Install Failure
- Custom Value File Parse Failure
- Helm Test Error Scenarios
- Upgrade or Rollback Failure

> **Note**
>
> `kubectl` commands might vary based on the platform deployment. Replace `kubectl` with Kubernetes environment-specific command line tool to configure Kubernetes resources through kube-api server. The instructions provided in this document are as per the Oracle Communications Cloud Native Core, Cloud Native Environment (CNE) version of kube-api server.

> **Caution**
>
> User, computer and applications, and character encoding settings may cause an issue when copy-pasting commands or any content from PDF. PDF reader version also affects the copy-pasting functionality. It is recommended to verify the copy-pasted content, especially when hyphens or any special characters are part of the copied content.

> **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.

4.1 Generic Checklist

The following sections provide a generic checklist for troubleshooting tips.

Deployment related tips

Perform the following checks after the deployment:

- Are NRF deployment, pods, and services created?
  Are NRF deployment, pods, and services running and available?

Run the following command:

```
# kubectl -n <namespace> get deployments,pods,svc
```

Inspect the output, check the following columns:

- AVAILABLE of deployment
- READY, STATUS, and RESTARTS of a pod
- PORT(S) of service

- Is the correct image used?
  Is the correct environment variables set in the deployment?

  Run the following command:

  ```
  # kubectl -n <namespace> get deployment <deployment-name> -o yaml
  ```

  Inspect the output, check the environment and image.

  ```
  # kubectl -n nrf-svc get deployment ocnrf-nfregistration -o yaml
  apiVersion: extensions/v1beta1
  kind: Deployment
  metadata:
    annotations:
      deployment.kubernetes.io/revision: "1"
      kubectl.kubernetes.io/last-applied-configuration: |
        {"apiVersion":"apps/v1","kind":"Deployment","metadata":
  {"annotations":{},"name":"ocnrf-nfregistration","namespace":"nrf-
  svc"},"spec":{"replicas":1,"selector":{"matchLabels":{"app":"ocnrf-
  nfregistration"}},"template":{"metadata":{"labels":{"app":"ocnrf-
  nfregistration"}},"spec":{"containers":[{"env":
  [{"name":"MYSQL_HOST","value":"mysql"},
  {"name":"MYSQL_PORT","value":"3306"},
  {"name":"MYSQL_DATABASE","value":"nrfdb"},
  {"name":"NRF_REGISTRATION_ENDPOINT","value":"ocnrf-nfregistration"},
  {"name":"NRF_SUBSCRIPTION_ENDPOINT","value":"ocnrf-nfsubscription"},
  {"name":"NF_HEARTBEAT","value":"120"},
  {"name":"DISC_VALIDITY_PERIOD","value":"3600"}],"image":"dsr-master0:5000/
  ocnrf-nfregistration:latest","imagePullPolicy":"Always","name":"ocnrf-
  nfregistration","ports":[{"containerPort":8080,"name":"server"}]}]}}}}
    creationTimestamp: 2018-08-27T15:45:59Z
    generation: 1
    name: ocnrf-nfregistration
    namespace: nrf-svc
    resourceVersion: "2336498"
    selfLink: /apis/extensions/v1beta1/namespaces/nrf-svc/deployments/ocnrf-
  nfregistration
    uid: 4b82fe89-aa10-11e8-95fd-fa163f20f9e2
  spec:
    progressDeadlineSeconds: 600
    replicas: 1
    revisionHistoryLimit: 10
    selector:
      matchLabels:
        app: ocnrf-nfregistration
  ```

```
    strategy:
      rollingUpdate:
        maxSurge: 25%
        maxUnavailable: 25%
      type: RollingUpdate
    template:
      metadata:
        creationTimestamp: null
        labels:
          app: ocnrf-nfregistration
      spec:
        containers:
        - env:
          - name: MYSQL_HOST
            value: mysql
          - name: MYSQL_PORT
            value: "3306"
          - name: MYSQL_DATABASE
            value: nrfdb
          - name: NRF_REGISTRATION_ENDPOINT
            value: ocnrf-nfregistration
          - name: NRF_SUBSCRIPTION_ENDPOINT
            value: ocnrf-nfsubscription
          - name: NF_HEARTBEAT
            value: "120"
          - name: DISC_VALIDITY_PERIOD
            value: "3600"
          image: dsr-master0:5000/ocnrf-nfregistration:latest
          imagePullPolicy: Always
          name: ocnrf-nfregistration
          ports:
          - containerPort: 8080
            name: server
            protocol: TCP
          resources: {}
          terminationMessagePath: /dev/termination-log
          terminationMessagePolicy: File
        dnsPolicy: ClusterFirst
        restartPolicy: Always
        schedulerName: default-scheduler
        securityContext: {}
        terminationGracePeriodSeconds: 30
  status:
    availableReplicas: 1
    conditions:
    - lastTransitionTime: 2018-08-27T15:46:01Z
      lastUpdateTime: 2018-08-27T15:46:01Z
      message: Deployment has minimum availability.
      reason: MinimumReplicasAvailable
      status: "True"
      type: Available
    - lastTransitionTime: 2018-08-27T15:45:59Z
      lastUpdateTime: 2018-08-27T15:46:01Z
      message: ReplicaSet "ocnrf-nfregistration-7898d657d9" has successfully
  progressed.
      reason: NewReplicaSetAvailable
```

```
    status: "True"
    type: Progressing
  observedGeneration: 1
  readyReplicas: 1
  replicas: 1
  updatedReplicas: 1
```

- Check if the microservices can access each other using REST interface.
  Run the following command:

  ```
  # kubectl -n <namespace> exec <pod name> -- curl <uri>
  ```

  Example:

  ```
  # kubectl -n nrf-svc exec ocnrf-nfregistration-44f4d8f5d5-6q92i -- curl
  http://ocnrf-nfregistration:8080/nnrf-nfm/v1/nf-instances
  ```

> **Note**
>
> These commands are in their simple form and display the logs only if there is a single nrf<registration> and nf<subscription> pod deployed.

Application related tips

Run the following command to check the application logs and look for exceptions:

```
# kubectl -n <namespace> logs -f <pod name>
```

You can use '-f' to follow the logs or 'grep' for a specific pattern in the log output.

Example:

```
# kubectl -n nrf-svc logs -f $(kubectl -n nrf-svc get pods -o name|cut -d'/' -
f2|grep nfr)
# kubectl -n nrf-svc logs -f $(kubectl -n nrf-svc get pods -o name|cut -d'/' -
f2|grep nfs)
```

> **Note**
>
> These commands are in their simple form and display the logs only if there is 1 nrf<registration> and nf<subscription> pod deployed.

4.2 Deployment Related Issues

This section describes the most common deployment related issues and their resolution steps. It is recommended to perform the resolution steps provided in this guide. If the issue still persists, then contact My Oracle Support (MOS).

4.2.1 Installation

This section describes the common installation related issues and their resolution steps.

4.2.1.1 Helm Install Failure

This section describes the various scenarios in which `helm install` might fail. Following are some of the scenarios:

- Incorrect image name in ocnrf-custom-values files
- Docker registry is configured incorrectly
- Continuous Restart of Pods

4.2.1.1.1 Incorrect image name in ocnrf-custom-values files

Problem

`helm install` might fail if an incorrect image name is provided in the *ocnrf-custom-values.yaml* file.

Error Code/Error Message

When `kubectl get pods -n <ocnrf_namespace>` is performed, the status of the pods might be ImagePullBackOff or ErrImagePull.

For example:

```
$ kubectl get pods -n ocnrf
```

```
NAME                                    READY  STATUS            RESTARTS  AGE
ocnrf-egressgateway-d6567bbdb-9jrsx     2/2    ImagePullBackOff  0         30h
ocnrf-egressgateway-d6567bbdb-ntn2v     2/2    Running           0         30h
ocnrf-ingressgateway-754d645984-h9vzq   2/2    Running           0         30h
ocnrf-ingressgateway-754d645984-njz4w   2/2    Running           0         30h
ocnrf-nfaccesstoken-59fb96494c-k8w9p    1/1    Running           0         30h
ocnrf-nfaccesstoken-49fb96494c-k8w9q    1/1    Running           0         30h
ocnrf-nfdiscovery-84965d4fb9-rjxg2      1/1    Running           0         30h
ocnrf-nfdiscovery-94965d4fb9-rjxg3      1/1    Running           0         30h
ocnrf-nfregistration-64f4d8f5d5-6q92j   1/1    Running           0         30h
ocnrf-nfregistration-44f4d8f5d5-6q92i   1/1    Running           0         30h
ocnrf-nfsubscription-5b6db965b9-gcvpf   1/1    Running           0         30h
ocnrf-nfsubscription-4b6db965b9-gcvpe   1/1    Running           0         30h
ocnrf-nrfauditor-67b676dd87-xktbm       1/1    Running           0         30h
ocnrf-nrfconfiguration-678fddc5f5-c5htj 1/1    Running           0         30h
ocnrf-appinfo-8b7879cdb-jds4r           1/1    Running           0         30h
```

Solution

Perform the following steps to verify and correct the image name:

1. Check *ocnrf-custom-values.yaml* file has the release specific image name and tags.

   ```
   vi ocnrf-custom-values-<release-number>
   ```

   For NRF images details, see "**Customizing NRF**" in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
2. Edit *ocnrf-custom-values* file in case the release specific image name and tags must be modified.
3. Save the file.

4. Run the following command to delete the deployment:

   ```
   helm delete --purge <release_namespace>
   ```

   Sample command:

   ```
   helm delete --purge ocnrf
   ```

5. In case the helm purge does not clean the deployment and Kubernetes objects completely, then see the "**Uninstalling NRF**" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
6. Run `helm install` command. For helm install command, see the "**Customizing NRF**" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
7. Run `kubectl get pods -n <ocnrf_namespace>` to verify if the status of all the pods is **Running**.
   For example:

   ```
   $ kubectl get pods -n ocnrf
   ```

   ```
   NAME                                    READY  STATUS   RESTARTS  AGE
   ocnrf-egressgateway-d6567bbdb-9jrsx     2/2    Running  0         30h
   ocnrf-egressgateway-d6567bbdb-ntn2v     2/2    Running  0         30h
   ocnrf-ingressgateway-754d645984-h9vzq   2/2    Running  0         30h
   ocnrf-ingressgateway-754d645984-njz4w   2/2    Running  0         30h
   ocnrf-nfaccesstoken-59fb96494c-k8w9p    1/1    Running  0         30h
   ocnrf-nfaccesstoken-49fb96494c-k8w9q    1/1    Running  0         30h
   ocnrf-nfdiscovery-84965d4fb9-rjxg2      1/1    Running  0         30h
   ocnrf-nfdiscovery-94965d4fb9-rjxg3      1/1    Running  0         30h
   ocnrf-nfregistration-64f4d8f5d5-6q92j   1/1    Running  0         30h
   ocnrf-nfregistration-44f4d8f5d5-6q92i   1/1    Running  0         30h
   ocnrf-nfsubscription-5b6db965b9-gcvpf   1/1    Running  0         30h
   ocnrf-nfsubscription-4b6db965b9-gcvpe   1/1    Running  0         30h
   ocnrf-nrfauditor-67b676dd87-xktbm       1/1    Running  0         30h
   ocnrf-nrfconfiguration-678fddc5f5-c5htj 1/1    Running  0         30h
   ocnrf-appinfo-8b7879cdb-jds4r           1/1    Running  0         30h
   ```

4.2.1.1.2 Docker registry is configured incorrectly

Problem

`helm install` might fail if the docker registry is not configured in all primary and secondary nodes.

Error Code or Error Message

When `kubectl get pods -n <ocnrf_namespace>` is performed, the status of the pods might be ImagePullBackOff or ErrImagePull.

For example:

```
$ kubectl get pods -n ocnrf
```

```
NAME                                    READY  STATUS            RESTARTS  AGE
ocnrf-egressgateway-d6567bbdb-9jrsx     2/2    ImagePullBackOff  0         30h
ocnrf-egressgateway-d6567bbdb-ntn2v     2/2    Running           0         30h
ocnrf-ingressgateway-754d645984-h9vzq   2/2    Running           0         30h
ocnrf-ingressgateway-754d645984-njz4w   2/2    Running           0         30h
ocnrf-nfaccesstoken-59fb96494c-k8w9p    1/1    Running           0         30h
ocnrf-nfaccesstoken-49fb96494c-k8w9q    1/1    Running           0         30h
ocnrf-nfdiscovery-84965d4fb9-rjxg2      1/1    Running           0         30h
ocnrf-nfdiscovery-94965d4fb9-rjxg3      1/1    Running           0         30h
ocnrf-nfregistration-64f4d8f5d5-6q92j   1/1    Running           0         30h
ocnrf-nfregistration-44f4d8f5d5-6q92i   1/1    Running           0         30h
```

```
ocnrf-nfsubscription-5b6db965b9-gcvpf    1/1    Running          0         30h
ocnrf-nfsubscription-4b6db965b9-gcvpe    1/1    Running          0         30h
ocnrf-nrfauditor-67b676dd87-xktbm        1/1    Running          0         30h
ocnrf-nrfconfiguration-678fddc5f5-c5htj  1/1    Running          0         30h
ocnrf-appinfo-8b7879cdb-jds4r            1/1    Running          0         30h
```

Solution

Configure docker registry on all primary and secondary nodes. For more information on configuring the docker registry, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

4.2.1.1.3 Continuous Restart of Pods

Problem

`helm install` might fail if the MySQL primary and secondary hosts are not configured properly in `ocnrf-custom-values.yaml`.

Error Code/Error Message

When `kubectl get pods -n <ocnrf_namespace>` is performed, the pods restart count increases continuously.

For example:

```
$ kubectl get pods -n ocnrf
```

```
NAME                                     READY  STATUS   RESTARTS  AGE
ocnrf-egressgateway-d6567bbdb-9jrsx      2/2    Running  0         30h
ocnrf-egressgateway-d6567bbdb-ntn2v      2/2    Running  0         30h
ocnrf-ingressgateway-754d645984-h9vzq    2/2    Running  0         30h
ocnrf-ingressgateway-754d645984-njz4w    2/2    Running  2         30h
ocnrf-nfaccesstoken-59fb96494c-k8w9p     1/1    Running  0         30h
ocnrf-nfaccesstoken-49fb96494c-k8w9q     1/1    Running  0         30h
ocnrf-nfdiscovery-84965d4fb9-rjxg2       1/1    Running  0         30h
ocnrf-nfdiscovery-94965d4fb9-rjxg3       1/1    Running  0         30h
ocnrf-nfregistration-64f4d8f5d5-6q92j    1/1    Running  0         30h
ocnrf-nfregistration-44f4d8f5d5-6q92i    1/1    Running  0         30h
ocnrf-nfsubscription-5b6db965b9-gcvpf    1/1    Running  0         30h
ocnrf-nfsubscription-4b6db965b9-gcvpe    1/1    Running  0         30h
ocnrf-nrfauditor-67b676dd87-xktbm        1/1    Running  0         30h
ocnrf-nrfconfiguration-678fddc5f5-c5htj  1/1    Running  0         30h
ocnrf-appinfo-8b7879cdb-jds4r            1/1    Running  0         30h
```

Solution

MySQL servers(s) may not be configured properly according to the preinstallation steps. For configuring MySQL servers, see the "Configuring Database, Creating Users, and Granting Permissions" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

4.2.1.2 Custom Value File Parse Failure

This section explains troubleshooting procedure in case of failure while parsing `ocnrf-custom-values.yaml` file.

Problem

Unable to parse *ocnrf-custom-values-x.x.x.yaml*, while running `helm install`.

Error Code/Error Message

Error: failed to parse ocnrf-custom-values-x.x.x.yaml: error converting YAML to JSON: yaml

Symptom

While creating the *ocnrf-custom-values-x.x.x.yaml* file, if the aforementioned error is received, it means that the file is not created properly. The tree structure may not have been followed or there may also be tab spaces in the file.

Solution

Perform the following:

1. Download the latest NRF templates zip file from My Oracle Support. For more information, see the "Downloading NRF package" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.
2. Follow the steps mentioned in the "Installation Tasks" section in *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

4.2.2 Postinstallation

This section describes the common postinstallation related issues and their resolution steps.

4.2.2.1 Helm Test Error Scenarios

Following are the error scenarios that may be identified using helm test.

1. Run the following command to get the Helm Test pod name:

   ```
   kubectl get pods -n <deployment-namespace>
   ```

2. When a helm test is performed, a new helm test pod is created. Check for the Helm Test pod that is in an error state.
3. Get the logs using the following command:

   ```
   kubectl logs <podname> -n <namespace>
   ```

   Example:

   ```
   kubectl get <helm_test_pod> -n ocnrf
   ```

   For further assistance, collect the logs and contact MOS.

4.3 Upgrade or Rollback Failure

When NRF upgrade or rollback fails, perform the following procedure.

1. Check the pre or post upgrade logs or rollback hook logs in Kibana as applicable. Users can filter upgrade or rollback logs using the following filters:
   - For upgrade: lifeCycleEvent=9001
   - For rollback: lifeCycleEvent=9002

   ```
   {
      "time_stamp":"2021-08-23 06:45:57.698+0000",
   ```

```
   "thread":"main",
   "level":"INFO",

"logger":"com.oracle.cgbu.cne.ocnrf.hooks.releases.ReleaseHelmHook_1_14_1",
   "message":"{logMsg=Starting Pre-Upgrade hook Execution,
lifeCycleEvent=9001 | Upgrade, sourceRelease=101400,
targetRelease=101401}",

"loc":"com.oracle.cgbu.ocnrf.common.utils.EventSpecificLogger.submit(EventS
pecificLogger.java:94)"
}
```

2. Check the pod logs in Kibana to analyze the cause of failure.
3. After detecting the cause of failure, do the following:
   - For upgrade failure:
     - If the cause of upgrade failure is database or network connectivity issue, contact your system administrator. When the issue is resolved, rerun the upgrade command.
     - If the cause of failure occurs during the preupgrade phase, do not perform the roll back.
     - If the upgrade failure occurs during the postupgrade phase, for example, post upgrade hook failure due to target release pod not moving to ready state, then perform a rollback.
   - For rollback failure: If the cause of rollback failure is database or network connectivity issue, contact your system administrator. When the issue is resolved, rerun the rollback command.
4. If the issue persists, contact My Oracle Support.

4.4 Troubleshooting CDS

Service Operations responses doesn't contain Remote NRF Set Data

Following are scenarios where response doesn't contain remote NRF set data:

- CDS is down
- CDS unable to sync with the remote NRF set
- Incorrect Feature Configuration

CDS is down

1. When the CDS is down, the OcnrfCacheDataServiceDown alert is raised. All the NRF core microservices fall back to cnDBTier for serving the requests.
   In this case, the NRF instance has the local set georeplicated view and not the segment-level view.
2. Check the resolution steps to resolve the OcnrfCacheDataServiceDown alert.
3. Once the alert is cleared and CDS is in the Running state, the NRF core microservices connect to CDS to serve the requests.
4. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.

CDS unable to synchronize with the remote NRF set

1. If the CDS from a set is unable to synchronize the in-memory cache from the remote NRF's CDS, then the CDS attempts to reach healthy remote NRFs to synchronize the in-memory cache.
2. The retry attempt to the same remote NRF is performed based on the configuration in Egress Gateway.
3. The reroute from local NRF is based on the NRF Growth feature configuration. For more information about the feature configuration, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
4. If all the remote NRFs are not reachable, then the CDS from NRF uses the last known data from the remote set to serve the service requests.

Incorrect Feature Configuration

1. If the CDS from a set is unable to synchronize the in-memory cache from the remote NRF's CDS, then the CDS attempts to reach healthy remote NRFs to synchronize the in-memory cache.
2. Check the NRF Growth feature configuration as mentioned in the REST configuration. For more information about the feature configuration, see *Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide*.
3. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.

CDS Unreachable

1. Check for the OcnrfDatabaseFallbackUsed alert.
   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.
2. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.

CDS unable to synchronization with the local cnDBTier

1. If the CDS is unable to synchronize the data with the local cnDBTier, then the CDS marks itself as not in the ready state.
2. With CDS not being ready, the NRF core services mark itself as not ready forcing the NF consumers and producers to move to mated and healthy NRFs.
3. The CDS to CDS synchronization request also fails so that the NRFs in the peer set move to healthy NRFs for updated data synchronization.

NF Records present in NRF after Deregistration

1. Check for the following alerts:
   a. OcnrfRemoteSetNrfSyncFailed
   b. OcnrfSyncFailureFromAllNrfsOfAnyRemoteSet
   c. OcnrfSyncFailureFromAllNrfsOfAllRemoteSets
      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.
2. Check the `nrfHostConfigList` configuration in the local NRF set.

3. In case the issue persists, capture all the outputs for the above steps and contact My Oracle Support.

4.5 TLS Connection Failure

This section describes the TLS related issues and their resolution steps. It is recommended to attempt the resolution steps provided in this guide before contacting Oracle Support.

Problem: Handshake is not established between NRFs.

Scenario: When the client version is TLS 1.2 and the server version is TLS 1.3

Server Error Message

```
The client supported protocol versions[TLSv1.2] are not accepted by
      server preferences [TLSv1.3]
```

Client Error Message

```
Received fatal alert: protocol_version
```

Scenario: When the client version is TLS 1.3 and the server version is TLS1.2

Server Error Message

```
The client supported protocol versions[TLSv1.3]are not accepted by
      server preferences [TLSv1.2]
```

Client Error Message

```
Received fatal alert: protocol_version
```

Solution:

If the error logs have the SSL exception, do the following:

Check the TLS version of both NRFs, if both support different and single TLS versions, (that is, NRF1 supports TLS 1.2 only and NRF2 supports TLS 1.3 only or vice versa), handshake fails. Ensure that the TLS version is same for both NRFs or revert to default configuration for both NRFs. The TLS version communication supported are:

Table 4-1 TLS Version Used

Client TLS Version	Server TLS Version	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
TLS 1.2, TLS 1.3	TLS 1.3	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

Check the cipher suites being supported by both NRFs, it should be either the same or should have common cipher suites present. If not, revert to default configuration.

Problem: Pods not coming up after populating the `clientDisabledExtension` or `serverDisabledExtension` parameter.

Solution:

- Check the values given in the Helm parameters. The values listed cannot be added in these parameters:
  - supported_versions
  - key_share
  - supported_groups
  - signature_algorithms
  - pre_shared_key

If any of the above values is present, remove them or revert to default configuration for the pod to come up.
Problem: Pods not coming up after populating `clientSignatureSchemes` parameter.

Solution:

- Check the values given in the Helm parameters.
- Value listed below should not be removed from these parameters:
  - rsa_pkcs1_sha512
  - rsa_pkcs1_sha384
  - rsa_pkcs1_sha256

  If any of the above values is not present, add them or revert to default configuration for the pod to come up.

Problem: Connection Failure Due to Cipher Mismatch: NRF -Client and Producer Server for TLS 1.3

Scenario: The NRF client is configured to request a connection using TLS 1.3 with specific ciphers that are not supported by the producer server. As a result, the connection fails due to the cipher mismatch, preventing secure communication between the client and server.

Client Error Message

```
No appropriate protocol(protocol is disabled or cipher suites are
inappropriate)
```

Server Error Message

```
Received fatal alert: handshake failure
```

Solution:

- Ensure that the following cipher suites are configured for the NRF client to use with TLS 1.3:
  - TLS_AES_128_GCM_SHA256
  - TLS_AES_256_GCM_SHA384
  - TLS_CHACHA20_POLY1305_SHA256

Ensure that both the client and server have at least one common TLS 1.3 cipher configured.

- Verify TLS 1.3 for secure communication between the NRF- Client and the producer server to ensure that the issue has been resolved.

Problem: Connection Failure for TLS 1.3 Due to Expired Certificates.

Scenario: The NRF -Client is attempting to establish a connection using TLS 1.3, but the connection fails due to expired certificates. Specifically, the NRF -Client is presenting TLS 1.3 certificates that have passed their validity period, which causes the Producer server to reject the connection.

Client Error Message

```
Service Unavailable for producer due to Certificate Expired
```

Server Error Message

```
Received fatal alert: handshake failure
```

Solution:

- Verify the validity of the current certificate.
- If the certificate has expired, renew it or extend its validity.
- Attempt to establish a connection between the NRF client and the Producer server to confirm that the issue has been resolved.
- Verify the TLS 1.3 for secure communication.

5
NRF Alerts

This section includes information about the NRF alerts.

The following table describes the various alert levels generated by NRF:

Table 5-1 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.

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.

5.1 System Level Alerts

This section lists the system level alerts.

5.1.1 OcnrfNfStatusUnavailable

Table 5-2 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.

Table 5-2 (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

5.1.2 OcnrfPodsRestart

Table 5-3 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

Table 5-3 (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

5.1.3 NnrfNFManagementServiceDown

Table 5-4 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

Table 5-4 (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.

Table 5-4 (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

Table 5-4 (Cont.) NnrfNFManagementServiceDown

Field	Details
	Oracle Communications Cloud Native Core, Network Function Data Collector User Guide.
Available in OCI	No

5.1.4 NnrfAccessTokenServiceDown

Table 5-5 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.

Table 5-5 (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

5.1.5 NnrfNFDiscoveryServiceDown

Table 5-6 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

Table 5-6 (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

5.1.6 OcnrfRegistrationServiceDown

Table 5-7 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.

Table 5-7 (Cont.) OcnrfRegistrationServiceDown

Field	Details
OID	1.3.6.1.4.1.323.5.3.36.1.2.7021
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

5.1.7 OcnrfSubscriptionServiceDown

Table 5-8 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.

Table 5-8 (Cont.) OcnrfSubscriptionServiceDown

Field	Details
OID	1.3.6.1.4.1.323.5.3.36.1.2.7022
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

5.1.8 OcnrfDiscoveryServiceDown

Table 5-9 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.

Table 5-9 (Cont.) OcnrfDiscoveryServiceDown

Field	Details
OID	1.3.6.1.4.1.323.5.3.36.1.2.7023
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

5.1.9 OcnrfAccessTokenServiceDown

Table 5-10 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

Table 5-10 (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

5.1.10 OcnrfAuditorServiceDown

Table 5-11 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

Table 5-11 (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

5.1.11 OcnrfConfigurationServiceDown

Table 5-12 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

Table 5-12 (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

5.1.12 OcnrfAppInfoServiceDown

Table 5-13 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.

Table 5-13 (Cont.) OcnrfAppInfoServiceDown

Field	Details
OID	1.3.6.1.4.1.323.5.3.36.1.2.7027
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

5.1.13 OcnrfArtisanServiceDown

Table 5-14 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

Table 5-14 (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

5.1.14 OcnrfAlternateRouteServiceDown

Table 5-15 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

Table 5-15 (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*.

5.1.15 OcnrfPerfInfoServiceDown

Table 5-16 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.

Table 5-16 (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

5.1.16 OcnrfIngressGatewayServiceDown

Table 5-17 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.

Table 5-17 (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

5.1.17 OcnrfEgressGatewayServiceDown

Table 5-18 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.

Table 5-18 (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

5.1.18 OcnrfTotalIngressTrafficRateAboveMinorThreshold

Table 5-19 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'

Table 5-19 (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

5.1.19 OcnrfTotalIngressTrafficRateAboveMajorThreshold

Table 5-20 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'

Table 5-20 (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

5.1.20 OcnrfTotalIngressTrafficRateAboveCriticalThreshold

Table 5-21 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.

Table 5-21 (Cont.) OcnrfTotalIngressTrafficRateAboveCriticalThreshold

Field	Details
Available in OCI	No

5.1.21 OcnrfTransactionErrorRateAbove0Dot1Percent

Table 5-22 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

5.1.22 OcnrfTransactionErrorRateAbove1Percent

Table 5-23 OcnrfTransactionErrorRateAbove1Percent

Field	Details
Description	'Transaction Error rate is above 1 Percent of Total Transactions (current value is {{ $value }})'

Table 5-23 (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

5.1.23 OcnrfTransactionErrorRateAbove10Percent

Table 5-24 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'

Table 5-24 (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

5.1.24 OcnrfTransactionErrorRateAbove25Percent

Table 5-25 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'

Table 5-25 (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

5.1.25 OcnrfTransactionErrorRateAbove50Percent

Table 5-26 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'

Table 5-26 (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

5.1.26 OcnrfTotalEgressTrafficRateAboveCriticalThreshold

Table 5-27 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

5.1.27 OcnrfTotalForwardingTrafficRateAboveCriticalThreshold

Table 5-28 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

5.1.28 OcnrfTotalSLFRateAboveCriticalThreshold

Table 5-29 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

Table 5-29 (Cont.) OcnrfTotalSLFRateAboveCriticalThreshold

Field	Details
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

5.1.29 OcnrfTotalDiscoveryRateAboveCriticalThreshold

Table 5-30 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

5.2 Service Level Alerts

This section lists the service level alerts.

5.2.1 OcnrfAccessTokenRequestsRejected

Table 5-31 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'
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

5.2.2 OcnrfAuditorMultiplePodUnavailable

Table 5-32 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

5.2.3 OcnrfAppInfoMultiplePodUnavailable

Table 5-33 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.
Available in OCI	No

5.2.4 OcnrfPerfInfoMultiplePodUnavailable

Table 5-34 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

5.2.5 OcnrfAccessTokenRequestsAboveThreshold

Table 5-35 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.

Table 5-35 (Cont.) OcnrfAccessTokenRequestsAboveThreshold

Field	Details
OID	1.3.6.1.4.1.323.5.3.36.1.2.7115
Metric Used	ocnrf_accessToken_rx_requests_total
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

5.2.6 OcnrfNfUpdateRequestsAboveThreshold

Table 5-36 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

5.2.7 OcnrfNfHeartBeatRequestsAboveThreshold

Table 5-37 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

5.2.8 OcnrfRegisteredNfCountAboveThreshold

Table 5-38 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

5.2.9 OcnrfNfProfileSizeAboveThreshold

Table 5-39 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 smaller than the critical threshold. **Note**: The threshold is configurable in the alert file. **Step**: 1. Verify which NF has registered a nfProfile greater than the threshold size, using the `nfInstanceId` in the `ocnrf_nf_profile_size` metric.
Available in OCI	No

5.2.10 OcnrfDiscoveryResponseSizeAboveThreshold

Table 5-40 OcnrfDiscoveryResponseSizeAboveThreshold

Field	Details
Description	'The size of nfDiscover response 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 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

5.2.11 OcnrfTotalSubscriptionsAboveThreshold

Table 5-41 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
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

5.2.12 OcnrfDiscoveryRequestsForUDRAboveThreshold

Table 5-42 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

Table 5-42 (Cont.) OcnrfDiscoveryRequestsForUDRAboveThreshold

Field	Details
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

5.2.13 OcnrfDiscoveryRequestsForUDMAboveThreshold

Table 5-43 OcnrfDiscoveryRequestsForUDMAboveThreshold

Field	Details
Description	'Total NfDiscover request rate for nfType UDM 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 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

5.2.14 OcnrfDiscoveryRequestsForAMFAboveThreshold

Table 5-44 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

Table 5-44 (Cont.) OcnrfDiscoveryRequestsForAMFAboveThreshold

Field	Details
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

5.2.15 OcnrfDiscoveryRequestsForSMFAboveThreshold

Table 5-45 OcnrfDiscoveryRequestsForSMFAboveThreshold

Field	Details
Description	'Total NfDiscover request rate for nfType SMF 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 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

5.3 NfProfile Status Change Alerts

This section lists the alerts raised when there is status change in NfProfile.

5.3.1 OcnrfRegisteredPCFsBelowCriticalThreshold

Table 5-46 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

Table 5-46 (Cont.) OcnrfRegisteredPCFsBelowCriticalThreshold

Field	Details
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'
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

5.3.2 OcnrfRegisteredPCFsBelowMajorThreshold

Table 5-47 OcnrfRegisteredPCFsBelowMajorThreshold

Field	Details
Description	'The number of registered NFs detected below major threshold (current value is: {{ $value }})'

Table 5-47 (Cont.) OcnrfRegisteredPCFsBelowMajorThreshold

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 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.
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

5.3.3 OcnrfRegisteredPCFsBelowMinorThreshold

Table 5-48 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
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

5.3.4 OcnrfRegisteredPCFsBelowThreshold

Table 5-49 OcnrfRegisteredPCFsBelowThreshold

Field	Details
Description	'The number of registered NFs is approaching 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 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

5.3.5 OcnrfTotalNFsRegisteredBelowCriticalThreshold

Table 5-50 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

5.3.6 OcnrfTotalNFsRegisteredBelowMajorThreshold

Table 5-51 OcnrfTotalNFsRegisteredBelowMajorThreshold

Field	Details
Description	'Number of active registrations in OCNRF (current value is: {{ $value }}) is below major threshold'

Table 5-51 (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

5.3.7 OcnrfTotalNFsRegisteredBelowMinorThreshold

Table 5-52 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

Table 5-52 (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

5.3.8 OcnrfTotalNFsRegisteredApproachingMinorThreshold

Table 5-53 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'

Table 5-53 (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

5.3.9 OcnrfNFStatusTransitionToRegistered

Table 5-54 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

5.3.10 OcnrfNFServiceStatusTransitionToRegistered

Table 5-55 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}}'

Table 5-55 (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

5.3.11 OcnrfNFStatusTransitionToSuspended

Table 5-56 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

Table 5-56 (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

5.3.12 OcnrfNFServiceStatusTransitionToSuspended

Table 5-57 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

Table 5-57 (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

5.3.13 OcnrfNFStatusTransitionToUndiscoverable

Table 5-58 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

Table 5-58 (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

5.3.14 OcnrfNFServiceStatusTransitionToUndiscoverable

Table 5-59 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

Table 5-59 (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

5.3.15 OcnrfNFStatusTransitionToDeregistered

Table 5-60 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

Table 5-60 (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

5.3.16 OcnrfNFServiceStatusTransitionToDeregistered

Table 5-61 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

Table 5-61 (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

5.4 Feature Specific Alerts

This section lists the feature specific alerts.

5.4.1 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 AccessToken" section in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.1.1 OcnrfAccessTokenCurrentKeyIdNotConfigured

Table 5-62 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'

Table 5-62 (Cont.) OcnrfAccessTokenCurrentKeyIdNotConfigured

Field	Details
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*.
Available in OCI	No

5.4.1.2 OcnrfAccessTokenCurrentKeyIdInvalidDetails

Table 5-63 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

5.4.1.3 OcnrfOauthCurrentKeyNotConfigured

Table 5-64 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

Table 5-64 (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

5.4.1.4 OcnrfOauthCurrentKeyDataHealthStatus

Table 5-65 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

5.4.1.5 OcnrfOauthNonCurrentKeyDataHealthStatus

Table 5-66 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

Table 5-66 (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

5.4.1.6 OcnrfOauthCurrentCertificateExpiringIn1Week

Table 5-67 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

5.4.1.7 OcnrfOauthNonCurrentCertificateExpiringIn1Week

Table 5-68 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

Table 5-68 (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

5.4.1.8 OcnrfOauthCurrentCertificateExpiringIn30days

Table 5-69 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

5.4.1.9 OcnrfOauthNonCurrentCertificateExpiringIn30days

Table 5-70 OcnrfOauthNonCurrentCertificateExpiringIn30days

Field	Details
Description	'OCNRF Oauth Access token non current Key Id certificate is expiring in less than 30 days'

Table 5-70 (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

5.4.2 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 Based on Percentage Discards" section in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.2.1 OcnrfMemoryUsageCrossedMinorThreshold

Table 5-71 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.

Table 5-71 (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

5.4.2.2 OcnrfMemoryUsageCrossedMajorThreshold

Table 5-72 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

5.4.2.3 OcnrfMemoryUsageCrossedCriticalThreshold

Table 5-73 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

5.4.2.4 OcnrfOverloadThresholdBreachedL1

Table 5-74 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

Table 5-74 (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

5.4.2.5 OcnrfOverloadThresholdBreachedL2

Table 5-75 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.

Table 5-75 (Cont.) OcnrfOverloadThresholdBreachedL2

Field	Details
Available in OCI	Yes

5.4.2.6 OcnrfOverloadThresholdBreachedL3

Table 5-76 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

5.4.2.7 OcnrfOverloadThresholdBreachedL4

Table 5-77 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

Table 5-77 (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

5.4.3 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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.3.1 OcnrfDnsNaptrFailureResponseStatus

Table 5-78 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.

5.4.3.2 OcnrfAlternateRouteUpstreamDnsRetryExhausted

Table 5-79 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

5.4.4 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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.4.1 OcnrfNotificationRetryExhausted

Table 5-80 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

5.4.4.2 OcnrfNotificationFailureOtherThanRetryExhausted

Table 5-81 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

5.4.5 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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.5.1 OcnrfIngressGatewayDDUnreachable

Table 5-82 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

5.4.5.2 OcnrfEgressGatewayDDUnreachable

Table 5-83 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

5.4.6 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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.6.1 OcnrfSubscriptionGlobalCountWarnThresholdBreached

Table 5-84 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

5.4.6.2 OcnrfSubscriptionGlobalCountMinorThresholdBreached

Table 5-85 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

5.4.6.3 OcnrfSubscriptionGlobalCountMajorThresholdBreached

Table 5-86 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

Table 5-86 (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

5.4.6.4 OcnrfSubscriptionGlobalCountCriticalThresholdBreached

Table 5-87 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

5.4.6.5 OcnrfSubscriptionMigrationInProgressWarn

Table 5-88 OcnrfSubscriptionMigrationInProgressWarn

Field	Details
Description	The subscription migration is pending and subscriptionLimit feature is disabled

Table 5-88 (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.

5.4.6.6 OcnrfSubscriptionMigrationInProgressCritical

Table 5-89 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

5.4.7 Pod Protection Support for NRF Subscription Microservice Feature

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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.7.1 OcnrfPodInDangerOfCongestionState

Table 5-90 OcnrfPodInDangerOfCongestionState

Field	Details
Description	'The pod {{$labels.kubernetes_pod_name}} of service {{$labels.app_kubernetes_io_name}} is in Danger of Congestion state'

Table 5-90 (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

5.4.7.2 OcnrfPodPendingMessageCountInDangerOfCongestionState

Table 5-91 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

Table 5-91 (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.
Available in OCI	No

5.4.7.3 OcnrfPodInCongestedState

Table 5-92 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

Table 5-92 (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

5.4.7.4 OcnrfPodCpuUsageInCongestedState

Table 5-93 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

Table 5-93 (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

5.4.7.5 OcnrfPodCpuUsageInDangerOfCongestionState

Table 5-94 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

Table 5-94 (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

5.4.7.6 OcnrfPodPendingMessageCountInCongestedState

Table 5-95 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

Table 5-95 (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

5.4.8 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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.8.1 OcnrfOperationalStateCompleteShutdown

Table 5-96 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

5.4.8.2 OcnrfAuditOperationsPaused

Table 5-97 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

5.4.9 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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.9.1 OcnrfAllSCPsMarkedAsUnavailable

Table 5-98 OcnrfAllSCPsMarkedAsUnavailable

Field	Details
Description	'All SCPs have been marked unavailable.'
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

5.4.9.2 OcnrfSCPMarkedAsUnavailable

Table 5-99 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

5.4.10 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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.10.1 OcnrfCcaRootCertificateExpiringIn4Hours

Table 5-100 OcnrfCcaRootCertificateExpiringIn4Hours

Field	Details
Description	'The CCA Root Certificates expiring in 4 hours'.

Table 5-100 (Cont.) OcnrfCcaRootCertificateExpiringIn4Hours

Field	Details
Summary	'kubernetes_namespace: {{$labels.kubernetes_namespace}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : CCA Root Certificate is expiring in 4 Hours'
Severity	Critical
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

5.4.10.2 OcnrfCcaRootCertificateExpiringIn1Day

Table 5-101 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

5.4.10.3 OcnrfCcaRootCertificateExpiringIn5Days

Table 5-102 OcnrfCcaRootCertificateExpiringIn5Days

Field	Details
Description	'The CCA Root Certificates expiring in 5 days.'

Table 5-102 (Cont.) OcnrfCcaRootCertificateExpiringIn5Days

Field	Details
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.
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

5.4.11 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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.11.1 OcnrfDbReplicationStatusInactive

Table 5-103 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

5.4.11.2 OcnrfReplicationStatusMonitoringInactive

Table 5-104 OcnrfReplicationStatusMonitoringInactive

Field	Details
Description	'OCNRF Replication Status Monitoring Inactive'
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>`
Available in OCI	No

5.4.12 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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.12.1 OcnrfNfAuthenticationFailureRequestsRejected

Table 5-105 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

5.4.13 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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.13.1 OcnrfRemoteSetNrfSyncFailed

Table 5-106 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.

5.4.13.2 OcnrfSyncFailureFromAllNrfsOfAnyRemoteSet

Table 5-107 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

Table 5-107 (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

5.4.13.3 OcnrfSyncFailureFromAllNrfsOfAllRemoteSets

Table 5-108 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

5.4.13.4 OcnrfCacheDataServiceDown

Table 5-109 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

5.4.13.5 OcnrfDatabaseFallbackUsed

Table 5-110 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'

Table 5-110 (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

5.4.13.6 OcnrfTotalNFsRegisteredAtSegmentBelowMinorThreshold

Table 5-111 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
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_24.2.6.yaml` or `ocnrf_alertrules_promha_24.2.6.yaml` file depending on Prometheus version.

Table 5-111 (Cont.) OcnrfTotalNFsRegisteredAtSegmentBelowMinorThreshold

Field	Details
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

5.4.13.7 OcnrfTotalNFsRegisteredAtSegmentBelowMajorThreshold

Table 5-112 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
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_24.2.6.yaml` or `ocnrf_alertrules_promha_24.2.6.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

Table 5-112 (Cont.) OcnrfTotalNFsRegisteredAtSegmentBelowMajorThreshold

Field	Details
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

5.4.13.8 OcnrfTotalNFsRegisteredAtSegmentBelowCriticalThreshold

Table 5-113 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_24.2.6.yaml` or `ocnrf_alertrules_promha_24.2.6.yaml` file depending on Prometheus version.
OID	1.3.6.1.4.1.323.5.3.36.1.2.7105
Metric Used	ocnrf_nf_registered_count

Table 5-113 (Cont.) OcnrfTotalNFsRegisteredAtSegmentBelowCriticalThreshold

Field	Details
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*.
Available in OCI	No

5.4.14 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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.14.1 OcnrfIngressGatewayPodInDangerOfCongestionState

Table 5-114 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

Table 5-114 (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

5.4.14.2 OcnrfIngressGatewayPodInCongestedState

Table 5-115 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

5.4.14.3 OcnrfIngressGatewayPodCpuUsageInCongestedState

Table 5-116 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

5.4.14.4 OcnrfIngressGatewayPodCpuUsageInDangerOfCongestionState

Table 5-117 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

Table 5-117 (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

5.4.14.5 OcnrfIngressGatewayPodPendingMessageInCongestedState

Table 5-118 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

5.4.14.6 OcnrfIngressGatewayPodPendingMessageInDangerOfCongestionState

Table 5-119 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

5.4.15 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 Feature" section in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.15.1 OcnrfMaxSlfAttemptsExhausted

Table 5-120 OcnrfMaxSlfAttemptsExhausted

Field	Details
Description	'NF discovery request with fqdn {{$labels.NfProfileFqdn}} NF type {{$labels.NfType}} has exhausted maximum SLF attempts'

Table 5-120 (Cont.) OcnrfMaxSlfAttemptsExhausted

Field	Details
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.'
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

5.4.16 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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.16.1 OcnrfNFDiscoveryEmptyListObservedNotification

Table 5-121 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

Table 5-121 (Cont.) OcnrfNFDiscoveryEmptyListObservedNotification

Field	Details
Metric Used	ocnrf_nfDiscover_emptyList_total
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

5.4.17 Support for TLS Feature

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 in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

5.4.17.1 OcnrfTLSCertificateExpireMinor

Table 5-122 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

5.4.17.2 OcnrfTLSCertificateExpireMajor

Table 5-123 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

5.4.17.3 OcnrfTLSCertificateExpireCritical

Table 5-124 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

5.4.18 Egress Gateway Pod Throttling

5.4.18.1 OcnrfEgressPerPodDiscardRateAboveMajorThreshold

Table 5-125 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

5.4.18.2 OcnrfEgressPerPodDiscardRateAboveCriticalThreshold

Table 5-126 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

Table 5-126 (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

5.5 NRF Alert Configuration

NRF Alert Configuration

Follow the steps below for NRF Alert configuration in Prometheus:

> **Note**
>
> 1. The `Name` is the release name used in helm install command.
> 2. The `Namespace` is the namespace used in helm install command. By default `Namespace` for NRF is `ocnrf` that must be update as per the deployment.
> 3. The `ocnrf-config-1.1.0.0.0.zip` file can be downloaded from OHC. Unzip the `ocnrf-config-1.1.0.0.0.zip` package after downloading to get `NrfAlertrules.yaml` file.

1. Take Backup of current configuration map of Prometheus:

   ```
   kubectl get configmaps _NAME_-server -o yaml -n _Namespace_ > /tmp/
   tempConfig.yaml
   ```

2. Check and Add NRF Alert file name inside Prometheus configuration map:

   ```
   sed -i '/etc\/config\/alertsnrf/d' /tmp/tempConfig.yaml
   sed -i '/rule_files:/a\  \- /etc/config/alertsnrf' /tmp/tempConfig.yaml
   ```

3. Update configuration map with updated file name of NRF alert file:

   ```
   kubectl replace configmap _NAME_-server -f /tmp/tempConfig.yaml
   ```

4. 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)"
   ```

> **Note**
>
> Prometheus server takes updated configuration map that is automatically reloaded after 60 seconds approximately. Refresh the Prometheus GUI to confirm that the NRF Alerts are loaded.

5.5.1 Disable Alerts

This section explains the procedure to disable the alerts in NRF.

1. Edit **NrfAlertrules-24.2.6.yaml** file to remove a specific alert.
2. Remove complete content of a specific alert from the **NrfAlertrules-24.2.6.yaml** file. For example: If you want to remove `OcnrfTrafficRateAboveMinorThreshold` alert, remove the complete content:

   ```
   ## ALERT SAMPLE START##

         - alert: OcnrfTrafficRateAboveMinorThreshold
           annotations:
             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.

5.5.2 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-
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_24.2.6.mib
  This is considered as NRF top level mib file, where the objects and their data types are defined.
- ocnrf_mib_24.2.6.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_24.2.6.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*.