

Oracle® Communications

Cloud Native Core, Service Communication Proxy User Guide



Release 24.2.5

F97299-06

July 2025

ORACLE®

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

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

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

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

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

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

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

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

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

Contents

1	Introduction	
1.1	References	3
2	Service Communication Proxy Architecture	
2.1	5G SBI Message Processing and Trigger Points	3
3	Service Communication Proxy Features	
3.1	Support for Multiple 3gpp-Sbi-Binding Headers	1
3.2	Error Response Enhancement	2
3.2.1	SCP Error IDs	5
3.3	NFProfile Processing Enhancements	17
3.4	Support for cnDBTier APIs in CNC Console	18
3.5	NRF Configuration Using DNS SRV Resolution	20
3.6	Support for Kubernetes Resources	23
3.6.1	Dual Stack Support	23
3.6.2	Network Policies	24
3.6.3	Pod Disruption Budget	25
3.6.4	SCP Microservices and Probes	25
3.7	SCP Response Timeout Extension	34
3.8	Overload Control Based on the Overload Control Information Header	35
3.9	Custom Header Support at SCP	38
3.10	Support for HTTPS	39
3.10.1	TLS Certificate Configuration Enhancement	41
3.10.2	Support for TLS 1.3	42
3.10.3	Support for Automated Certificate Lifecycle Management	47
3.11	Routing Options Enhancement to Control SCP Routing Features	47
3.12	Enhanced NF Status Processing	48
3.13	Support for Default Notification Message Routing	50
3.13.1	Default Notification Routing Behavior at SCP	50
3.14	Location Header for Host Mismatch	83
3.15	Support for the Client Credentials Assertion Header	84
3.16	Support for 5G SBI Message Correlation using the 3gpp-Sbi-Correlation-Info Header	85

3.17	SCP Health Check API	86
3.18	Verbose Logging for SCP	89
3.19	Support for OAuth2.0	96
3.20	Pod Overload Control	99
3.21	Host Preference for Egress Message Requests	100
3.22	Enhanced 5G SBI Message Failure Handling	102
3.23	Extension of HTTP Status Code List for Rerouting	110
3.24	Load Balancing	111
3.25	5G SBI Message Routing	112
3.25.1	5G SBI Notification Requests Processing	115
3.25.2	5G SBI Service Requests Processing	120
3.26	5G SBI Message Mediation Support	123
3.26.1	Mediation Rules Configuration	124
3.26.1.1	Mediation Rule Migration Process	130
3.26.2	Mediation Log Level Configuration	136
3.27	Message Priority Assignment and Override	137
3.28	Outlier Detection	138
3.29	Circuit Breaking	139
3.30	Canary Release	140
3.31	Static Configuration of NF Profiles	141
3.32	Preventive Audit on Last NF Instance	142
3.33	Rate Limiting	143
3.33.1	Local Rate Limit Feature	143
3.33.2	Global Rate Limit Feature	144
3.33.3	Ingress Rate Limiting	146
3.33.4	Egress Rate Limiting	149
3.33.4.1	Global Egress Rate Limiting	150
3.33.5	Egress and Ingress Key Combinations	153
3.34	Support for Multus Container Network Interface	153
3.35	Support for Model C Indirect 5G SBI Communication	155
3.36	Multiple SCP Deployment in the Same Kubernetes Cluster	157
3.37	Support for SCP Triplet Deployment	158
3.38	Alternate Routing based on the DNS SRV Records	159
3.39	Support for Multiple Signaling Service IPs	161
3.40	Egress Congestion Control Support at SCP	163
3.41	Support for Model D Indirect 5G SBI Communication with Delegated Discovery	164
3.41.1	Managing Delegated Discovery Response Caching Based on Discovery Query Parameters	167
3.42	Support for 5G SBI Roaming	168
3.43	Routing Options Enhancements for Next Hop SCP and Notification or Callback Messages	171
3.44	Support for Alternate Routing Using Static Configuration	174

3.45	Support for Thread Watchdog	174
3.46	Message Feed Enhancements	175
3.47	Observability Enhancements for SCP Inter-microservice Connectivity	180
3.48	Load Control based on the Load Control Information Header	180
3.49	Support for Three-site Georedundancy	181
3.50	Automated Test Suite Support	182

4 Configuring Service Communication Proxy using the CNC Console

4.1	Support for Multicluster Deployment	1
4.2	CNC Console Interface	1
4.2.1	Configuring SCP Features	3
4.2.1.1	Configuring Error Response Enhancement	4
4.2.1.2	Configuring Enhanced NF Status Processing	4
4.2.1.3	Configuring Global Egress Rate Limiting	5
4.2.1.4	Configuring Support for 5G SBI Roaming	6
4.2.1.5	Configuring Mediation	7
4.2.1.6	Configuring Model D Indirect 5G SBI Communication feature	7
4.2.1.7	Configuring SCP User Agent Info	8
4.2.1.8	Configuring Message Feed	9
4.2.1.9	Configuring Load Control Information (LCI)	9
4.2.1.10	Configuring Host Preference for Egress Message Requests	11
4.2.1.11	Configuring CCA Header Validation	12
4.2.1.12	Configuring Location Header Update for Host Mismatch	14
4.2.1.13	Configuring Support for OAuth2.0	15
4.2.1.14	Configuring SCP Health Check API	17
4.2.1.15	NRF Configuration using DNS SRV Resolution	19
4.2.1.16	Configuring Overload Control Based on the Overload Control Information Header	20
4.2.1.17	Configuring Enhanced NFProfile Processing	23
4.2.2	Configuring OCI	23
4.2.3	Configuring OCI Threshold Levels	24
4.2.4	Viewing cnDBTier APIs in CNC Console	25
4.2.4.1	Backup List	25
4.2.4.2	cnDBTier Version	26
4.2.4.3	Database Statistics Report	26
4.2.4.4	Georeplication Recovery	27
4.2.4.5	Georeplication Status	29
4.2.4.6	HeartBeat Status	30
4.2.4.7	Local Cluster Status	30
4.2.4.8	On Demand Backup	30
4.2.4.9	cnDBTier Health	31

4.2.5	Configuring NRF SRV	34
4.2.5.1	Mapping NFInstanceId and ServiceInstanceId	35
4.2.6	Configuring NRF	36
4.2.7	Configuring Upgrade and Rollback Event	37
4.2.8	OAuth2.0 Configurations	38
4.2.8.1	Configuring Access Token Granularity	38
4.2.8.2	Configuring OAuth2.0 Local PLMN Required	39
4.2.9	Configuring Error Profile	40
4.2.10	Configuring Canary Release	42
4.2.11	Configuring Global Options	44
4.2.12	Configuring Mediation	47
4.2.12.1	Configuring Routing Options for Mediation	49
4.2.12.2	Configuring Mediation Rules	50
4.2.12.3	Configuring Mediation Log Levels	52
4.2.13	Configuring NF Rule Profile	54
4.2.13.1	Fetching NF Rule Profile Data	60
4.2.14	Configuring Routing Options	61
4.2.15	Configuring SCP Profile	62
4.2.16	Configuring Topology Source Info	64
4.2.17	Fetching Routing Rules	65
4.2.18	Configuring Ingress Rate Limiting	66
4.2.19	Configuring Egress Rate Limiting	67
4.2.20	Configuring Server Header	68
4.2.21	Configuring Pod Overload Control Policies	69
4.2.22	Configuring SBI Message Priority	70
4.2.23	Configuring DNS SRV based Alternate Routing	71
4.2.23.1	Configuring Routing Modes	72
4.2.23.2	Configuring the Default TTL Value	72
4.2.23.3	Configuring Alternate NF Group	73
4.2.23.4	Configuring Refresh DNS SRV Data	74
4.2.24	Configuring SBI Message Priority and Error Profiles for Egress Congestion Control	75
4.2.25	Configuring the nextHopSCP Routing Option	76
4.2.26	Configuring SEPP Inter PLMN Routing	78
4.2.27	Configuring Route Groups	80
4.2.28	Configuring Ingress Rate Limiter Consumer Info	81
4.2.29	Configuring SCP Services	83
4.2.30	Configuring Outlier Detection	86
4.2.31	Configuring Traffic Feed	87
4.2.32	Congestion Control Configurations	87
4.2.33	Routing Options Configurations	89
4.2.34	Routing Config Set	89

4.2.35	Circuit Breaking Configurations	90
4.2.36	Viewing SCP Feature Status	91
4.2.37	Get NF Discovery Response Cache Configuration Data	92
4.2.38	Configuring NF Delegated Discovery Response Cache	93

5 Metrics

5.1	SCP OCI Metrics Dashboard	1
5.2	Dimension Description	1
5.2.1	SCP Metrics	9
5.2.2	Mediation Metrics	114

6 Alerts

6.1	System level alerts	1
6.1.1	SCPNotificationPodMemoryUsage	1
6.1.2	SCPWorkerPodMemoryUsage	2
6.1.3	SCPInstanceDown	3
6.2	Application level alerts	3
6.2.1	SCPCcaFeatureEnabledWithoutHttps	3
6.2.2	SCPIngressTrafficRateAboveMinorThreshold	4
6.2.3	SCPIngressTrafficRateAboveMajorThreshold	4
6.2.4	SCPIngressTrafficRateAboveCriticalThreshold	5
6.2.5	SCPRoutingFailedForProducer	5
6.2.6	SCPAuditErrorResponse	6
6.2.7	SCPAuditEmptyNFArrayResponse	7
6.2.8	DuplicateLocalityFoundInForeignNF	8
6.2.9	ForeignNFLocalityNotServed	8
6.2.10	UnknownLocalityFoundInForeignNF	9
6.2.11	SCPUstreamResponseTimeout	9
6.2.12	SCPSingleNfInstanceAvailableForNFType	10
6.2.13	SCPNoNfInstanceForNFType	11
6.2.14	SCPIngressTrafficRateExceededConfiguredLimit	11
6.2.15	SCPIngressTrafficRoutedWithoutRateLimitTreatment	12
6.2.16	SCPEgressTrafficRateExceededConfiguredLimit	13
6.2.17	SCPEgressTrafficRoutedWithoutRateLimitTreatment	13
6.2.18	SCPNotificatoinRejectTopologySourceLocal	14
6.2.19	SCPNotificationProcessingFailureForNF	14
6.2.20	SCPSubscriptionFailureForNFType	16
6.2.21	SCPReSubscriptionFailureForNFType	18
6.2.22	SCPNrRegistrationFailureForRegionOrSetId	18
6.2.23	SCPNrHeartbeatFailureForRegionOrSetId	20

6.2.24	SCPDBOperationFailure	20
6.2.25	SCPGeneratedErrorsResponseForNFService	21
6.2.26	SCPCircuitBreakingAppliedForNF	21
6.2.27	SCPUUpgradeStarted	22
6.2.28	SCPUUpgradeFailed	22
6.2.29	SCPUUpgradeSuccessful	23
6.2.30	SCPRollbackStarted	23
6.2.31	SCPRollbackFailed	23
6.2.32	SCPRollbackSuccessful	24
6.2.33	ScpWorkerPodCpuUtilizationAboveWarnThreshold	25
6.2.34	ScpWorkerPodCpuUtilizationAboveMinorThreshold	25
6.2.35	ScpWorkerPodCpuUtilizationAboveMajorThreshold	26
6.2.36	ScpWorkerPodCpuUtilizationAboveCriticalThreshold	27
6.2.37	SCPUnhealthyPeerSCPDetected	27
6.2.38	SCPDnsSrvQueryFailure	28
6.2.39	SCPProducerOverloadThrottled	29
6.2.40	SCPProducerOverloadAlternateRouted	29
6.2.41	SCPSeppNotConfigured	30
6.2.42	SCPSeppRoutingFailed	31
6.2.43	SCPGlobalEgressRLRemoteParticipantConnectivityFailure	31
6.2.44	SCPGlobalEgressRLRemoteParticipantWithDuplicateNFInstanceId	32
6.2.45	SCPMediationConnectivityFailure	33
6.2.46	SCPNotificationQueuesUtilizationAboveMinorThreshold	34
6.2.47	SCPNotificationQueuesUtilizationAboveMajorThreshold	35
6.2.48	SCPNotificationQueuesUtilizationAboveCriticalThreshold	36
6.2.49	SCPNrProxyQueuesUtilizationAboveMinorThreshold	36
6.2.50	SCPNrProxyQueuesUtilizationAboveMajorThreshold	37
6.2.51	SCPNrProxyQueuesUtilizationAboveCriticalThreshold	38
6.2.52	SCPWorkerQueuesUtilizationAboveMinorThreshold	38
6.2.53	SCPWorkerQueuesUtilizationAboveMajorThreshold	39
6.2.54	SCPWorkerQueuesUtilizationAboveCriticalThreshold	40
6.2.55	SCPCacheQueuesUtilizationAboveMinorThreshold	40
6.2.56	SCPCacheQueuesUtilizationAboveMajorThreshold	41
6.2.57	SCPCacheQueuesUtilizationAboveCriticalThreshold	42
6.2.58	SCPLoadManagerQueuesUtilizationAboveMinorThreshold	42
6.2.59	SCPLoadManagerQueuesUtilizationAboveMajorThreshold	43
6.2.60	SCPLoadManagerQueuesUtilizationAboveCriticalThreshold	44
6.2.61	SCPProducerNfSetUnhealthy	44
6.2.62	SCPPeerSeppUnhealthy	45
6.2.63	SCPMicroServiceUnreachable	46
6.2.64	SCPTrafficFeedSendFailed	47
6.2.65	SCPTrafficFeedKafkaClusterUnhealthy	47

6.2.66	SCPTrafficFeedPartitionUnhealthy	48
6.2.67	SCPServiceMeshFailure	48
6.2.68	SCPHealthCheckFailedForPeerSCP	49
6.2.69	SCPHealthCheckFailed	49
6.2.70	ScpWorkerPodPendingTransUtilizationAboveMinorThreshold	50
6.2.71	ScpWorkerPodPendingTransUtilizationAboveMajorThreshold	50
6.2.72	ScpWorkerPodPendingTransUtilizationAboveCriticalThreshold	51
6.2.73	ScpWorkerPodPendingTransUtilizationAboveWarnThreshold	52
6.2.74	ScpWorkerPodResourceUtilizationAboveMinorThreshold	52
6.2.75	ScpWorkerPodResourceUtilizationAboveMajorThreshold	53
6.2.76	ScpWorkerPodResourceUtilizationAboveWarnThreshold	53
6.2.77	ScpWorkerPodResourceUtilizationAboveCriticalThreshold	54
6.2.78	SCPDNSSRVNRFMigrationTaskFailure	54
6.2.79	SCPDNSSRVNRFNonMigrationTaskFailure	55
6.2.80	SCPDNSSRVNRFDuplicateTargetDetected	56
6.2.81	SCPHighResponseTimeFromProducer	56
6.2.82	SCPCGroupVersionDetectionFailed	57
6.2.83	SCPCPUUsageFileReadFailed	57
6.2.84	SCPIgnoreUnknownService	58
6.2.85	SCPWorkerSSLCertificateOnCriticalExpiry	58
6.2.86	SCPWorkerSSLCertificateOnMajorExpiry	59
6.2.87	SCPWorkerSSLCertificateOnMinorExpiry	59
6.2.88	SCPIngressConnectionEstablishmentFailure	60
6.2.89	SCPEgressConnectionEstablishmentFailure	60
6.2.90	SCPNrfProxyOAuthQueuesUtilizationAboveCriticalThreshold	61
6.2.91	SCPNrfProxyOAuthQueuesUtilizationAboveMajorThreshold	61
6.2.92	SCPNrfProxyOAuthQueuesUtilizationAboveMinorThreshold	62
6.3	Configuring Alerts	62
6.3.1	Applying Alerts Rule to CNE without Prometheus Operator	62
6.3.2	Applying Alerts Rule to CNE with Prometheus Operator	64
6.3.3	Configuring Service Communication Proxy Alert using the SCPAlertrules.yaml file	64
6.3.4	Configuring Alert Manager for SNMP Notifier	65
6.4	Configuring SCP Alerts for OCI	67

7 KPIs and Traces

7.1	Key Performance Indicators	1
7.1.1	Platform KPIs	1
7.1.2	Application KPIs	2
7.1.3	Feature KPIs	5

8 SCP Interactions with NRF

9 Supported Query Parameters

10 Supported NF Types and Services

11 Supported 3GPP Headers

A HTTP Status Code and Applicability for Rerouting

B Error Code Generated by SCP

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

Acronym	Description
0-RTT	Zero Round Trip Time
3GPP	3rd Generation Partnership Project
5GC	5G Core Network
5GS	5G System
AEAD	Authenticated Encryption with Associated Data
AF	Application Function
API	Application Programming Interface
AUSF	Authentication Server Function
CCA	Client Credentials Assertion, 3gpp-Sbi-Client-Credentials header is defined by 3GPP. The header contains a client credentials assertion.
CNC Console	Oracle Communications Cloud Native Configuration Console
CNE	Oracle Communications Cloud Native Core, Cloud Native Environment
CNI	Container Network Interface
CRD	Custom Resource Definition
DNS	Domain Name System
FQDN	Fully Qualified Domain Name
GPSI	Generic Public Subscription Identifier
HPLMN	Home Public Land Mobile Network
HR	Home Routing
HTTP	Hypertext Transfer Protocol
HTTPS	Hypertext Transfer Protocol Secure
IMSI	International Mobile Subscriber Identity
IWF	InterWorking and Mediation Function
K8s	Kubernetes
LBO	Local Breakout
MAC	Message Authentication Code
MCC	Mobile Country Code
MNC	Mobile Network Code
MSISDN	Mobile Station International Subscriber Directory Number
NAD	Network Attachment Definition
NF	Network Function
NRF	Oracle Communications Cloud Native Core, Network Repository Function
OAM	Operations, Administration, and Maintenance
OCCEM	Oracle Communications Cloud Native Core, Certificate Management
OCI	Oracle Cloud Infrastructure

Table (Cont.) Acronyms

Acronym	Description
OCNADD	Oracle Communications Network Analytics Data Director
PFS	Provide Forward Secrecy
PLMN	Public Land Mobile Network
PSK	Pre-Shared Keys
QoS	Quality of Service
RTT	Round Trip Time
SBA	Service Based Architecture
SBI	Service Based Interface
SCP	Oracle Communications Cloud Native Core, Service Communication Proxy
SCPC	Service Communication Proxy Control
SEPP	Oracle Communications Cloud Native Core, Security Edge Protection Proxy
SMF	Session Management Function
SPN	Service Proto Name
SSL	Secure Sockets Layer
SUPI	Subscription Permanent Identifier
SRV	Service
TLS	Transport Layer Security
UDR	Oracle Communications Cloud Native Core, Unified Data Repository
UDSF	Unstructured Data Storage Function
URI	Uniform Resource Identifier
UDM	Unified Data Management
VPLMN	Visited Public Land Mobile Network
XFCC	x-forwarded-client-cert is a proxy header that indicates TLS certificate information.
LCI	Load Control Information
LC-H	Load Control based on LCI Header
OCI	Overload Control Information
OLC-H	Overload Control based on OCI Header

What's New in This Guide

This section introduces the documentation updates for Release 24.2.x.

Release 24.2.5 - F97299-06, July 2025

Updated the release number to 24.2.5 throughout the document.

Release 24.2.4 - F97299-05, April 2025

Updated the release number to 24.2.4 throughout the document.

Release 24.2.3 - F97299-04, January 2025

Updated the release number to 24.2.3 throughout the document.

Release 24.2.2 - F97299-03, November 2024

Updated the release number to 24.2.2 throughout the document.

Release 24.2.1 - F97299-02, September 2024

Updated the release number to 24.2.1 throughout the document.

Release 24.2.0 - F97299-01, July 2024

- Updated the following sections for the Support for TLS 1.3 feature:
 - Added the [Support for TLS 1.3](#) section to describe the feature.
 - Added the following metrics in the [Metrics](#) section:
 - * Added the `ocscp_worker_https_ingress_connection_success_total` metric.
 - * Renamed the `ocscp_metric_https_ingress_connection_failure_total` metric to `ocscp_worker_https_ingress_connection_failure_total` and updated the dimension.
 - * Added the `ocscp_worker_https_egress_connection_success_total` metric.
 - * Renamed the `ocscp_metric_https_egress_connection_failure_total` metric to `ocscp_worker_https_egress_connection_failure_total` and updated the dimension.
 - * Added the `ocscp_metric_ssl_certificate_expire_total` metric.
 - Added the following alerts in the [Alerts](#) section:
 - * [SCPWorkerSSLCertificateOnCriticalExpiry](#)
 - * [SCPWorkerSSLCertificateOnMajorExpiry](#)
 - * [SCPWorkerSSLCertificateOnMinorExpiry](#)
 - * [SCPIngressConnectionEstablishmentFailure](#)
 - * [SCPEgressConnectionEstablishmentFailure](#)
 - Removed the following alerts from the [Alerts](#) section:
 - * `SCPWorkerHTTPSConnectionFailure`
 - * `SCPWorkerSSLCertificateExpire`

- Added the `ocscp_worker_5gsbi_upstream_connect_failure_total` metric in the [SCP Metrics](#) section.
- Updated the [Enhanced 5G SBI Message Failure Handling](#) section to inform about SCP generated error responses.
- Updated the [Verbose Logging for SCP](#) section to include verbose logging details for the SCPC-Notification microservice in the control plane.
- Added the [Fetching NF Rule Profile Data](#) section to retrieve the default notification subscription data for a specific NF rule profile.
- Added the following metrics in the [SCP Metrics](#) section to convert gauge metrics to counter metrics:
 - `ocscp_worker_pod_overload_control_cpu_utilization_total`
 - `ocscp_worker_self_oci_above_oci_conveyance_threshold_total`
- Added the `ocscp_nrfproxy_oauth_queue_alert` metric and the following alerts as part of the OAuth2.0 feature:
 - [SCPNrfProxyOauthQueuesUtilizationAboveCriticalThreshold](#)
 - [SCPNrfProxyOauthQueuesUtilizationAboveMajorThreshold](#)
 - [SCPNrfProxyOauthQueuesUtilizationAboveMinorThreshold](#)
- Added the `nnrf-oauth2` service type to the NRF NF in [Table 10-1](#).
- Removed the following service types for PCF NF from [Table 10-1](#):
 - `ampolicy`
 - `bdtpolicycontrol`
 - `smpolicy`
 - `ue-policy`
- Updated the Resource URIs in the [SCP Health Check API](#) section.
- Updated the [SCP Error IDs](#) section as part of improvement activity:
 - Added the following error codes:
 - * `OSCP-WRK-ROUTE-E007`
 - * `OSCP-OAU-TKNRSP-E003`
 - Updated the description of the following error codes:
 - * `OSCP-WRK-OAUTH-E001`
 - * `OSCP-WRK-OAUTH-E002`
 - * `OSCP-WRK-DELDISC-E002`
 - * `OSCP-WRK-OAUTH-E006`
 - * `OSCP-NPY-DISREQ-E002`
 - * `OSCP-OAU-TKNREQ-E002`
 - Removed the following error codes:
 - * `OSCP-WRK-OAUTH-E007`
 - * `OSCP-WRK-DELDISC-E003`
 - * `OSCP-NPY-DISREQ-E003`
 - * `OSCP-OAU-TKNREQ-E003`

- Renamed the `ocscp_producer_nf_instance_id` dimension to `ocscp_peer_nf_instance_id` and updated the descriptions of the following metrics in the [SCP Metric](#) section:
 - `ocscp_metric_nf_lci_tx_total`
 - `ocscp_metric_scp_lci_tx_total`
 - `ocscp_metric_nf_lci_rx_total`
- Updated the `ocscp_service_instance_id` dimension to `ocscp_producer_service_instance_id` for the "ocscp_metric_attempts_to_forward_route_total" metric in the [SCP Metric](#) section.
- Updated the definitions of the metrics `ocscp_metric_interscp_http_tx_req_total` and `ocscp_metric_interscp_http_rx_res_total` in the [SCP Metric](#) section.
- Updated the cause value for the following metrics [SCP Metric](#) section:
 - `ocscp_alternate_resolution_dnssrv_res_total`
 - `ocscp_alternate_resolution_dnssrv_rx_error_res`
- Updated the cause value of the [SCPDnsSrvQueryFailure](#) alert.
- Added the support for additional standard HTTP2 headers in the [Support for Multiple 3gpp-Sbi-Binding Headers](#) section.
- Added a note regarding the "ocscp_nf_service_type" dimension for notifications in the [Metrics](#) section.
- Added Georeplication Recovery API to mark cnDBTier cluster as failed, start georeplication recovery, and monitor their status in the [Support for cnDBTier APIs in CNC Console](#) and [Georeplication Recovery](#) sections.
- Updated the description of the NRF Configuration Using DNS SRV Resolution feature with the InterPLMN routing details in the [NRF Configuration Using DNS SRV Resolution](#) section.
- Replaced the dimension "queue_name" with queueName throughout the metrics in the [Metrics](#) section.
- Renamed the `ocscp_producer_load_throttled` metric to `ocscp_load_manager_peer_load_throttled_threshold` and updated the description in the [Metrics](#) section.
- Renamed the `ocscp_producer_load_alternateRoute` metric to `ocscp_load_manager_peer_load_alternateRoute_threshold` and updated the description in the [Metrics](#) section.
- Updated the metric type and definition of `ocscp_med_active_rule_count` and `ocscp_med_total_rule_count` in the [Mediation Metrics](#) section.
- Replaced the `ocscp_producer_fqdn` dimension with the `ocscp_producer_host` dimension in the [Alerts](#) and [Metrics](#) section.
- Added a note about pegging metrics for both HTTP and HTTPS traffic in the [Metrics](#) section.
- Removed the Routing Support for NEF Services section and added NEF services information in the [Supported NF Types and Services](#) section.
- Removed the "3gpp-Sbi-Nrf-Uri" header and added the "3gpp-Sbi-Oci" header to the [Supported 3GPP Headers](#) section.
- Updated the [Configuring Mediation Log Levels](#) section with the description of the "OFF" log level.

- Updated the [Support for Alternate Routing Using Static Configuration](#) feature with details about prioritization in the static routing configuration.

1

Introduction

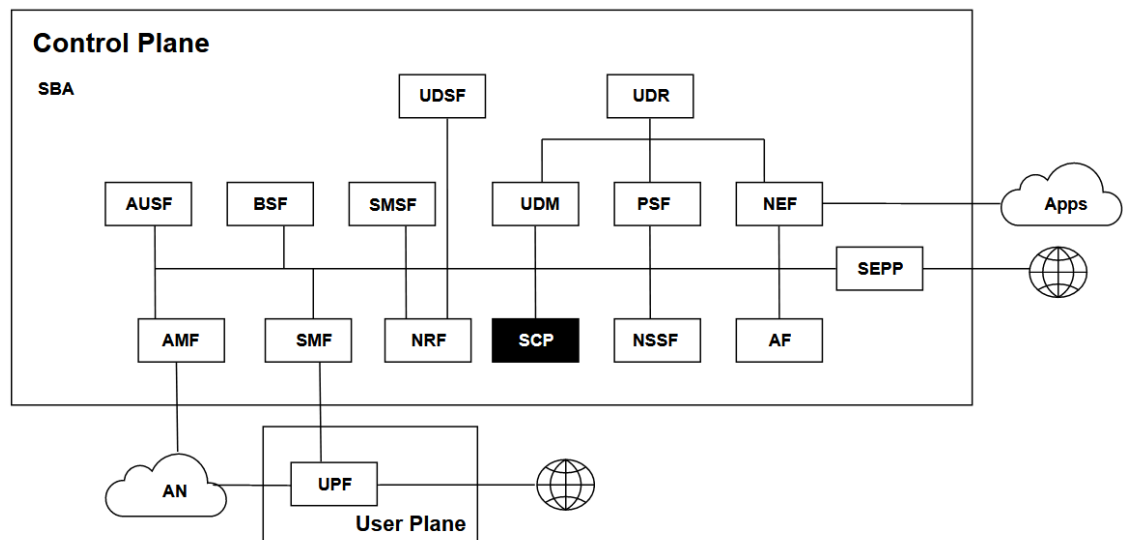
This document provides information about how to use Oracle Communications Cloud Native Core, Service Communication Proxy (SCP) in the cloud native 5G core network.

SCP installation is supported over the following platforms:

- Oracle Communications Cloud Native Core, Cloud Native Environment (CNE)
- Oracle Cloud Infrastructure (OCI)
For more information, see *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

SCP is a 3GPP-defined network entity that works as an HTTP/HTTPS proxy in the 5G SBA control plane. 5G NFs communicate indirectly via the SCP in the indirect communication models.

Figure 1-1 Service Based Architecture (SBA) for 5G Core



The core network in 5G follows a service-based architecture, where network functions provide services that can be utilized using REST APIs by other functions. This allows the adoption of web-scale technologies and software that are used by different organizations in the telecommunications network.

SCP provides the following functionalities in the 5G core:

- **Routing/Selection:** Routing rules, refresh cache, and handle application failures and redirects.
 - **Dynamic Discovery:** The 5G topology is learned from Network Repository Function (NRF) and routing rules are created automatically based on the learned NF profiles.
 - **Static Configuration:** SCP allows the configuration of NF profiles.

- **Load Balancing:** Load balancing based on NF set, priority, and capacity published in the NF profile.
- **NF Subscription:** Subscription for all NF types to learn the network topology.
- **Circuit Breaking:** Pro-active method to detect the degraded producer NF instances in the network. Implemented on a per FQDN basis when outstanding transactions exceed a configurable value.
- **Message Priority:** Message Priority assignment and override based on the 3GPP-SBI-Message-Priority header.
- **Congestion Control:** Congestion control for producer NF instances based on their published NF load.
- **Overload Control:** Avoids SCP failures by providing overload control for SCP-worker pods based on CPU usage. Uniform load balancing and routing strategy across the network and protects the pod from overload related to various system resources.

SCP Availability

Oracle Communications Cloud Native Core, Service Communication Proxy (SCP) availability depends on many factors. SCP applications are designed to achieve 99.999% availability according to the applicable Telecommunications Industry Association TL9000 standards with the following deployment requirements:

- Deploy on a Cloud Native Environment with at least 99.999% availability.
- Deploy with $n + k$ (up to 2) application redundancy, where k is greater than or equal to one.
- Maintain production software within $n-3$ software releases, where n is the current general availability release.
- Apply bug fixes, critical patches, and configuration recommendations provided by Oracle promptly.
- Maintain fault recovery procedures external to the applications for the reconstruction of lost or altered files, data, programs, or Cloud Native environment.
- Install, configure, operate, and maintain SCP as per Oracle's applicable installation, operation, administration, and maintenance specifications.
- Maintain an active support contract and provide access to the deployed SCP and your personnel to assist Oracle in addressing any outage.

Note

The performance and capacity of the SCP system may vary based on the call model, feature or interface configuration, network conditions, and underlying CNE and hardware environment.

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

Table 1-1 Measuring SCP Availability

Availability	Description
Planned Product Availability	(Product available time in each month) less (Excluded Time (defined below) in each month).

Table 1-1 (Cont.) Measuring SCP Availability

Availability	Description
Actual Product Availability	(Planned Product Availability) less (any Unscheduled Outage)
Product Availability Level	(Actual Product Availability across all Production instances divided by Planned Product Availability across all Production instances) x 100

Note**Excluded Time** means:

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

1.1 References

- *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*
- *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*
- *Oracle Communications Cloud Native Core, Automated Test Suite Guide*
- *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Network Repository Function User Guide*
- *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Cloud Native Environment User Guide*
- *Oracle Communications Network Analytics Data Director User Guide*
- *Oracle Communications Cloud Native Core, Certificate Management Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Certificate Management User Guide*
- *Oracle Communications Cloud Native Core, OCI Adaptor Deployment Guide*

- <https://github.com/k8snetworkplumbingwg/multus-cni>.

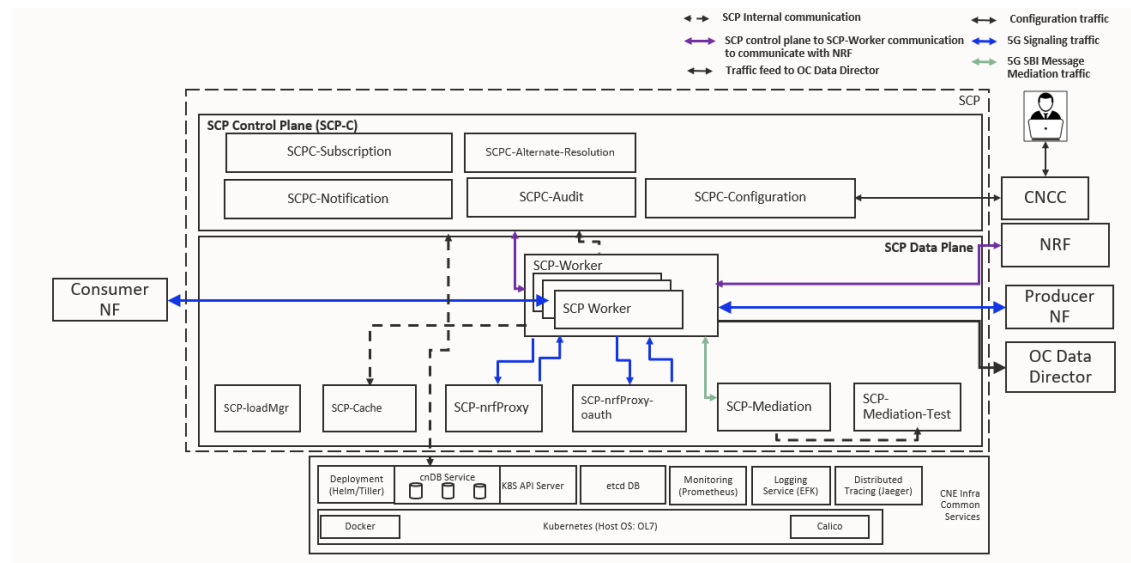
2

Service Communication Proxy Architecture

This section explains the Oracle Communications Cloud Native Core, Service Communication Proxy (SCP) system architecture.

SCP is a decentralized solution composed of control plane and data plane. This solution is deployed with 5G Network Functions (NFs) for providing routing control, resiliency, and observability to the core network.

Figure 2-1 Service Communication Proxy Architecture



The SCP solution is deployed either as a default outbound proxy to NF instances or as a router model, where SCP is configured as HTTP2 outbound proxy at each NF in cloud native environments.

SCP Microservices

The list of SCP microservices is as follows:

Table 2-1 SCP Microservices

Microservices	Description
SCPC-Notification	It configures routing rules as per 5G NF topology information obtained from NRF, or as result of audit. It tracks the 5G NF topology for any change and updates in the route rules or configuration. It translates the topology into low-level SCP-Worker configuration and provides the configuration to the Worker when requested.

Table 2-1 (Cont.) SCP Microservices

Microservices	Description
SCPC-Audit	It performs periodic audit with NRF to synchronize the 5G NF topology information and updates routing rules through notifications if required.
SCPC-Configuration	It provides an interface for SCP configuration.
SCPC-Subscription	It performs NRF management and NRF discovery service operation for SCP registration and subscription for NF change notification.
SCPC-Alternate-Resolution	It finds an alternate NF service by performing DNS SRV queries with the DNS server and receives the DNS SRV responses. - Refreshes DNS SRV records to get update in DNS SRV records. - Provides DNS SRV records to scp-worker when required.
SCP-Worker	It receives 5G signaling traffic and controls communication and routing among 5G NFs and services. 5G NF selection for routing and rerouting across available 5G NFs - Endpoint selection based on NF profile information registered with NRF - Congestion control and traffic throttling using Ingress and Egress - Load balancing and overload control 5G SBI message priority assignment 5G SBI Mediation 5G Observability data
SCP-NrfProxy	It operates as an intermediate proxy between SCP-Worker and NRF for NF Discovery service requests originated by SCP. - It stores NF Discovery responses received from NRF through SCP-Worker. - It receives NF Discovery requests, along with query parameters from SCP-Worker, and forwards the created NF Discover requests to NRF through SCP-Worker. After receiving the NF Discovery responses from NRF through SCP-Worker, the nrfproxy service sends the response to SCP-Worker and stores the response for subsequent use.
SCP-Cache	It maintains the present status of ingress and egress traffic rate limiting and serves to the SCP-Worker microservice by storing the keys of rate limiting when required.
SCP-Mediation	It allows the modification of 5G SBI message content such as HTTP2 header values or JSON message body or the both, based on user-defined mediation rule sets to solve any inter-op issues. It applies user configured message mediation policy rules on ingress 5G SBI messages to execute message mediation transformation.
SCP-Mediation-Test	It enables to test the user configured message mediation policy test rules on ingress 5G SBI messages without actual execution on messages and mediation transformation.

Table 2-1 (Cont.) SCP Microservices

Microservices	Description
SCP-LoadManager	It manages the load and overload control information of peer NFs and SCPs on its own. It determines the latest load information from NRF or the LCI header using timestamp and shares the updated information with SCP-Worker pods using distributed cache. SCP-Worker pods process the received LCI information, update the routing rules, and use it for 5G SBI request routing and selection.
SCP-NrfProxy-Oauth	It is an intermediary proxy between SCP-Worker and NRF for processing OAuth2.0 access token service requests. It caches the OAuth2.0 access token received from NRF through SCP-Worker. It receives the OAuth2.0 access token request from SCP-Worker and forwards the created OAuth2.0 access token service request to NRF through SCP-Worker. After receiving the OAuth2.0 access token response from NRF through SCP-Worker, SCP-nrfProxy-Oauth sends the response to SCP-Worker and caches the response for subsequent use.

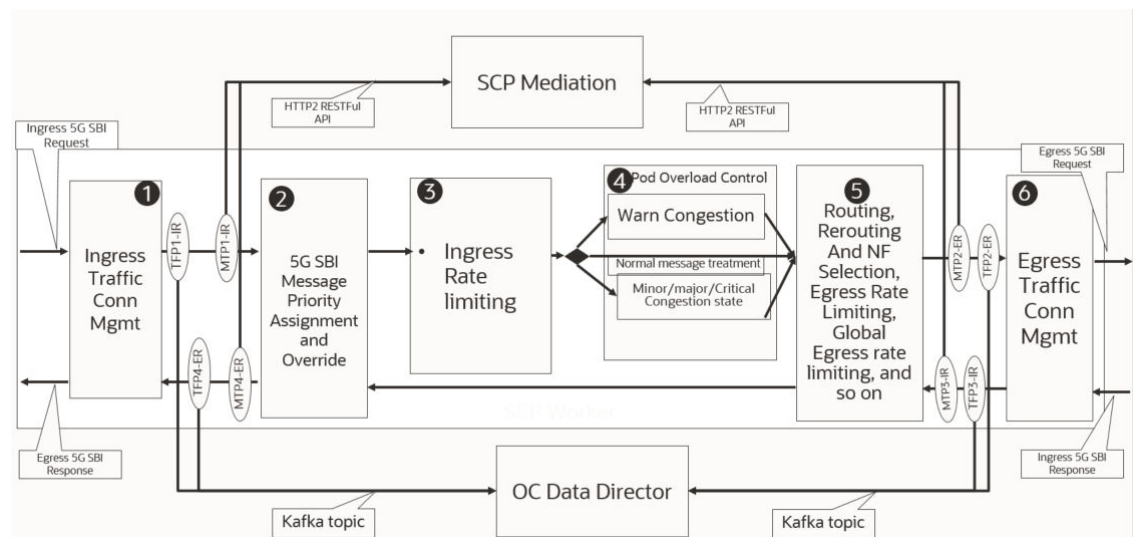
Ingress and Egress Gateway Traffic Management

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

2.1 5G SBI Message Processing and Trigger Points

The following image represents how SCP processes ingress 5G SBI message requests from consumer NFs.

Figure 2-2 5G SBI Message Processing and Trigger Points



3

Service Communication Proxy Features

This chapter describes the features of Oracle Communications Cloud Native Core, Service Communication Proxy (SCP).

Note

The performance and capacity of the SCP system may vary based on the call model, feature or interface configuration, network conditions, and underlying CNE and hardware environment.

3.1 Support for Multiple 3gpp-Sbi-Binding Headers

SCP enhanced its capabilities to support multiple 3GPP-SBI-binding headers within a single Service-Based Interface (SBI) message. In accordance with the support for additional standard HTTP2 headers specified in 3GPP TS 29.500, the SCP routes SBI request and response messages that contain multiple 3GPP-SBI-binding headers. It ensures the headers in the messages are forwarded to the producer exactly as received, without any additional processing or interpretation of the headers. The request and response messages includes:

- Notification messages
- Service messages

The feature supports the following header formats:

- Simple key-value pairs: k=v1, k=v2
- Array format: k=[v1,v2]

This feature is applicable to both Model C and Model D call flows and is implemented for the R16 deployment only.

Managing Support for Multiple 3gpp-Sbi-Binding Headers

Enable

Not applicable

Configure

Not applicable

Observe

There is no specific metrics and alerts required for the Multiple 3gpp-Sbi-Binding Headers.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.2 Error Response Enhancement

SCP used to send error responses to consumer NFs while processing message requests from or to consumer NFs. SCP used to generate these error responses in conditions, such as connection failure, timeout, an unknown host at SCP, and so on. The error responses generated by SCP comprised the error description in the detail parameter of ProblemDetails, which was inadequate to troubleshoot an error.

The following sample ProblemDetails displays various parameters:

```
{
  "title": "Service Unavailable",
  "status": "503",
  "detail": "Currently Congested. Could not process request.",
  "cause": "NF_CONGESTION"
}
```

With the enhanced error response mechanism, SCP sends additional information, such as server FQDN, NF service name, vendor name, and error ID, in the detail parameter of ProblemDetails to find the specific cause of the error. This helps in error identification and reduces troubleshooting time.

SCP has enhanced ProblemDetails for all the SCP generated 5G SBI message responses to identify the cause of the errors and troubleshoot them in the production environment.

The detail parameter in ProblemDetails includes information about problem details and error IDs in the following format, which is applicable only for SCP generated error responses:

<SCP FQDN>: <SCP specific problem detail>: <OSCP-scp error Id>

Where,

- <SCP FQDN> is the FQDN of SCP.
- <SCP specific problem detail> is the problem details.
- <OSCP-scp error Id> is the error ID with respect to SCP. It follows this format:

OSCP-<Service code>-<Category>-<3 digit error code>

.

- <Service code> is the microservice code that generates the error.
- <Category> is the error type for a particular microservice.
- <3 digit error code> is the error code prefixed by the alphabet E.
For more information about supported error IDs, see [SCP Error IDs](#).

Example of a sample error response:

```
ProblemDetails
{
  "title": "Access Token Denied",
  "status": 403,
  "detail": "scp1.e.com: OAuth token denied for Server1-FQDN: OSCP-OAU-
TKNRSP-E001",
  "cause": ACCESS_TOKEN_DENIED
}
```

SCP combines all the error responses from both internal and external sources, such as error responses generated by SCP and error responses received from producer NFs while performing alternate routing. SCP uses the `enhanced_error_rsp` attribute to combine all the error responses from both internal and external sources. For more information about `enhanced_error_rsp`, see [Configuring Error Response Enhancement](#).

Note

- When the `enhanced_error_rsp` parameter is disabled, `ProblemDetails` includes information about the last routing attempt made by SCP.
- When the `enhanced_error_rsp` parameter is enabled, `ProblemDetails` includes information about all the routing attempts made by SCP.

When SCP receives a message request from a consumer NF, SCP routes the request to the target producer NF. If the target producer NF is unavailable, SCP performs alternate routing to the next available producer NF. If SCP does not receive a success response from any producer NF after performing all the routing attempts, SCP generates an error response by updating the `detail` parameter with all the routing attempts made in the following format:

Latest attempt error string ; Status Code of previous attempt; Previous attempt error string

Sample detail format:

```
"detail" : "<Server-NF FQDN>: <NF Service>: <Application specific problem
description>: <VendorNF-error-ID>; StatusCode; <SCP FQDN>: <Problem
Description>: <OSCP-scp error Id>"
```

Where,

- `<Server-NF FQDN>` is the FQDN of the NF that sends this error response. For example, `NRF-d5g.oracle.com`.
- `<NF Service>` is the NF service name applicable for 3GPP service operation. For example, `nnrf-disc`.
- `<Application specific problem description>` is a short description of the error. For example, `requesterNfType` should be present.
- `<VendorNF-error-ID>` is the error code generated by any vendor NF. For example, `0NRF-DIS-DISC-E0006`.

- `StatusCode` is the HTTP status code for the previous routing attempt made by SCP. For example, 403.
- `<SCP FQDN>` is the FQDN of SCP. For example, `ocscp-scp-worker.scpsvc.svc.cluster.local`.
- `<Problem Description>` is the problem details. For example, Request cannot be routed due to rejection of access token request by NRF.
- `<OSCP-scp error Id>` is the ID of SCP. For example, `OSCP-0AU-TKNRSP-E001`.

The following sample `ProblemDetails` is based on the scenario when SCP has made two retry attempts to two different Producer NFs and both resulted in error responses (first attempt: 503 and second attempt: 400) returned by producer NFs. SCP returns the last error response (400) to the consumer NF:

```
ProblemDetails
{
  "title": "Bad Request",
  "status": 400,
  "detail": "Server2-FQDN: [NF Service]:Application specific problem detail: VendorNF-App-Error ID ; 503; Server1-FQDN: Application specific problem detail: VendorNF-App-Error ID",
  "cause": INVALID_API
}
```

The title, status, and cause parameters in `ProblemDetails` provide information about the last routing attempt made. The detail parameter provides information about all the routing attempts in the reverse order.

Note

The error response sent by producer NFs is forwarded as it is to the consumer NF.

The detail parameter in `ProblemDetails` has a length restriction managed by the **Error Detail Max Size** field as described in [Configuring Error Response Enhancement](#). The extra length leads to truncation of data in the detail parameter from the end.

The following sample `ProblemDetails` is based on the scenario when a timeout has occurred at SCP during inter SCP communication:

```
{
  "title":"Timeout at SCP","status":504,
  "detail":"ocscp-scp-worker.scpsvc.svc.cluster.local:Service Unavailable for pcf6svc.scpsvc.svc.cluster.local via scp61svc.scpsvc.svc.cluster.local:OSCP-WRK-ROUTE-E001;504;ocscp-scp-worker.scpsvc.svc.cluster.local:Service Unavailable for pcf6svc.scpsvc.svc.cluster.local via scp51svcxyz.scpsvc.svc.cluster.local due to UnknownHostException:OSCP-WRK-ROUTE-E002",
  "cause":"TIMED_OUT_REQUEST"}
```

The cause parameter of `ProblemDetails` provides a reason for the occurrence of the error, such as overload, access token denied, timeout, and so on. The cause parameter in `ProblemDetails` is managed as defined in Section 5.2.7 of 3GPP TS 29.500 v16.6.0. If this

section does not mention the cause value for any specific error condition, SCP leaves the value of this parameter empty.

The following sample ProblemDetails does not specify the cause of the error:

```
ProblemDetails
{
  "title": "Unauthorized",
  "status": 401,
  "detail": "SCP-FQDN: Application specific problem detail: SCP-App-Error
ID",
  "cause": <empty>
}
```

Managing Error Response Enhancement

Enable

This feature is enabled automatically at the time of SCP deployment. You can enable or disable this feature using the CNC Console as described in [Configuring Error Response Enhancement](#).

Configure

You can configure this feature using REST APIs, CNC Console, and Helm.

- **Configure using CNC Console:** For information about CNC Console configurations, see [Configuring Error Response Enhancement](#).
- **Configure using REST APIs:** For information about configuring REST APIs, see "Configuring SCP Features" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
- **Configure using Helm:** For information about configuring the SCP-Mediation microservice, see "SCP-Mediation Parameters" in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Observe

The `ocscp_worker_problemdetails_details_maxlengthexceeded_total` metric is added in the [Metrics](#) section.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.2.1 SCP Error IDs

The following table describes error IDs supported by SCP while generating error responses:

Table 3-1 SCP-Worker Microservice Error IDs

Error Details	Description	Possible Causes and Corrective Actions
Issue: Header Validation Error ID: OSCP-WRK-HDRVLD-E001 HTTP Response Code: The HTTP status code is populated based on the configured value in the ErrorProfileConfiguration REST API OR 403	This ID indicates that the received CCA header failed validation.	The Cause field is set based on the configured value in the ErrorProfileConfiguration REST API (such as, <code>errorProfile.getCause()</code>). OR CCA_VERIFICATION_FAILURE Corrective Actions The Consumer NF should send a valid CCA header.
Issue: Header Validation Error ID: OSCP-WRK-HDRVLD-E002 HTTP Response Code: 400	This ID indicates that the received SBI request does not contain all mandatory discovery headers required for Delegated NF Discovery.	Mandatory discovery headers were not received, resulting in missing required headers. MANDATORY_QUERY_PARAM_MISSING Corrective Actions The Consumer NF must send all mandatory discovery headers in Model-D SBI messages.
Issue: Header Validation Error ID: OSCP-WRK-HDRVLD-E003 HTTP Response Code: 400	This ID indicates that the SBI request received has a malformed 3gpp-sbi-routing-binding header.	A malformed Routing Binding header was received: <malformed-hdr>. MANDATORY_IE_INCORRECT Corrective Actions The Consumer NF should send a valid 3gpp-sbi-routing-binding header.
Issue: Header Validation Error ID: OSCP-WRK-HDRVLD-E004 HTTP Response Code: 400	This ID indicates that the received SBI request has imperfectly formed target PLMN.	A malformed Target PLMN was received. MANDATORY_IE_INCORRECT Corrective Actions The Consumer NF should send a valid value for the target PLMN in the Inter-PLMN SBI message.
Issue: Header Validation Error ID: OSCP-WRK-HDRVLD-E005 HTTP Response Code: 400	This ID indicates that the received SBI request has a malformed default notification request.	A malformed default notification request was received. MANDATORY_IE_INCORRECT Corrective Actions Consumer NF should send a valid header and URI for the default notification request.
Issue: Header Validation Error ID: OSCP-WRK-HDRVLD-E006 HTTP Response Code: 400	This ID indicates that the message could not be routed due to the absence of expected headers required to select a producer for routing.	Insufficient routing details were received in the request. MANDATORY_IE_MISSING Corrective Actions Consumer NF should send the appropriate headers for the 5G SBI message to be routed. For a Model-C request, either the 3gpp-sbi-target-apiroot or 3gpp-sbi-routing-binding header should be present. For a Model-D request, the 3gpp-sbi-discovery-* headers should be present.
Issue: Header Validation Error ID: OSCP-WRK-HDRVLD-E007 HTTP Response Code: 400	This ID indicates that the message could not be routed due to an invalid host value received in the 3gpp-sbi-target-apiroot header.	An invalid host value was received in the 3gpp-sbi-target-apiroot header. MANDATORY_IE_INCORRECT Corrective Actions Consumer NF should send a valid value for the host in the 3gpp-sbi-target-apiroot header.

Table 3-1 (Cont.) SCP-Worker Microservice Error IDs

Error Details	Description	Possible Causes and Corrective Actions
Issue: NF Selection Error ID: OSCP-WRK-NFSEL-E001 HTTP Response Code: 404	This ID indicates that the message could not be routed due to the absence of a learnt producer for the received 3gpp-sbi-routing-binding header and the absence of the 3gpp-sbi-target-apiroot header.	No routing rule was found for the received SBI request. Corrective Actions The Consumer NF should either send the 3gpp-sbi-target-apiroot header or a different binding value for the 3gpp-sbi-routing-binding header to route the SBI message.
Issue: NF Selection Error ID: OSCP-WRK-NFSEL-E002 HTTP Response Code: 400	This ID indicates that the message could not be routed due to the presence of an API version in the incoming SBI request that is not supported by the NF producer, as per the learned NF profile available at the SCP.	NF selection failed due to an API version mismatch. INVALID_API Corrective Actions The Consumer NF should send the correct API version in the SBI message URI that is supported by the Producer NF providing the NF service.
Issue: NF Selection Error ID: OSCP-WRK-NFSEL-E003 HTTP Response Code: 400	This error ID indicates that the message could not be routed due to unavailability of producer NFs with valid mcc-mnc host (found after Delegated NF Discovery). Therefore, it could not fetch SEPP details.	Next hop SEPP details were not found after NF discovery for <targetPlmn>. UNSPECIFIED_MSG_FAILURE Corrective Actions The Consumer NF should send a different value for the 3gpp-sbi-discovery-* headers to enable Inter-PLMN routing.
Issue: NF Selection Error ID: OSCP-WRK-NFSEL-E004 HttpStatusCode is populated based on the configured value for ocscp.http.routing_failure.no_host in RoutingOptions.	This ID indicates that the message could not be routed due to unavailability of serving SEPP for the inter-plmn. This is applicable for both Model-C and Model-D request routing models.	The Detail field is set based on the configured value in RoutingOptions against ocscp.http.routing_failure.no_host (for example, eer.getError_response()) Corrective Actions The Consumer NF should send a different value for the 3gpp-sbi-target-apiroot header to enable Inter-PLMN routing.
Issue: NF Selection Error ID: OSCP-WRK-NFSEL-E005 HttpStatusCode is populated based on the configured value in the ErrorProfileConfiguration REST API	This ID indicates that the message could not be routed because the selected producer is congested and there is no other eligible producer available for routing.	The Cause field is set based on the configured value in the ErrorProfileConfiguration REST API (for example, NFSvcLoadBasedErrorProfileConfiguration.getErrCause()) Corrective Actions The Consumer NF should either send a different value for the headers to enable routing, or resend the request with the same header value once the load on the producer NF returns to a normal level.
Issue: NF Selection Error ID: OSCP-WRK-NFSEL-E006 HttpStatusCode is populated based on the configured value for ocscp.http.routing_failure.resource_exhausted in RoutingOptions, or it defaults to 500.	This ID indicates that the message could not be routed because eligible producers were proactively filtered out due to circuit breaking, outlier detection, egress rate limiting, and so on.	The Detail field is populated based on the configured value for ocscp.http.routing_failure.resource_exhausted in RoutingOptions or Destination unreachable due to exhausted destination search. Corrective Actions Consumer NF should either send a different value for the headers to enable routing, or resend the request with the same header value once the producer NF becomes healthy.

Table 3-1 (Cont.) SCP-Worker Microservice Error IDs

Error Details	Description	Possible Causes and Corrective Actions
Issue: OCI Error ID: OSCP-WRK-OCI-E001 The HttpStatusCode is populated based on the configured value in the ErrorProfile Configuration, if available, or, from the value of <code>ocscp.http.routing_failure.resource_exhausted</code> in the RoutingOptions.	This ID indicates that the message could not be routed because the eligible producer was filtered out for OCI.	The Detail field is populated based on the configured value in the ErrorProfile Configuration, if available. Alternatively, it is populated using the value for <code>ocscp.http.routing_failure.resource_exhausted</code> in RoutingOptions. Corrective Actions The consumer NF should either send different header values to facilitate routing or resend the request with the same header values once the OCI reduction criteria are removed from the producer NF.
Issue: SCP-W Mediation Interface Errors Error ID: OSCP-WRK-MEDITN-E001 The HttpStatusCode is populated based on the value configured in the ErrorProfile, or 500.	This ID indicates that an error response was received by the SCP-Worker client from the Mediation service.	The Cause field is populated based on the value configured in AppRoutingOptionsConfig (for example, <code>errorProfileConfig.getErrorCause()</code>), or <code>continue_processing</code> . or Mediation client errors. Corrective Actions Ensure that the Mediation service is running and accessible.
Issue: SCP-W Mediation Interface Errors Error ID: OSCP-WRK-MEDITN-E005 HTTP Response Code: 500	This ID indicates that a decode failure occurred when processing the response received from the mediation.	Failure to decode the mediation message. or UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the appropriate rules are correctly configured in the Mediation.
Issue: Oauth Request Error ID: OSCP-WRK-OAUTH-E001 HTTP Response Code: 400	This ID indicates that the message could not be routed because the discovery headers, required to obtain the OAuth2 access token, are missing from the received request.	The request cannot be routed due to missing required OAuth2 access token information in the received request. MISSING_ACCESS_TOKEN_INFO Corrective Actions The consumer NF must include all mandatory headers required for the OAuth token request.
Issue: Oauth Request Error ID: OSCP-WRK-OAUTH-E002 HTTP Response Code: 400	This ID indicates that the request could not be routed because OAuth2 access token was not received from consumer NF.	The request cannot be routed because the OAuth2 access token was not received from the consumer NF. ACCESS_TOKEN_NOT_RECEIVED Corrective Actions The consumer NF must include the OAuth token when sending an SBI message.
Issue: Oauth Request Error ID: OSCP-WRK-OAUTH-E003 HTTP Response Code: 400	This ID indicates that the message could not be routed due to an invalid OAuth2 access token.	The request cannot be routed because the OAuth2 access token is invalid. INVALID_ACCESS_TOKEN Corrective Actions Ensure that the mandatory headers required to obtain the access token from the NRF are valid.
Issue: Oauth Request Error ID: OSCP-WRK-OAUTH-E004 HTTP Response Code: 403	This ID indicates that the message could not be routed due to OAuth2 access token being denied.	The request cannot be routed because the OAuth2 access token has been denied. ACCESS_TOKEN_DENIED Corrective Actions The consumer NF should send the SBI message with updated header values in order to request the access token.

Table 3-1 (Cont.) SCP-Worker Microservice Error IDs

Error Details	Description	Possible Causes and Corrective Actions
Issue: OAuth Request Error ID: OSCP-WRK-OAUTH-E005 HTTP Response Code: 500	This ID indicates that the message could not be routed because the access token request from the SCP-worker to the SCP-NrfProxy-Oauth service timed out.	The access token request timed out from SCP-worker to the SCP-NrfProxy-OAuth service. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the scp-nrfproxy-oauth service responds within the specified timeout limit.
Issue: OAuth Request Error ID: OSCP-WRK-OAUTH-E006 HTTP Response Code: 500	This ID indicates that the message could not be routed because the access token request from the SCP-worker to the SCP-NrfProxy-Oauth service encountered a transport error.	The access token request could not be processed due to a transport error between SCP-worker and the SCP-NrfProxy-OAuth service. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the scp-nrfproxy-oauth service is running and accessible.
Issue: OAuth Request Error ID: OSCP-WRK-OAUTH-E008 HTTP Response Code: 500	This ID indicates that the message could not be routed because the access token request from SCP-worker to SCP-NrfProxy-Oauth encountered an exception other than TimeoutException, ConnectException, or UnknownHostException.	The access token request could not be processed due to unknown errors when connecting SCP-worker to the SCP-NrfProxy-OAuth service. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the scp-nrfproxy-oauth service is running and accessible.
Issue: OAuth Request Error ID: OSCP-WRK-OAUTH-E009 HTTP Response Code: 500	This ID indicates that the message could not be routed because the access token request from SCP-Worker to SCP-NrfProxy-Oauth failed due to decoding exception in the access token error response body to the ProblemDetails instance.	A failure in decoding the response body into a ProblemDetails instance for the access token request. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the scp-nrfproxy-oauth service returns an error response with the ProblemDetails information element (IE) in the response body.
Issue: Ingress Rate Limiting Error ID: OSCP-WRK-IRTLMT-E001 The HttpStatusCode is populated based on the configured value in the IngressRateLimiter REST API.	This ID indicates that the message could not be routed because the ingress rate was more than the allowed rate configured for the consumer as specified by the rules configured in the Ingress Rate Limiter REST API.	The Detail field is populated based on the configured value in the IngressRateLimiter REST API (for example, <code>configData.getErrorResponse().getErrorResponse().NF_CONGESTION_RISK</code>). Corrective Actions To address the issue, either reduce the ingress rate or increase the allowed rate in the Ingress Rate Limiter REST API configuration.
Issue: Egress Rate Limiting Error ID: OSCP-WRK-ERTLMT-E001 The HttpStatusCode is populated based on the configured value in the EgressRateLimiter REST API.	This ID indicates that the message could not be routed because the ingress rate was more than the allowed rate configured for the consumer as specified by the rules configured in the Egress Rate Limiter REST API.	The Detail field is populated based on the configured value in the EgressRateLimiter REST API (for example, <code>streamInfo.getEgressRateLimitconfigData().getErrorResponse().getErrorResponse()</code>). NF_CONGESTION_RISK Corrective Actions To address the issue, either reduce the ingress rate or increase the allowed rate in the Egress Rate Limiter REST API configuration.

Table 3-1 (Cont.) SCP-Worker Microservice Error IDs

Error Details	Description	Possible Causes and Corrective Actions
Issue: Routing Error ID: OSCP-WRK-ROUTE-E001 The HttpStatusCode is populated based on the value configured in RoutingOptions.	This ID indicates that the message could not be routed because a TimeoutException was encountered at the Worker Client while reaching the upstream producer.	The Detail field is populated based on the configured value in RoutingOptions, along with the attempted NF FQDN or IP. TIMED_OUT_REQUEST Corrective Actions Ensure that the producer NF responds within the timeout limit at SCP, or increase the response timeout limit at SCP.
Issue: Routing Error ID: OSCP-WRK-ROUTE-E002 The HttpStatusCode is populated based on the value configured in RoutingOptions.	This ID indicates that the message could not be routed because WebClientRequestException was encountered at the Worker Client while reaching upstream the producer NF. The WebClient exception could be a timeout, a connection failure, or any other error from WebClient.	The Detail field is populated based on the configured value in RoutingOptions, including the attempted NF FQDN or IP address. TARGET_NF_NOT_REACHABLE Corrective Actions Ensure that the producer NF is running and accessible.
Issue: Routing Error ID: OSCP-WRK-ROUTE-E003 The HttpStatusCode is populated based on the value configured in RoutingOptions.	This ID indicates that the message could not be routed due to an exception other than WebClientResponseException, WebClientRequestException, or TimeoutException encountered at the Worker Client while reaching upstream the producer NF.	The Detail field is populated using the configured value in RoutingOptions, along with the attempted NF FQDN or IP address. TARGET_NF_NOT_REACHABLE Corrective Actions Ensure that the producer NF is running and accessible.
Issue: Routing Error ID: OSCP-WRK-ROUTE-E004 HTTP Response Code: 508	This ID indicates that the message could not be routed due to the presence of the SCP's self FQDN in the incoming SBI request's Via header. It also suggests that required headers or information, such as target-apiroot, binding headers, discovery headers, or any other necessary routing information, are missing from the request header.	A message loop has been detected. Corrective Actions Ensure that the SBI message does not contain the self FQDN in the "Via" header.
Issue: Routing Error ID: OSCP-WRK-ROUTE-E005 HTTP Response Code: 307 or 308 (as received from upstream NF)	This ID indicates that the SBI message has been redirected	A 3xx redirection was received from the upstream producer at ` <code><producer fqdn or IP></code> `.

Table 3-1 (Cont.) SCP-Worker Microservice Error IDs

Error Details	Description	Possible Causes and Corrective Actions
Issue: Routing Error ID: OSCP-WRK-ROUTE-E006 The HttpStatusCode is populated based on the error response received from the platform service mesh.	This ID indicates that the message could not be routed due to an error response received from the platform service mesh.	An error response was received from the platform service mesh at <service mesh fqdn or IP>. TARGET_NF_NOT_REACHABLE Corrective Actions Ensure that the producer NF is online and accessible, and confirm that the platform's service mesh is configured properly.
Issue: Routing Error ID: OSCP-WRK-ROUTE-E007 The HttpStatusCode is set based on the value configured in the RoutingOptions.	This ID indicates that the message could not be routed because the total transaction timeout for the service was reached.	The Detail field is populated based on the configured value in RoutingOptions, along with the attempted NF FQDN or IP address. TIMED_OUT_REQUEST Corrective Actions Ensure that the total transaction timeout is configured correctly and ensure there are no exceptions in the worker logs.
Issue: Routing Error ID: OSCP-WRK-ROUTE-E008 HTTP Response Code: HttpStatusCode is populated based on 'Insufficient_Time' exceptionErrorResponse configured in relevant Routing Config Set	This ID indicates that the message could not be routed due to a request timeout at SCP based on the timeout value received in the 3GPP-sbi-max-rsp-time header.	The request timed out at SCP based on the 3GPP-sbi-max-rsp-time header. TIMED_OUT_REQUEST Corrective Actions • Ensure that the 3gpp-sbi-max-rsp-time value in the ingress SBI request is correct. • Verify that the configured responseTimeout value is accurate. • Confirm that SCP and NFs are properly time synchronized. • Check if the producer NF responds within the time specified in the 3gpp-sbi-max-rsp-time header of the egress request.
Issue: Routing Error ID: OSCP-WRK-ROUTE-E009 HTTP Response Code: 500	This error indicates that SCP has received a nonstandard HTTP status code from a producer NF.	Unknown Status Code UNKNOWN_STATUS_CODE Corrective Actions No corrective actions available.
Issue: Delegated Discovery Error ID: OSCP-WRK-DELDISC-E001 HTTP Response Code: 500	This ID indicates that the delegated discovery request from SCP-Worker to SCP-NrfProxy failed due to TimeoutException.	The delegated discovery request timed out from SCP-worker to the SCP-NrfProxy service. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure the scp-nrf-proxy service responds within the specified timeout limit.
Issue: Delegated Discovery Error ID: OSCP-WRK-DELDISC-E002 HTTP Response Code: 500	This ID indicates that the delegated discovery request from the SCP-worker to the SCP-NrfProxy service failed due to a WebClientRequestException.	Delegated discovery from SCP-worker to the SCP-NrfProxy service failed due to a transport error. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure the scp-nrf-proxy service is running and accessible.

Table 3-1 (Cont.) SCP-Worker Microservice Error IDs

Error Details	Description	Possible Causes and Corrective Actions
Issue: Delegated Discovery Error ID: OSCP-WRK-DELDISC-E004 HTTP Response Code: 500	This ID indicates that the delegated discovery request from SCP-Worker to SCP-NrfProxy failed due to an exception other than TimeoutException, ConnectException, or UnknownHostException.	Delegated discovery from SCP-worker to the SCP-NrfProxy service failed due to an unknown error. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure the scp-nrf-proxy service is operational and reachable.
Issue: Delegated Discovery Error ID: OSCP-WRK-DELDISC-E005 HTTP Response Code: 500	This ID indicates that the delegated discovery request from SCP-Worker to SCP-NrfProxy failed due to decoding exception in the delegated NF discovery error response body to the ProblemDetails instance.	Failure in decoding the response body into a ProblemDetails instance for delegated discovery. UNSPECIFIED_NF_FAILURE Corrective Actions Ensures that the scp-nrf-proxy service returns an error response with a problemdetails information element in the response body.
Issue: Pod Overload Error ID: OSCP-WRK-OVERLD-E001 The HttpStatusCode is populated based on the configured value in the ErrorProfile REST API.	This ID indicates that the SCP-worker pod is overloaded due to CPU overload or pending transactions.	The Cause field is populated based on the configured value in the ErrorProfile REST API, using the method <code>errorProfileConfiguration.getErrorCause()</code> . Corrective Actions The Consumer NF should send an SBI message to the SCP once it has recovered from the overload situation (either due to CPU usage or pending transactions). This recovery can be verified by monitoring the relevant SCP alerts.
Issue: Internal Server Errors Error ID: OSCP-WRK-INTRNL-E001 HTTP Response Code: 500	This ID indicates that the message could not be routed due to downstream task queue being full.	The downstream task queue is full. UNSPECIFIED_NF_FAILURE Corrective Actions The Consumer NF may send the request after some time, once the SCP is able to process the message.
Issue: Internal Server Errors Error ID: OSCP-WRK-INTRNL-E002 HTTP Response Code: 500	This ID indicates that message could not be routed due to an exception caused failure of downstream asynchronous call.	The downstream asynchronous call failed. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the request contains valid data to prevent any exceptions from occurring.
Issue: Internal Server Errors Error ID: OSCP-WRK-INTRNL-E003 HTTP Response Code: 500	This ID indicates that the message could not be routed due to an exception encountered on the request processing path.	An uncaught exception occurred on the request path, resulting in an internal server error. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the request contains valid data to prevent any exceptions from occurring.
Issue: Internal Server Errors Error ID: OSCP-WRK-INTRNL-E004 HTTP Response Code: 500	This ID indicates that the message could not be routed due to an exception encountered on the response processing path.	An uncaught exception occurred on the response path, leading to an internal server error. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the response contains valid data to prevent any exceptions from occurring.
Issue: Internal Server Errors Error ID: OSCP-WRK-INTRNL-E005 HTTP Response Code: 500	This ID indicates that the message could not be routed due to an exception encountered during SBI message processing.	Unable to process the message due to an unknown error. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that both the request and response contain valid data to prevent any exceptions from occurring.

Table 3-1 (Cont.) SCP-Worker Microservice Error IDs

Error Details	Description	Possible Causes and Corrective Actions
Issue: Internal Server Errors Error ID: OSCP-WRK-INTRNL-E006 HTTP Response Code: The status code from the producer.	This ID indicates that error rsp(4xx or 5xx) is received from upstream, and the content-type is problem-json but the response body is not in problemdetails format.	Failure in decoding the response body into a ProblemDetails instance. Corrective Actions Ensure that the producer NF sends a valid ProblemDetails information element in the error response body when the Content-Type is application/problem+json.
Issue: Internal Server Errors Error ID: OSCP-WRK-INTRNL-E007 HTTP Response Code: The status code from the producer.	This ID indicates that error rsp(4xx or 5xx) is received from upstream, but the content-type is not problem-json, application/json, or any other allowed format.	The Content-Type of the error response is not application/problem+json. Corrective Actions Ensure that the producer NF sends a valid ProblemDetails information element in the error response body when the Content-Type is application/problem+json.
Issue: Internal Server Errors Error ID: OSCP-WRK-INTRNL-E008 HTTP Response Code: 500	This ID indicates that Live Object pool is exhausted.	The live object pool is exhausted. Corrective Actions Ensure the object pool can adjust to instance creation and reclamation requests, or increase the pool size as needed.

Table 3-2 SCP OAUTH MICROSERVICE

Error Details	Description	Possible Causes and Corrective Actions
Issue: Access Token Request Error ID: OSCP-OAU-TKNREQ-E001 HTTP Response Code: 500	This ID indicates that NrfProxy-OAuth timed out while attempting to connect to the Worker to initiate the OAuth request towards NRF.	The access token request timed out at the NrfProxy-OAuth service. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the worker sends a response within the response timeout limit at the NrfProxy-OAuth service.
Issue: Access Token Request Error ID: OSCP-OAU-TKNREQ-E002 HTTP Response Code: 500	This ID indicates that NrfProxy-OAuth failed to connect to the Worker to initiate the OAuth request towards NRF due to a WebClientRequestException.	The access token request failed due to a transport error at the NrfProxy-OAuth service. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the worker is up and reachable for successful communication.
Issue: Access Token Request Error ID: OSCP-OAU-TKNREQ-E004 HTTP Response Code: 500	This ID indicates that NrfProxy-OAuth failed to connect to the Worker to initiate the OAuth request towards NRF due to an exception other than TimeoutException, ConnectException, or UnknownHostException.	The access token request failed due to an unknown error at the NrfProxy-OAuth service. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the worker is up and reachable.

Table 3-2 (Cont.) SCP OAUTH MICROSERVICE

Error Details	Description	Possible Causes and Corrective Actions
Issue: Access Token Response Error ID: OSCP-OAU-TKNRSP-E001 HTTP Response Code: 403	This ID indicates that the NrfProxy-OAuth service received a 400 Bad Request with an AccessTokenError payload from NRF for the access token request, and SCP converts it to a 403 with a ProblemDetails information element.	The request cannot be routed due to the rejection of the access token request by the NRF. ACCESS_TOKEN_DENIED Corrective Actions Ensure that the Consumer NF sends the access token request with valid headers and parameters.
Issue: Access Token Response Error ID: OSCP-OAU-TKNRSP-E002 HTTP Response Code: 403	This ID indicates that the NrfProxy-OAuth service received a 200 response from NRF with an access token payload that is expired, and SCP converts it to a 403 with a ProblemDetails information element.	An expired access token was received from the NRF, with an expiration time of <exp-time>. ACCESS_TOKEN_DENIED Corrective Actions Ensure that valid request and response data is received by the NrfProxy-OAuth service for proper processing.
Issue: Access Token Response Error ID: OSCP-OAU-TKNRSP-E003 HTTP Response Code: 504	This ID indicates that the NrfProxy-OAuth service received a 307 response from NRF via the worker. SCP converts it to a 504, NRF_NOT_REACHABLE with `ProblemDetails` information element.	The SCP could not obtain access from the NRF. The access token request failed because NRFs were not reachable. NRF_NOT_REACHABLE Corrective Actions Ensure that the NRFs configured for the access token procedure are reachable.
Issue: Internal Server Errors Error ID: OSCP-OAU-INTRNL-E001 HTTP Response Code: 500	This ID indicates that an exception was encountered while processing the access token request.	An uncaught exception occurred on the server, leading to an internal server error. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that valid request and response data is received by the NrfProxy-OAuth service for correct processing.
Issue: Internal Server Errors Error ID: OSCP-OAU-INTRNL-E002 HTTP Response Code: 500	This ID indicates that a downstream thread in the NrfProxy-OAuth service is stuck while processing the access token request.	The downstream thread for the Nrf-Proxy-OAuth service experienced a timeout. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the NrfProxy-OAuth service receives valid request and response data for proper processing.
Issue: Internal Server Errors Error ID: OSCP-OAU-INTRNL-E003 HTTP Response Code: 500	This ID indicates that an exception was encountered while processing the access token request or response.	The message could not be processed due to an unknown error. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the NrfProxy-OAuth service receives valid request and response data to ensure correct processing and avoid errors.

Table 3-3 SCP NRF-PROXY MICROSERVICE

Error Details	Description	Possible Causes and Corrective Actions
Issue: Delegated Discovery Request Error ID: OSCP-NPY-DISREQ-E001 HTTP Response Code: 500	This ID indicates that the NrfProxy timed out while attempting to connect to the Worker to initiate the delegated discovery request towards NRF.	The delegated NF discovery request timed out at the NrfProxy service. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure the worker sends a response within the specified response timeout limit.
Issue: Delegated Discovery Request Error ID: OSCP-NPY-DISREQ-E002 HTTP Response Code: 500	This ID indicates that the NrfProxy failed to connect to the Worker to initiate the delegated discovery request towards NRF due to a WebClientRequestException.	The delegated NF discovery request failed due to a transport error at the NrfProxy service. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the worker is up and reachable.
Issue: Delegated Discovery Request Error ID: OSCP-NPY-DISREQ-E004 HTTP Response Code: 500	This ID indicates that the NrfProxy failed to connect to the Worker to initiate the delegated discovery request towards NRF due to an exception other than TimeoutException, ConnectException, or UnknownHostException.	The delegated NF discovery request failed due to an unknown error at the NrfProxy service. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the worker is up and reachable.
Issue: Delegated Discovery Request Error ID: OSCP-NPY-DISREQ-E005 HTTP Response Code: 500	This ID indicates that the NrfProxy could not find the cache key, and as a result, was unable to process the delegated discovery request.	The cache key was not received from the SCP-Worker. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that the cache key is generated at the scp-worker and sent to the scp-nrfproxy service.
Issue: Delegated Discovery Request Error ID: OSCP-NPY-DISREQ-E006 HTTP Response Code: 500	This ID indicates that the NrfProxy could not find the cache for the delegated discovery response due to an exception encountered during JSON parsing.	An exception occurred while parsing the delegated NF discovery cached response into a JSON array. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that valid response data is cached in the scp-nrfproxy service.
Issue: Delegated Discovery Response Error ID: OSCP-NPY-DISRSP-E001 HTTP Response Code: 400	This ID indicates that an empty list of NF profiles was received in the delegated NF discovery response.	An empty list of NF profiles was received in the delegated NF discovery response. NF_DISCOVERY_FAILURE Corrective Actions The Consumer NF should send different values for the 3gpp-sbi-discovery-* headers to ensure that the delegated discovery returns a non-empty response.
Issue: Delegated Discovery Response Error ID: OSCP-NPY-DISRSP-E002 HTTP Response Code: 500	This ID indicates that no response was received for the delegated NF discovery.	No response was received for the delegated NF discovery. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that valid response data is received by the scp-nrfproxy service.
Error ID: OSCP-NPY-DISRSP-E003 HTTP Response Code: 500	This ID indicates that the NrfProxy failed to build the delegated NF discovery response.	Failed to build the delegated NF discovery response. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that valid response data is received by the scp-nrfproxy service for processing.

Table 3-3 (Cont.) SCP NRF-PROXY MICROSERVICE

Error Details	Description	Possible Causes and Corrective Actions
Issue: Delegated Discovery Response Error ID: OSCP-NPY-DISRSP-E004 HTTP Response Code: 500	This ID indicates that validation failed for the NF profiles received in the delegated NF discovery response.	Validation of the NF profiles received in the delegated NF discovery response failed. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that a response with valid NF profiles data is received by the scp-nrfproxy service.
Issue: Delegated Discovery Response Error ID: OSCP-NPY-DISRSP-E005 HTTP Response Code: 500	This ID indicates that there was a failure in deserializing the delegated NF discovery response.	Failed to deserialize the delegated NF discovery response. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that valid response data is properly cached in the scp-nrfproxy service.
Issue: Internal Server Errors Error ID: OSCP-NPY-INTRNL-E001 HTTP Response Code: 500	This ID indicates that the asynchronous call failed in the NrfProxy service for the delegated NF discovery request.	The asynchronous call for the delegated NF discovery request failed. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that valid response data is successfully received by the scp-nrfproxy service for processing.
Issue: Internal Server Errors Error ID: OSCP-NPY-INTRNL-E002 HTTP Response Code: 500	This ID indicates that the NrfProxy service failed to process the delegated NF discovery request due to an exception.	The NrfProxy service failed to process the delegated NF discovery request. UNSPECIFIED_NF_FAILURE Corrective Actions Ensure that valid response data is received by the scp-nrfproxy service and processed correctly.

Table 3-4 SCP NOTIFICATION MICROSERVICE

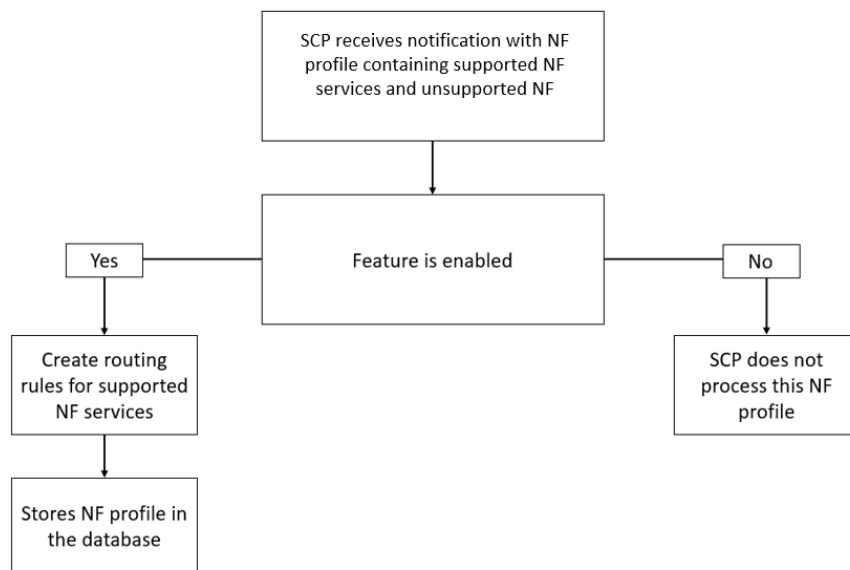
Error Details	Description	Possible Causes and Corrective Actions
Issue: Profile Validations Error ID: OSCP-NTF-PRFVLD-E001 HTTP Response Code: 400	This ID indicates that NF profile received from NRF is invalid as it contains both nfProfile and profileChange elements for NF_PROFILE_CHANGED event.	Problem Cause: The NF profile received from the NRF is invalid because it contains both nfProfile and profileChange for the NF_PROFILE_CHANGED event. INVALID_REQUEST_BODY Corrective Actions NRF should send only the profileChange attribute for the NF_PROFILE_CHANGED event.
Issue: Profile Validations Error ID: OSCP-NTF-PRFVLD-E002 HTTP Response Code: 400	This ID indicates that NF profile received from NRF is missing some mandatory parameters amongst event, nfType, nfStatus, and nfInstanceId.	The NF profile received from the NRF is missing some mandatory parameters, including event, nfType, nfStatus, and nfInstanceId. MANDATORY_IE_MISSING Corrective Actions NF profiles sent by NRF must include all mandatory fields.
Issue: Profile Validations Error ID: OSCP-NTF-PRFVLD-E003 HTTP Response Code: 400	This ID indicates that the NF profile received from NRF has an invalid JSON format.	The NF profile received from the NRF has an invalid JSON format. INVALID_MSG_FORMAT Corrective Actions NF profiles sent by NRF must contain valid JSON content.

3.3 NFProfile Processing Enhancements

In the previous releases, SCP did not support custom NF services. When SCP receives a notification from NRF about an NF profile that has 3GPP-defined supported NF services along with custom, unknown, or unsupported NF services, it will reject the whole NF profile. Therefore, routing rules were not generated for 3GPP-defined supported NF services when a custom, unknown, or unsupported NF service was present in the NF profile.

With this enhancement, whenever SCP receives a notification regarding an NF profile that includes 3GPP-defined supported NF services, custom, unknown, or unsupported NF services, SCP ignores the presence of custom, unknown, or unsupported NF services within the NF profile. Routing rules are generated by SCP for 3GPP-defined SCP supported NF services that are present in the NF profile. Consequently, SCP enhanced the process of handling NF profiles by streamlining the routing process and ensuring efficient and reliable communication between NFs.

Figure 3-1 Call Flow of NFProfile Processing



1. SCP receives a notification with an NF profile including 3GPP-defined supported NF services, custom, unknown, or unsupported NF services.
 2. If the feature is enabled, then SCP creates routing rules for the 3GPP-defined supported NF services and ignores the custom, unknown, or unsupported NF services present in the NF profile.
 3. SCP stores the NF profile in the database.
 4. If the feature is disabled, then SCP does not process that NF profile and no routing rules will be created for that NF profile.
- This call flow ensures that SCP appropriately handles notifications and routing rules procedures with supported NF services while disregarding unsupported services.

Managing NFProfile Processing Enhancements

Enable

You can enable this feature by setting the **ignore_unknown_nfservice** parameter to true. For more information, see [Configuring Enhanced NFProfile Processing](#)

Configure

You can enable this feature using the CNC Console or REST API.

- **Configure using CNC Console:** For more information about CNC Console configurations, see [Configuring Enhanced NFProfile Processing](#).
- **Configure using REST API:** For information about REST API configurations, see "Configuring SCP Features" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

- Added the `ocscp_ignore_unknown_service_total` metric in the [SCP Metrics](#) section.
- Added the [SCPIgnoreUnknownService](#) in the [Alerts](#) section.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.4 Support for cnDBTier APIs in CNC Console

With the implementation of this feature, cnDBTier APIs are integrated into the CNC Console, and SCP users can view specific cnDBTier APIs, such as checking the cnDBTier version, status of cnDBTier clusters, and georeplication status on the CNC Console.

The following cnDBTier APIs are read only and can be viewed in CNC Console:

- **Backup List:** This API displays the details of stored backups, such as the ID and size of the backup.
- **cnDBTier version:** This API displays the cnDBTier version.
- **Database Statistics Report:** This API displays the number of available database.
- **Georeplication Recovery:** This API provides the recovery status of georeplication for the cnDBTier cluster:
 - **Update Cluster As Failed:** This API is used to mark a disrupted cluster as failed.
 - **Start Georeplication Recovery:** This API is used to start the georeplication recovery process.
 - **Georeplication Recovery Status:** This API is used to monitor the recovery status of georeplication for both FAILED and ACTIVE cnDBTier sites.
- **Georeplication Status:**

- Real Time Overall Replication Status: This API displays the overall replication status in multisite deployments. For example, in a four-site deployment, it provides the replication status between the following sites: site1-site2, site1-site3, site1-site4, site2-site3, site2-site4, and site2-site1. This is applicable for all other sites.
 - Site Specific Real Time Replication Status: This API displays the site-specific replication status.
- HeartBeat Status: This API displays the connectivity status between the local site and the remote site name to which SCP is connected.
- Local Cluster Status: This API displays the status of the local cluster.
- On-Demand Backup: This API displays the status of initiated on-demand backups.
- cnDBTier Health: This API displays the health status of the following services:
 - Replication Health Status: This API displays the health status of the replication service. It checks the following:
 - * if the replication service is up or not
 - * if the replication service can connect to database or not
 - Monitor Health Status: This API displays the health status of the monitor service. It checks the following:
 - * if the monitor service is up or not
 - * if the service can connect to database or not
 - * if the metrics are fetched or not (the metrics are fetched when the service is up and vice versa)
 - NDB Health Status: This API displays the health status of the NDB service pods like (data pods, sql pods, app-my-sql pods, mgmt pods). It checks the following:
 - * if the pod is connected to PVC or not
 - * if the pods status is up or not

Note

PVC Health Status attribute is set to NA when some of the database pods are not connected to the PVC.

- Backup Manager Health Status: This API displays the health status of the backup manager service. It checks the following:
 - * if the backup manager service is up or not
 - * if the service can connect to database or not

Managing cnDBTier APIs at CNC Console

Enable

This feature is enabled automatically when cnDBTier is configured as an instance during the CNC Console deployment. For more information about integrating cnDBTier APIs in CNC Console, see *Oracle Communications Cloud Native Core, cnDBTier User Guide*.

Configure

You can view cnDBTier APIs at CNC Console in the [Viewing cnDBTier APIs in CNC Console](#) section.

Maintain

If you encounter alerts at the system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.5 NRF Configuration Using DNS SRV Resolution

Currently, SCP supports the static configuration (helm configuration) of NRF profiles from the `custom-values.yaml` file and uses them for registration, subscription, discovery, and audit.

With the introduction of this feature, SCP supports the configuration (RESTful) of NRF Service (SRV) FQDN at SCP. SCP uses this Domain Name System Service (DNS) SRV Resource record, known as Service Proto Name (SPN), to query the DNS server to locate the actual target NRF FQDN (DNS SRV NRF FQDN). Thus, the user can configure primary (mandatory), secondary, and tertiary NRFs based on their priority and weights from different regions in the DNS server against a NRF SRV FQDN.

During migration, SCP deregisters and unsubscribes from static NRFs to update the NRF details stored statically at SCP with DNS SRV NRFs. Hence, it creates new rules for DNS SRV NRFs and deletes old static NRF rules. SCP instructs Audit microservice to perform an on-demand audit. The user can configure the DNS SRV and update all the data in the database for subscription, registration, discovery, and audit services. Configuration of different NRF SRV FQDN's with duplicate `nfsetIds` is not allowed. SCP allows users to configure multiple `nfsetIDs`, provided they are unique, but SCP will support only the 0th `nfsetID`.

To use the same `NFInstance ID` from the static NRF config, the following conditions must be met:

- The NF set ID should match
- NF service counts should match, and NF service FQDNs should also match

Hence, if the LCI comes from NRF against a different `nfinstanceID`, SCP will ignore the received LCI header as unknown.

NRFs are configured on the DNS server or stub against the NRF SRV FQDN. These NRFs should have PLMN IDs in the FQDN value matching the serving PLMN IDs of SCP. Setting the `isInterPLMnfqdn` to true will enable InterPLMN request routing coming from a foreign PLMN to SUT SCP. SCP will be able to perform intra-PLMN routing to local NRFs based on the `interplmnFQDN`. If `isInterPLMnfqdn` is set to false, the above-mentioned routing will not be performed if the Inter-PLMN request for Intra-PLMN routing is received with only `3gpp-Sbi-Target-apiRoot` and no additional information, such as a routing binding header.

Note

SCP never performs InterPLMN routing through SEPP for NRF as a destination, which is learned through DNSSRV.

Note

To avoid the loss of service, it is recommended that the user configure the same static target FQDN and NFSetID in the DNS SRV configuration, which are configured for Support for Model D and Support for OAuth2.0 Feature.

If the same target NRF FQDN is detected against different NRF SRV FQDNs, the `ocscp_configuration_dnssrv_nrf_duplicate_target_detected` metric is pegged, where the value of `nrfSrvFqdn` indicates the NRF SRV FQDN against which a duplicate target NRF is detected.

Managing NRF Configuration using DNS SRV Resolution**Enable**

During deployment:

Set the `enableDnsBasedNrfBootStrapInfoFeature` to true and provide the NRF SRV Config under the `nrfSrvConfiguration` field of the deployment file to enable the DNS SRV NRF feature during fresh deployment.

Sample of changes in deployment file:

```
#-----NRF Bootstrapping based on DNS SRV-----#
#Enable/Disable nrf_bootstrap_info feature
  enableDnsBasedNrfBootStrapInfoFeature : false
  deRegisterScpDuringMigration: false

#Preferred DNSSRV NrfSetId to be used for on demand discovery when
nrf_bootstrap_info feature is enabled during deployment
  preferredDNSSRVNrfSetIdForOnDemandDiscovery:
["setnrf11.nrfset.5gc.mnc012.mcc345"]

#Configure NRF Configuration using DNS SRV Resolution when the
nrf_bootstrap_info feature is enabled.
nrfSrvConfiguration:
- nrfSrvFqdn: nrf1svc.scpsvc.svc.cluster.local
  data:
    "v1":
      nfSetIdList:
      - "setnrf11.nrfset.5gc.mnc012.mcc345"
      #plmnList:
      # - mcc: 330
      #   mnc: 143
      performSubscription: true
      performAudit: true
      registerScp: true
      scheme: http
      #apiPrefix: USEast
      versions:
      - apiVersionInUri: v1
        apiFullVersion: 1.0.0
      serviceNames:
      - nnrf-nfm
      - nnrf-disc
      - nnrf-oauth2
      isInterPlmnFqdn: false
```

For more information, refer *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

By default, this feature is disabled. You can use the `nrf_bootstrap_info` option to enable or disable NRF configuration using the DNS SRV resolution feature and its configuration as described in the [Step 15](#) of the [Configuring SCP Features](#) section.

Note

DNS SRV alternate routing feature should be enabled to enable NRF Configuration using DNS SRV Resolution feature.

Configure

You can configure this feature using REST API and the CNC Console.

- **Configure using CNC Console:** For information about configuring this feature, see the [Configuring NRF SRV](#) and [Mapping NFIInstanceId and ServiceInstanceId](#) sections. To view the runtime status of NRF migration task, see the [Viewing SCP Feature Status](#) Status section.
- **Configure using REST API:**
You can configure this feature using REST APIs as described in "Configuring SCP Features", "SCP NRF SRV Configuration", and "SCP NRF FQDN InstanceId Mapping". You can view the runtime status of NRF migration task in the "Viewing SCP Feature Status" section in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

The following metrics are added in the [SCP Metrics](#) section:

- `ocscp_configuration_dnssrv_nrf_non_migration_task_failure_count_total`
- `ocscp_configuration_dnssrv_nrf_non_migration_task_success_count_total`
- `ocscp_configuration_dnssrv_nrf_non_migration_task_failure`
- `ocscp_configuration_dnssrv_nrf_migration_task_failure`
- `ocscp_configuration_dnssrv_nrf_duplicate_target_detected`

The following alerts are added in the [Alerts](#) section:

- [SCPDNSSRVNRFMigrationTaskFailure](#)
- [SCPDNSSRVNRFNonMigrationTaskFailure](#)
- [SCPDNSSRVNRFDuplicateTargetDetected](#)

Maintain

If you encounter alerts at the system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.6 Support for Kubernetes Resources

This section provides you with information about the Network Policies feature and the Pod Disruption Budget feature.

3.6.1 Dual Stack Support

Using the dual stack mechanism, applications or NFs can establish connections with pods and services in a Kubernetes cluster using IPv4 or IPv6 or both simultaneously. Dual stack provides:

- coexistence strategy that allows hosts to reach IPv4 and IPv6 simultaneously.
- IPv4 and IPv6 allocation to the Kubernetes clusters during cluster creation. This allocation is applicable for all the Kubernetes resources unless explicitly specified during the cluster creation.

SCP can be deployed on CNE that supports dual stack networking. In case both IPv4 and IPv6 addresses are present in the IPADDRESS route group REST API configuration or NfService's IPEndpoint, the load balancing mechanism is used to route the traffic between IPv4 and IPv6. For more information about IPADDRESS, see "Configuring Route Groups" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Managing Dual Stack Support

Configuring IpFamilyPolicy and IpFamilies

1. To configure IpFamilyPolicy and IpFamilies, the following Helm parameters can be configured:
 - serviceIpFamilyPolicy.scpcAudit
 - serviceIpFamilyPolicy.scpcConfiguration
 - serviceIpFamilyPolicy.scpcSubscription
 - serviceIpFamilyPolicy.scpcNotification
 - serviceIpFamilyPolicy.scpcAlternateResolution
 - serviceIpFamilyPolicy.scpcCache
 - serviceIpFamilyPolicy.scpNrfProxy0auth
 - serviceIpFamilyPolicy.scpNrfproxy
 - serviceIpFamilyPolicy.scpWorker
 - serviceIpFamilyPolicy.scpMediation
 - serviceIpFamilyPolicy.scpMediationTest
 - serviceIpFamilyPolicy.scpcLoadManager
 - serviceIpFamilies.scpcAudit
 - serviceIpFamilies.scpcConfiguration
 - serviceIpFamilies.scpcSubscription
 - serviceIpFamilies.scpcNotification
 - serviceIpFamilies.scpcAlternateResolution

- `serviceIpFamilies.scpCache`
 - `serviceIpFamilies.scpNrfProxyOauth`
 - `serviceIpFamilies.scpNrfproxy`
 - `serviceIpFamilies.scpWorker`
 - `serviceIpFamilies.scpMediation`
 - `serviceIpFamilies.scpMediationTest`
 - `serviceIpFamilies.scpLoadManager`
 - `serviceSpecifications.workerServices.ipFamilyPolicy`
 - `serviceSpecifications.workerServices.ipFamilies`
 - `serviceSpecifications.workerServices.publicSignalingIPv6`
 - `serviceSpecifications.workerServices.publicSignalingIPv6Specified`
 - `scpPreferEgressTrafficOnIPv6`
2. To configure IPv4 or IPv6 used for constructing callback URI for NF profile notification from NRF, use the following parameter:

`serviceSpecifications.scpSubscriptionInfo.ip`

3. To configure NRF profiles with ipEndpoints having either IPv4 or IPv6 or both IPv4 and IPv6, use the following parameter:

`nrfProfiles.nfServices.ipEndpoints`

For more information about above mentioned Helm parameters, see *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Observe

The `ocscp_worker_5gsbi_upstream_connect_failure_total` metric is added in the [SCP Metrics](#) section.

Maintain

If you encounter alerts at the system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.6.2 Network Policies

Network Policies are an application-centric construct that allows you to specify how a pod communicates with various network entities. It creates pod-level rules to control communication between the cluster's pods and services, and to determine which pods and services can access one another inside a cluster.

Previously, the pods under deployment can be contacted by any other pods in the Kubernetes cluster without any restrictions. Now, Network Policy provides namespace-level isolation, which allows secured communications to and from SCP with rules defined in respective Network Policies. The network policies enforce access restrictions for all the applicable data flows except communication from Kubernetes node to pod for invoking container probe. For example, SCP internal microservices cannot be contacted directly by any other pods; The scp-workers in the data plane have special privileges to communicate with other NFs. Pods in the control plane can communicate with other pods only within their namespace.

Managing Support for Network Policies

Enable

To use this feature, network policies need to be applied to the namespace wherein SCP is applied.

Configure

You can configure this feature using Helm. For information about Configuring Network Policies for SCP Deployment, see *Oracle Communications Cloud Native Core, Cloud Native Environment Installation, Upgrade, and Fault Recovery Guide*.

Observe

There are no specific metrics and alerts required for the Network Policy feature.

3.6.3 Pod Disruption Budget

PodDisruptionBudget (PDB) is a Kubernetes resource that helps to achieve the high availability of scalable application services in voluntary disruptions performed by cluster administrators to manage the cluster nodes. It is defined for highly available and scalable SCP services such as scp-worker, scp-cache, and scp-nrfproxy microservices. PDB restricts the number of pods of a highly available application that are down simultaneously from voluntary disruptions, and allows safe eviction of pods when a node is drained of pods to perform maintenance. The `maxPdbUnavailable` parameter is specified in the Helm chart to determine the maximum number of pods that can remain unavailable during a voluntary disruption.

Managing Pod Disruption Budget

Enable

This feature is enabled automatically if you're deploying SCP with release 16.

Configure

You can configure this feature using Helm. For information about configuring PDB, see *Oracle Communications Cloud Native Core, Cloud Native Environment Installation, Upgrade, and Fault Recovery Guide*.

Observe

There is no specific metrics and alerts required for the Support of PDB functionality.

3.6.4 SCP Microservices and Probes

SCP microservices uses Kubernetes probes to monitor and manage the SCP application's initialization, health, and status. These are HTTP probes that provide a mechanism to identify the health status of the application. This probing occurs on a regular basis at a fixed interval, and appropriate action is taken if the probes fail.

The Kubernetes probes are as follows:

- **Liveness probes:** These probes allow you to evaluate whether an application that is running in a container is in a healthy state. If not, Kubernetes terminates the container, and the container is subjected to its restart policy.
If a container doesn't provide a liveness probe, the default state is success.
- **Readiness probes:** These probes continuously verify whether a container is ready to serve requests. If the probe returns a failed state, Kubernetes removes the IP address for the pod from the endpoints of all services.
The default state of readiness before the initial delay is failure. If a container doesn't provide a readiness probe, the default state is success.

A pod is deemed ready when all of its containers have reached a state of readiness. The use of the signal is to regulate the selection of pods that serve as backends for services. When a pod is in an unready state, it's removed from the load balancer service. The stable support of the readiness probe has been available since the release of Kubernetes 1.0.
- **Startup probes:** These probes verify whether an application within a container has started. All other probes are disabled if a startup probe is provided until it succeeds, and if a container fails this probe, the container is terminated and follows the restart policy for the pod.
This can be used to adopt liveness checks on slow-starting containers, avoiding them getting killed by the kubelet before they're up and running. The stable support of the startup probe has been available since the release of Kubernetes 1.0.

The following table describes the probes used by SCP microservices and their details.

Table 3-5 SCP Microservices and Startup Probes

SCP Microservice	Startup probes			
	Probe Support	Probe Condition	Probe Configuration	Action on Failure
SCP-Worker	Yes	If all of the conditions listed below are met, the starting probe succeeds; otherwise, the probe fails: - Routing rules received from the Notification microservice. - Routing options received from Configurations - The following data objects are received from the Configuration microservice: - WORKER_POD_OVERLOAD_ACTION_POLICY - WORKER_POD_OVERLOAD_CTRL_POLICY - WORKER_POD_OVERLOAD_DISCARD_POLICY - SERVICE_SPECIFICATIONS - SCP_NF_PROFILES - SCP_FEATURES - LOCAL_NF_AUTH2_REQUIRED - ACCESS_TO_KEN_GRANULARITY_CONFIG - The scp-worker is able to access the scp-cache if rate limiting is enabled.	- Initial delay: 35s - Probe interval: 2s - Failure threshold: 5	Pod Restart

Note

Startup probes are only implemented for SCP_worker as of the current release.

Table 3-6 SCP Microservices and Readiness Probes

SCP Microservice	Readiness probes			
	Probe Support	Probe Condition	Probe Configuration	Action on Failure
SCP-Worker	Yes	The scp-worker service is marked as ready if all the below conditions are met; otherwise, it is marked as not ready: - Routing rules received from the Notification microservice. - Routing options received from Configurations - The following data objects are received from the Configuration microservice: - WORKER_POD_OVERLOAD_ACTION_POLICY - WORKER_POD_OVERLOAD_CTRL_POLICY - WORKER_POD_OVERLOAD_DISCARD_POLICY - SERVICE_SPECIFICATIONS - SCP_NF_PROFILES - SCP_FEATURES - LOCAL_NF_AUTH2_REQUIRED - ACCESS_TO_KEN_GRANULARITY_CONFIG - The scp-worker is able to access the scp-cache if rate limiting is enabled.	- The readiness probe starts after the startup probe is successful. - Initial delay: 1s - Probe interval: 2s - Failure threshold: 5	Pod is marked as not ready, and hence its endpoint will not be created. As a result, the pods become inaccessible. For more information see, Readiness probes

Table 3-6 (Cont.) SCP Microservices and Readiness Probes

SCP Microservice	Readiness probes			
	Probe Support	Probe Condition	Probe Configuration	Action on Failure
SCP-nrfproxy	Yes	The SCP-nrfproxy service is marked as ready if all the below conditions are met; otherwise, it is marked as not ready. - Check if the NRF configuration is available and can be read from the NRF Profiles table in the cnDBtier service. - Check if the list of prioritized NRFs is available as per preference.	- Initial delay: 30s - Probe interval: 10s - Failure threshold: 5	Pod is marked as not ready, and hence its endpoint will not be created. As a result, the pods become inaccessible. For more information see, Readiness probes
SCP-loadManager	Yes	The readiness probe for the SCP-loadManager service returns false during application initialization, and after the application is initialized, it starts returning the true value.	- Initial delay: 5s - Probe interval: 2s - Failure threshold: 5	Pod is marked as not ready, and hence its endpoint will not be created. As a result, the pods become inaccessible. For more information see, Readiness probes
SCP-nrfProxy-Oauth	Yes	The readiness probe for the scp-nrfproxy-oauth service returns false during application initialization, and after the application is initialized, it starts returning the true value.	- Initial delay: 5s - Probe interval: 2s - Failure threshold: 5	Pod is marked as not ready, and hence its endpoint will not be created. As a result, the pods become inaccessible. For more information see, Readiness probes
SCP-Cache	Yes	The scp-cache service is marked as ready if the below conditions are met; otherwise, it is marked as not ready. Check if application maps in scp-cache service are initialized.	- Initial delay: 5s - Probe interval: 2s - Failure threshold: 5	Pod is marked as not ready, and hence its endpoint will not be created. As a result, the pods become inaccessible. For more information see, Readiness probes

Table 3-6 (Cont.) SCP Microservices and Readiness Probes

SCP Microservice	Readiness probes			
	Probe Support	Probe Condition	Probe Configuration	Action on Failure
SCPC-Notification	Yes	The scpc-notification service is marked as ready if the below conditions are met; otherwise, it is marked as not ready. - Check whether the following tables have been created and populated: - SCP_NF_PROFILES - NRF_NF_DETAILS - CANARY_RELEASES - ROUTING_OPTIONS - SYSTEM_OPTIONS - VIRTUAL_SERVICES - SERVICE_ENTRIES	- Initial delay: 30s - Probe interval: 10s - Failure threshold: 5	Pod is marked as not ready, and hence its endpoint will not be created. As a result, the pods become inaccessible. For more information see, Readiness probes
SCPC-Audit	Yes	The scp-audit service is marked as ready if all the below conditions are met; otherwise, it is marked as not ready. - Check if the NRF configuration is available and can be read from the NRF Profiles table in the cnDBtier service. - For Release 15, check if virtual services are created for NRF profiles. Note: Release 15 deployment model is not supported from SCP 23.4.0.	- Initial delay: 30s - Probe interval: 10s - Failure threshold: 5	Pod is marked as not ready, and hence its endpoint will not be created. As a result, the pods become inaccessible. For more information see, Readiness probes

Table 3-6 (Cont.) SCP Microservices and Readiness Probes

SCP Microservice	Readiness probes			
	Probe Support	Probe Condition	Probe Configuration	Action on Failure
SCPC-Configuration	Yes	The scpc-configuration service is marked as ready if the below conditions are met; otherwise, it is marked as not ready. Check if the cnDBtier service is accessible and the required DB tables are available or created.	- Initial delay: 30s - Probe interval: 10s - Failure threshold: 5	Pod is marked as not ready, and hence its endpoint will not be created. As a result, the pods become inaccessible. For more information see, Readiness probes
SCPC-Subscription	Yes	The scpc-subscription service is marked as ready if all the below conditions are met; otherwise, it is marked as not ready. Check if the cnDBteir service is accessible and if NRF and SCP profiles are available or created in the cnDBteir service.	- Initial delay: 30s - Probe interval: 10s - Failure threshold: 5	Pod is marked as not ready, and hence its endpoint will not be created. As a result, the pods become inaccessible. For more information see, Readiness probes
SCPC-Alternate-resolution	Yes	The scpc-alternate resolution service is marked as ready if all the below conditions are met; otherwise, it is marked as not ready. - Check if the cnDBtier service is accessible and if DNS SRV configuration for alternate route groups is available or created in the cnDBtier service. - Validate if system options are synced from the scpc-configuration service.	- Initial delay: 30s - Probe interval: 10s - Failure threshold: 5	Pod is marked as not ready, and hence its endpoint will not be created. As a result, the pods become inaccessible. For more information see, Readiness probes

Table 3-7 SCP Microservices and Liveness Probes

SCP Microservice	Liveness probes			
	Probe Support	Probe Condition	Probe Configuration	Action on Failure
SCP-Worker	Yes	- Threadwatchdog controls the Liveness probe. - ThreadWatchDog monitors all the threads running in the pod at a fixed interval. - If threadwatchdog identifies that a thread is stuck, then the liveness probe is marked as failed.	- The liveness probe starts after the startup probe is successful. - Initial delay: 1s - Probe interval: 2s - Failure threshold: 5	Pod restart
SCP-nrfproxy	Yes	- Threadwatchdog controls the Liveness probe. - ThreadWatchDog monitors all the threads running in the pod at a fixed interval. - If threadwatchdog identifies that a thread is stuck, then the liveness probe is marked as failed.	- Initial delay: 180s - Probe interval: 2s - Failure threshold: 5	Pod restart
SCP-loadManager	Yes	- Threadwatchdog controls the Liveness probe. - ThreadWatchDog monitors all the threads running in the pod at a fixed interval. - If threadwatchdog identifies that a thread is stuck, then the liveness probe is marked as failed.	- Initial delay: 180s - Probe interval: 2s - Failure threshold: 5	Pod restart

Table 3-7 (Cont.) SCP Microservices and Liveness Probes

SCP Microservice	Liveness probes			
	Probe Support	Probe Condition	Probe Configuration	Action on Failure
SCP-nrfProxy-Oauth	Yes	- Threadwatchdog controls the Liveness probe. - ThreadWatchDog monitors all the threads running in the pod at a fixed interval. - If threadwatchdog identifies that a thread is stuck, then the liveness probe is marked as failed.	- Initial delay: 180s - Probe interval: 2s - Failure threshold: 5	Pod restart
SCP-Cache	Yes	The liveness probe for scp-cache always returns true.	- Initial delay: 180s - Probe interval: 2s - Failure threshold: 5	N/A
SCPC-Notification	Yes	- Threadwatchdog controls the Liveness probe. - ThreadWatchDog monitors all the threads running in the pod at a fixed interval. - If threadwatchdog identifies that a thread is stuck, then the liveness probe is marked as failed.	- Initial delay: 180s - Probe interval: 2s - Failure threshold: 5	Pod restart
SCPC-Audit	Yes	- Threadwatchdog controls the Liveness probe. - ThreadWatchDog monitors all the threads running in the pod at a fixed interval. - If threadwatchdog identifies that a thread is stuck, then the liveness probe is marked as failed.	N/A	Pod restart

Table 3-7 (Cont.) SCP Microservices and Liveness Probes

SCP Microservice	Liveness probes			
	Probe Support	Probe Condition	Probe Configuration	Action on Failure
SCPC-Configuration	Yes	- Threadwatchdog controls the Liveness probe. - ThreadWatchDog monitors all the threads running in the pod at a fixed interval. - If threadwatchdog identifies that a thread is stuck, then the liveness probe is marked as failed.	N/A	Pod restart
SCPC-Subscription	Yes	- Threadwatchdog controls the Liveness probe. - ThreadWatchDog monitors all the threads running in the pod at a fixed interval. - If threadwatchdog identifies that a thread is stuck, then the liveness probe is marked as failed.	N/A	Pod restart
SCPC-Alternate-resolution	Yes	- Threadwatchdog controls the Liveness probe. - ThreadWatchDog monitors all the threads running in the pod at a fixed interval. - If threadwatchdog identifies that a thread is stuck, then the liveness probe is marked as failed.	N/A	Pod restart

3.7 SCP Response Timeout Extension

Currently, users can configure a response timeout, a time interval to receive a response by SCP, up to 10 seconds.

This feature allows users to set a maximum response timeout of up to 15 seconds for all the services supported by the SCP. Once the timeout expires, SCP initiates rerouting to an available alternate NF or pod.

Note

SCP raises an alert if total forwarded request rate with a response time greater than 10-seconds cross 200 requests per second. If the number of such requests is high, then overload control based on pending transactions or CPU-based overload control will be triggered.

Managing SCP Response Timeout in Routing Option Configuration**Enable**

This feature is enabled automatically at the time of SCP deployment.

Configure

You can configure this feature using REST APIs as described in "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

- The `ocscp_metric_upstream_service_time_total` metric is updated with new buckets for response timeout in the [SCP Metrics](#) section.
- Added `SCPHighResponseTimeFromProducer` alert in the [SCPHighResponseTimeFromProducer](#) section.

Maintain

If you encounter alerts at the system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.8 Overload Control Based on the Overload Control Information Header

SCP, SEPP, producer NFs, and consumer NFs get overloaded when they operate beyond their signaling capacity. An enhancement is required to include Overload Control Information (OCI) in 5G Service Based Interface (SBI) messages to convey the overload capacity of NFs to their peers.

Using the Overload Control Based on the Overload Control Information Header feature, SCP, SEPP, consumer NFs, and producer NFs can gracefully reduce their incoming signaling traffic by conveying consumer NFs to reduce sending service requests or by conveying producer NFs to reduce sending notification requests, respectively, so that these entities or peer NFs can process the traffic according to their configured capacities. After receiving the `3gpp-Sbi-Oci` header from consumer NFs, producer NFs perform the signaling reduction toward the consumer NF only for the notifications or callback requests according to the overload scope and not for any NF services that may be produced by the same NF.

Depending on SCP's configuration, SCP can:

- send its load information in the OCI header with reduction metrics based on the REST API configuration.
- reduce traffic toward the peer based on the OCI received from a peer.

OCI is conveyed within the 3gpp-Sbi-Oci header. When SCP, SEPP, consumer NF, or producer NF experiences overload conditions, it sends overload condition information through the 3gpp-Sbi-Oci header, which is tagged along with the messages to peers.

OCI contains Overload Timestamp, Overload-Reduction-Metric, Period-of-Validity, and Scope attributes in the 3gpp-Sbi-Oci header.

A sample format of the 3gpp-Sbi-Oci header for an NF instance is as follows:

Timestamp: "Mon, 04 Dec 2023 08:49:37 GMT"; Period-of-Validity: 75s; Overload-Reduction-Metric: 50%; NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8

The following table describes attributes of the 3gpp-Sbi-Oci header format:

Table 3-8 3gpp-Sbi-Oci Header Attributes

Attribute	Description
Timestamp	Indicates the time when the OCI was generated. This attribute is used by the OCI recipient such as peers or SCP, to decide whether the newly received OCI is the same as the previously received OCI or different. The OCI recipient can overwrite any stored OCI for a peer NF, NF set, NF service, NF service set, Callback URI, SCP, or SEPP with the newly received OCI if the new OCI is more recent than the previously stored OCI. Example: Timestamp: "Mon, 04 Dec 2023 08:49:37 GMT".
Period-of-Validity	Indicates the duration during which the overload condition specified by the 3gpp-Sbi-Oci header is considered valid until overridden by subsequent new OCIs. Example: Period-of-Validity: 75s.
Overload-Reduction-Metric	Indicates the percentage of traffic reduction requested by the OCI sender to the OCI recipient. This value can range from 0 to 100. Example: Overload-Reduction-Metric: 50%.
Scope	Indicates the message requests or notification requests to which the OCI is applicable. It identifies the traffic that the OCI sender requests the receiver to process in accordance with the OCI. Example: NF-Instance: 54804518-4191-46b3-955c-ac631f953ed8. List of supported olcScope in the OCI header at SCP: • nfProducerScope – NF-Instance – NF-Service-Instance • nfConsumerScope – Callback-Uri – NF-Instance – NF-Service-Instance – NF-Instance and Service-Name • scpScope – SCP-FQDN • seppScope – SEPP-FQDN

Conveyance of OCI

The following scenarios explain the conveyance of OCI:

Peer OCI Enforcement

SCP performs OCI enforcement based on the OCI header received from the peer and the REST API configuration.

When a peer NF is overloaded, it sends OCI to SCP so that SCP can throttle the traffic destined for that peer NF. Based on the `ociEnforcementAction` REST API parameter configuration, SCP can perform alternate routing to another equivalent peer NF or directly send an error response to the peer consumer NF. SCP relays the OCI received from the peer NF to other peer NFs based on the `relayPeerOci` REST API parameter configuration as described in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. For example, if the `Overload-Reduction-Metric` attribute of the OCI header is 10%, the producer NF conveys SCP to reduce the message rate by 10%. If SCP has 100 messages to be routed to that producer NF, SCP will route 90 messages to the target producer NF. To process the pending 10 messages, SCP can reroute the messages to another equivalent NF or send an error response to the consumer NF based on the `ociEnforcementAction` REST API parameter.

After the expiry of peer OCI `Period-of-Validity` attribute, SCP can continue sending traffic toward the peer immediately or incrementally based on the `ociTrafficRecoveryPolicy` REST API parameter.

Self-OCI Generation and Conveyance

SCP conveys its overload control information, which is known as self-OCI, with its peer NFs when SCP is overloaded.

The overload condition of SCP is determined by the CPU utilization of all SCP-Worker pods. SCP is overloaded when the average CPU utilization of all the pods exceeds the configured OCI threshold levels.

When SCP is overloaded, SCP generates OCI based on the threshold levels such as `WARN`, `MINOR`, `MAJOR`, and `CRITICAL`, configured using the `ociThresholdConfig` REST API parameter as described in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. SCP sends the self-generated OCI only when there is any message to be routed to the peer NF.

SCP starts conveying OCI to peers when it gets overloaded based on the configured threshold level. OCI is conveyed for the following scenarios:

- When OCI threshold changes: Whenever SCP's traffic load changes resulting in OCI threshold changes, SCP creates a new OCI based on the current threshold level configuration values and current timestamp.
- When the `Period-of-Validity` attribute expires: When the `Period-of-Validity` attribute of the last sent OCI expires, SCP creates a new OCI based on the current threshold level configuration values and current timestamp.
- Periodically: After every `scpOciConveyanceInterval` interval, SCP sends the same OCI that was sent last time to the peer NF.

Managing Overload Control Information

Enable

By default, this feature is disabled. You can use the `oci` option to enable or disable this feature as described in [Configuring SCP Features](#).

Configure

You can configure this feature using REST API and the CNC Console.

- **Configure using CNC Console:** For information about CNC Console configurations, see
 - [Configuring OCI](#)
 - [Configuring OCI Threshold Levels](#)
- **Configure using REST API:** For information about configuring REST APIs, see "Configuring SCP Features", "Configuring Overload Configuration Information Threshold" and "Configuring Overload Configuration Information" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

The following metrics are added in the [Metrics](#) section:

- ocscp_worker_peer_oci_conveyance_total
- ocscp_worker_oci_conveyance_total
- ocscp_worker_oci_enforced_total
- ocscp_worker_rx_unsupported_oci_scope_total
- ocscp_metric_nf_oci_rx_total

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.9 Custom Header Support at SCP

To enhance efficiency across the converged networks and optimize the 5G core traffic, Custom Header Support at SCP feature is introduced.

Support for 3gpp-Sbi-Custom-UDR-Nf-Group-Id

When an NF sends a SBI message request or response with the custom header, 3gpp-Sbi-Custom-UDR-Nf-Group-Id, SCP sends the message request or response, including the custom header, to the target NF.

Managing Custom Header Support at SCP

Enable

This feature doesn't require a flag to be enabled or disabled and is enabled by default.

Configure

There's no specific configuration required for this feature.

Maintain

If you encounter alerts at the system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.10 Support for HTTPS

Hypertext Transfer Protocol secure (HTTPS) is an extension of Hypertext Transfer Protocol (HTTP). HTTPS establishes a secure connection between Network Functions (NFs) in a 5G network and uses Transport Layer Security (TLS) to encrypt the communication protocols.

With the support of HTTPS connections, SCP accepts secured ingress connection requests from a consumer NF and establishes a secured egress connection with a producer NF. HTTPS provides confidentiality and integrity protection to 5G Service Based Interface (SBI) messages.

SCP processes 5G SBI messages over HTTPS connection based on the `sbiProxySslConfigurations` parameter configuration as specified in the `custom-values.yaml` file. For more information, see *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*. The `sbiProxySslConfigurations` configuration authorizes both ingress and egress connections from consumer NFs and producer NFs, respectively with SCP.

SCP supports a single and multiple TLS certificates to enable HTTPS as described in [TLS Certificate Configuration Enhancement](#). SCP supports two-way TLS authentication for HTTPS connections and accesses Certificate Authority (CA) from Kubernetes secret.

Note

- This feature supports HTTPS for inter-NF communication but not intra-NF communication, that is, communication between two microservices within SCP.
- SCP supports both HTTP and HTTPS concurrently. You can select to expose service IP for HTTP, HTTPS, or both.
- This feature supports TLSv1.2.

Managing Support for HTTPS

Enable

This feature is enabled by setting the following Helm parameters in the `custom-values.yaml` file:

- Enabling HTTPS for ingress connections:
 - For Release 16 deployment: The `scpProfileInfo.scpInfo.scpPorts.https` parameter enables SCP to establish ingress connection with consumer NFs. You must set the `https` port to the same value as the `publicSignallingPortHttps` parameter.
 - For Release 15 deployment: To enable HTTPS for ingress connections, a new `nscp-5g-sbi-proxy` service with scheme as `https` must be added in the `nfServices` section of `scpProfileInfo` in the `custom-values.yaml` file. You must set the port to the same value as the `publicSignallingPortHttps` parameter.

Note

Release 15 deployment model is not supported from SCP 23.4.0.

Example of NF Service for HTTPS:

```
# - serviceInstanceId: f86b54b7-aef9-4c78-b346-3bfb7f380813
#   serviceName: nscp-5g-sbi-proxy
#   fqdn: <scprel>-scp-worker.scpsvc.svc.cluster.local
#   scheme: https
#   priority: 0 # Default value is 0
#   capacity: 100 # Default value is 100
#   load: 0
#   nfServiceStatus: REGISTERED
#   ipEndpoints:
#     - ipv4Address: 10.104.121.240 # this IP should be same as
      "publicSignalingIPHttps" of SCP mentioned above.
#     port: '9443' # this port should be same as
      "publicSignalingPortHttps" of SCP mentioned above.
#   apiPrefix: # Ignored in SCP processing
#   versions: # Ignored in SCP processing, but kept for
      registering with NRF
#     - apiFullVersion: 1.0.0
#     apiVersionInUri: v1
```

- Enabling HTTPS for egress connections: The `nativeEgressHttpsSupport` parameter must be set to `true` to enable SCP to establish a connection with producer NFs over HTTPS. By default, this value is set to `false`. If this value is set to `false`, then all the connections with producer NFs are established through HTTP. For more information about above mentioned parameters, see *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Note

You can configure the required Helm parameters for this feature if you have performed a fresh installation of SCP or upgraded SCP within the same release version, for example, 24.2.0 to 24.2.0.

Configure

`sbiProxySslConfigurations` and `ddSslConfiguration` Helm parameters are introduced as part of this feature. To use this feature, complete the configuration as described in "Global Parameters" in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Observe

The `ocscp_metric_ssl_certificate_expire_total` metric is updated in the [Metrics](#) section.

Maintain

If you encounter alerts at the system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.10.1 TLS Certificate Configuration Enhancement

SCP uses Hypertext Transfer Protocol Secure (HTTPS) to establish secured ingress and egress connections with consumer NFs and producer NFs, respectively. These communication protocols are encrypted using Transport Layer Security (TLS).

In SCP 23.2.0, SCP used to support a single TLS certificate configuration to enable HTTPS.

From 23.3.0, this enhancement enables SCP to configure different Transport Layer Security (TLS) certificates on ingress and egress interfaces. It also allows users to provide primary and secondary certificates for each interface.

Multiple TLS certificates can be configured across the following communication interfaces:

- Egress connections on 5G Service Based Architecture (SBA): SCP acts as a client while sending message requests to producer NFs, producer SCP-P, or NRF. For example:
 - A consumer SCP-C sends health check request to a producer SCP-P.
 - SCP forwards notifications to consumer NFs.
 - SCP forwards Service, Discovery, or Access Token requests to NRF or producer NFs.
- Ingress connections on 5G SBA: SCP acts as a server to process message requests received from consumer NFs, SCP-C, or NRF. For example:
 - SCP-P receives health check request from SCP-C.
 - SCP receives notifications from NRF or NF producer.
 - SCP receives service or health check requests from consumer NFs.
- Oracle Communications Network Analytics Data Director (OCNADD) connection on message feed: SCP acts as a client toward OCNADD on message feed interfaces, for example, SCP sends message feed data to OCNADD.

Note

You can use a single or multiple TLS certificates to enable HTTPS.

If the primary TLS certificate expires or is about to expire, SCP switches to an alternate or secondary TLS certificate. If there is no alternate TLS certificate configured, SCP uses a single certificate to enable HTTPS. At each communication interface such as ingress, egress, and OCNADD, you can configure the `caBundle` Helm parameter to manage the Certificate Authority (CA) bundle. The CA can be the same or different at these interfaces. For example, CA signing the TLS certificate at the ingress communication interface can be the same or different from the CA at the egress communication interface.

Note

It is recommended to maintain multiple TLS certificates with overlapping expiry time to avoid service interruption during certificate expiry. For example, if the primary certificate validity is changed from T0 and T1, then the validity of the secondary certificate must be changed from T0 to T2 so that when the primary certificate expires, the secondary certificate gets activated within the expiry period of the primary certificate, where, $T0 < T1 < T2$.

By default, SCP uses a single TLS certificate to enable HTTPS. To enable support for multiple TLS certificates, you can modify the `sbiProxySslConfigurations` Helm parameter configurations for ingress and egress connections and the `ddSslConfiguration` Helm parameter for OCNADD connections.

SCP uses the Helm parameters `sslCertExpiryMinorThreshold`, `sslCertExpiryMajorThreshold`, and `sslCertExpiryCriticalThreshold` to set up minor, major, and critical alerts for TLS certificate expiration. These alerts are triggered at the specified times to signal the impending expiry of the certificates.

SCP provides the `terminateTLSConnsOnCertExpiry` Helm configuration at the egress communication interface to determine whether to terminate or maintain the existing HTTPS connections when the configured TLS certificate is updated or renewed. When the TLS certificate expires, then SCP:

- maintains the existing HTTPS connections that were using the expired certificates.
- creates new HTTPS connections which uses the updated or renewed TLS certificate.

For more information about configuring Helm parameters for multiple TLS certificates, see "Global Parameters" in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

3.10.2 Support for TLS 1.3

SCP uses Hypertext Transfer Protocol Secure (HTTPS) to establish secure connections with consumer NFs and producer NFs, respectively. These communication protocols are encrypted using Transport Layer Security (TLS).

TLS comprises the following components:

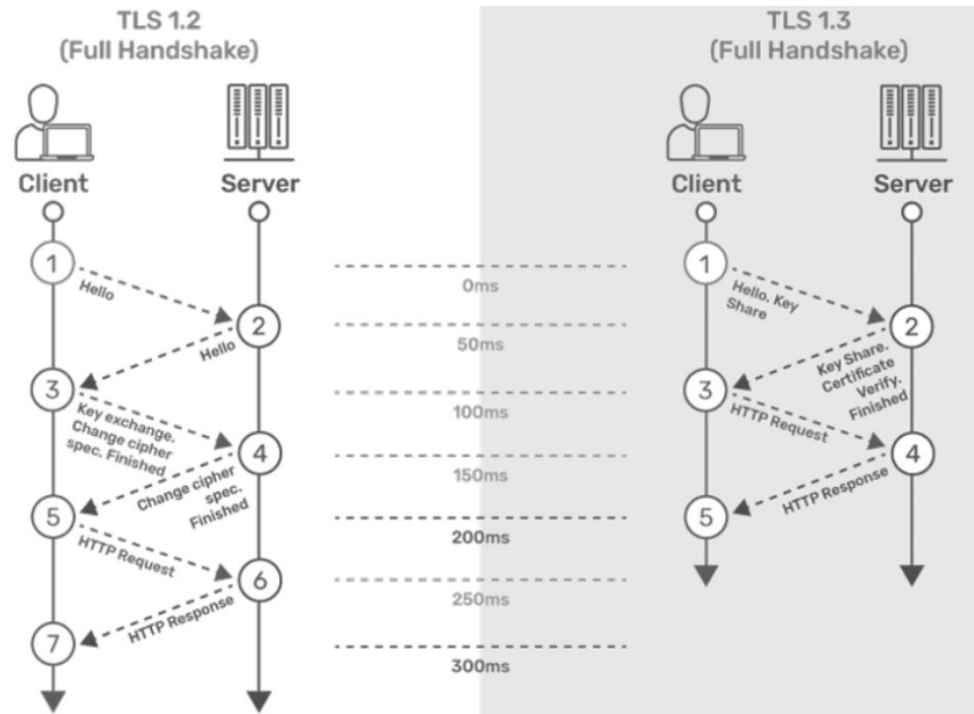
- **Handshake Protocol:** Exchanges the security parameters of a connection. Handshake messages are supplied to the TLS record layer.
- **Record Protocol:** Receives the messages to be transmitted, fragments the data into multiple blocks, secures the records, and then transmits the result. Received data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

This feature enables the support for TLS 1.3 to all consumer NFs, producer NFs, the Data Director, SBI Interfaces, and any interfaces previously supporting TLS 1.2. Support for TLS 1.2 will remain available.

TLS 1.2 and TLS 1.3 Handshake

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

Figure 3-2 TLS 1.2 and TLS 1.3 Handshake

**TLS 1.2**

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

Step 2: To respond to the "client hello" message, the server sends a 'server hello' message. This message contains the CipherSuite that the server has selected from the options provided by the client. The server also sends its certificate, along with the session ID and another random value.

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

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

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

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

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

TLS 1.3

Step 1: The connection or handshake begins when the client sends a "client hello" message to the server, which includes the list of supported cipher suites and the client's key share for the specific key agreement protocol.

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

Step 3: The client verifies the server certificate, generates keys since it has the server's key share, and then sends the "Client Finished" message along with an HTTP request.

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

Comparison Between TLS 1.3 and TLS 1.2

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

Table 3-9 Comparison of TLS 1.3 and TLS 1.2

Feature	TLS 1.3	TLS 1.2
TLS Handshake	- The initial handshake involves the exchange of key shares, ensuring cryptographic security. - A typical handshake in TLS v1.3 involves the exchange of up to 3 packets, thereby facilitating quicker and more responsive connections.	- The initial handshake occurs in clear text. - A typical handshake in TLS v1.2 involves a more extended process, typically requiring 5-7 packets for completion, resulting in slower connection establishment times.
Cipher Suites and Privacy	- Improving security by narrowing the selection of cipher suites, which streamlines configuration and reduces errors. Supported algorithms are free from known vulnerabilities, thus making the cipher suites more secure. - TLS_AES_128_GCM_SHA256 - TLS_AES_256_GCM_SHA384 - TLS_CHACHA20_POLY1305_SHA256	- Supporting a wide range of cipher suites that are vulnerable to attacks results in less secure cipher suites. - TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 - TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 - TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 - TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 - TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 - TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 - TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256 - TLS_DHE_RSA_WITH_AES_128_GCM_SHA256
Round-Trip Time	Efficient session resumption modes, such as Zero Round Trip Time (0-RTT), enable clients to resume sessions without additional round trips. Data can be transmitted in the initial message using Pre-Shared Keys (PSK), further enhancing the efficiency of the communication process.	TLS 1.2 does not support a 0-RTT mode for resuming sessions with reduced round-trip time.
Perfect Forward Secrecy (PFS)	TLS v1.3 supports Perfect Forward Secrecy (PFS). PFS ensures that each session key is entirely independent of long-term private keys, which are used over an extended period for decrypting encrypted data.	TLS v1.2 does not provide PFS.

Table 3-9 (Cont.) Comparison of TLS 1.3 and TLS 1.2

Feature	TLS 1.3	TLS 1.2
Performance	The handshake process is optimized to reduce latency by minimizing the number of required round trips, resulting in a more responsive connection.	The handshake process involves multiple round trips between the client and server, may result in increased latency and a less responsive connection.

TLS 1.3 handshake offers the following improvements over earlier versions:

- The symmetric encryption algorithms are removed and Authenticated Encryption with Associated Data (AEAD) algorithms are supported.
- In TLS 1.3, all handshake messages after the ServerHello are encrypted.
- A zero round-trip time (0-RTT) mode is added in the handshake process, enabling fewer round trips compared to TLS 1.2. Additionally, it utilizes cryptographic algorithms that are faster.
- Static Rivest-Shamir-Adleman (RSA) and Diffie-Hellman cipher suites are removed, ensuring that all public-key-based key exchange mechanisms now Provide Forward Secrecy (PFS).
- TLS 1.3 has better security than TLS 1.2, addressing known vulnerabilities in the handshake process.
- TLS 1.3 eliminates data compression.
- Version Downgrade Protection in TLS v1.3 is a security mechanism that prevents attackers from using an older, potentially less secure version of the TLS protocol. This protection is achieved through the exchange of a "Finished" message between the client and server, which includes a Message Authentication Code (MAC) covering all preceding handshake messages. This mutual verification process ensures that negotiated parameters remain unaltered by potential attackers. This helps ensure that connections are established using the latest, most secure version of TLS available, reducing the risk of vulnerabilities associated with older protocol versions.

Managing Support for TLS

Using TLS Versions

Using TLS 1.3

If you want to use TLS 1.3, perform the following procedure:

1. In the `custom_values.yaml` file, configure the following Helm parameters:
 - `sbiProxySslConfigurations.server` to `TLSv1_3`
 - `sbiProxySslConfigurations.client[0]` to `TLSv1_3`

Note

By default, the helm parameter is set to `TLSv1.3,TLSv1.2`.

2. Add the following ciphers to the `cipherSuitesTlsV1_3` parameter:
 - `TLS_AES_128_GCM_SHA256`
 - `TLS_AES_256_GCM_SHA384`

- TLS_CHACHA20_POLY1305_SHA256

Using TLS for OCNADD

By default, SCP uses a single TLS certificate to enable HTTPS. To enable support for multiple TLS certificates, you can modify the `sbiProxySslConfigurations` Helm parameter configurations for ingress and egress connections and the `ddSslConfiguration` Helm parameter for OCNADD connections.

Using TLS 1.2

If you want to use only TLS 1.2, perform the following procedure:

1. In the `custom_values.yaml` file, configure the following Helm parameter:
 - `sslEnabledProtocol` to `TLSv1.2`

Note

By default, the helm parameter is set to `TLSv1.3`

2. Add the following ciphers to the `cipherSuitesTlsV1_2` parameter:
 - TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384
 - TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384
 - TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256
 - TLS_DHE_RSA_WITH_AES_256_GCM_SHA384
 - TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256
 - TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256
 - TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256
 - TLS_DHE_RSA_WITH_AES_128_GCM_SHA256

Configure

You can configure this feature using Helm. For information about Helm configurations, see "Global Parameters" in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Observe

The following metrics are added in the [Metrics](#) section:

- `ocscp_worker_https_ingress_connection_success_total`
- `ocscp_worker_https_ingress_connection_failure_total`
- `ocscp_worker_https_egress_connection_success_total`
- `ocscp_worker_https_egress_connection_failure_total`
- `ocscp_metric_ssl_certificate_expire_total`

The following alerts are added in the [Alerts](#) section:

- [SCPWorkerSSLCertificateOnCriticalExpiry](#)
- [SCPWorkerSSLCertificateOnMajorExpiry](#)
- [SCPWorkerSSLCertificateOnMinorExpiry](#)

- [SCPIngressConnectionEstablishmentFailure](#)
- [SCPEgressConnectionEstablishmentFailure](#)

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.10.3 Support for Automated Certificate Lifecycle Management

In SCP 23.3.x and earlier releases, X.509 and Transport Layer Security (TLS) certificates were managed manually. When multiple instances of SCP were deployed in a 5G network, certificate management, such as certificate creation, renewal, removal, and so on, was tedious and error-prone.

Using the Support for Automated Certificate Lifecycle Management feature, you can integrate SCP with OCCM to support automation of certificate lifecycle management. OCCM manages TLS certificates stored in Kubernetes secrets by integrating with Certificate Authority (CA) using the Certificate Management Protocol Version 2 (CMPv2) protocol. OCCM obtains and signs TLS certificates and stores them in Kubernetes secrets of SCP namespace. For more information about OCCM, see *Oracle Communications Cloud Native Core, Certificate Management User Guide*.

Using this feature, you can monitor, create, recreate, and renew TLS certificates based on their validity. For information about creating Kubernetes secrets, see "Configuring SSL or TLS Certificates to Enable HTTPS" in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

3.11 Routing Options Enhancement to Control SCP Routing Features

At SCP, there are a number of routing features such as Circuit Breaking, Outlier Detection and so on that proactively track the producer NF to determine the load conditions and health status of the producer NF and perform alternate routing of incoming SBI messages that are targeted towards the degraded producer NFs.

With this enhancement, SCP supports proactive rerouting, in which a user can control the applicability of routing features at the NF service level by providing options to enable or disable rerouting.

The following routing features at the NF service level can be enabled or disabled:

- Congestion control based on Producer load
 - Using producer load from NRF
 - Using producer load in the LCI header
- Circuit Breaking
- Outlier Detection

Note

Alternate routing based on the following conditions are not impacted by this enhancement:

- Error received from the producer
- Request a timeout at the SCP
- Transport error at SCP during connection creation

Managing Routing Options Enhancements**Enable**

You can enable this feature by setting `odEnabled`, `loadBasedCongestionControlEnabled`, `cbEnabled` parameters to true. For more information, see [Configuring Global Options](#), [Configuring Routing Options](#), and [Routing Config Set](#) respectively.

Configure

You can configure this feature using REST APIs as described in "Routing Option Configuration", "Congestion Control Configuration", "Routing Config Set", "Circuit Breaking Configuration", "Configuring Routing Option", and "Configuring System Option" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Maintain

If you encounter alerts at the system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.12 Enhanced NF Status Processing

In older SCP releases, SCP removes routing rules when SCP receives an NFProfile with the SUSPENDED status.

With the enhanced NF Status Processing feature, Service Communication Proxy (SCP) provides the following 5G Service Based Interface (SBI) message request routing modes when an NFProfile is in the SUSPENDED state:

- Mode 1: SCP does not route message requests to producer NFs that are in the SUSPENDED state.
- Mode 2: SCP routes the message request to the target producer NF even if it is in the SUSPENDED state.
- Mode 3: SCP routes the message request to producer NFs that are in the SUSPENDED state only if alternate REGISTERED producer NFs are unavailable.

Note

When an NFProfile is SUSPENDED, it remains in the SUSPENDED state for an infinite duration unless NRF marks it as REGISTERED, or it gets DE-REGISTERED.

In case of Mode 2 routing, when SCP receives a 5G SBI message request from a consumer NF, SCP considers all NFs that are configured using the `suspendedStateRouting` REST API parameter to route the message request.

In case of Mode 3 routing, when SCP receives a 5G SBI message request from a consumer NF, if the target producer NF is SUSPENDED, SCP routes the message to the first alternate producer NF, which is in the REGISTERED state. If the alternate REGISTERED producer NF is unavailable or responds with an error response, SCP routes the message request to the next available producer NF, which is in the REGISTERED state.

After exhausting all the routing attempts on REGISTERED producer NFs, if alternate routing attempts are available, SCP routes the message request to the SUSPENDED producer NFs if the `enhanced_nf_status_processing.enhancedSuspendedStateRouting` REST API parameter is set for a particular NF type. Otherwise, SCP returns the same error response to the consumer NF that it received due to the routing attempted on the last producer NF. SCP determines the target producer NF using the "3gpp-Sbi-Target-apiRoot" header and NF Set ID in the message request.

Note

You must not configure the same NF type for `enhancedSuspendedStateRouting` and `suspendedStateRouting` for both Mode 2 and Mode 3.

Managing Enhanced NF Status Processing

Enable

You can enable this feature using the CNC Console as described in [Configuring SCP Features](#).

Configure

You can enable and configure one of the following routing modes of this feature:

- Mode 1: Set `enhanced_nf_status_processing` to false.
- Mode 2: Set `enhanced_nf_status_processing` to true and configure `suspendedStateRouting`.
- Mode 3: Set `enhanced_nf_status_processing` to true and `enhancedSuspendedStateRouting` must be present.

For more information about the above mentioned REST API parameters, see "Configuring SCP Features" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about metrics, see [Metrics](#).

Maintain

If you encounter alerts at the system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.13 Support for Default Notification Message Routing

During NF profile registration or update, NF has the ability to publish the Notification URI for default notifications. This is crucial due to the expected default Subscriptions in NF profiles. SCP plays a vital role in this process by creating routing rules for the purpose of routing 5G SBI messages. SCP achieves this by obtaining knowledge of the 5G topology from the NRF through NF status notifications. Previously, the SCP's topology learning did not include the default notification callback URI information published in the defaultNotificationSubscription attribute of the NF profile. Consequently, SCP remained unaware of these default notification callback URIs and was unable to do alternate routing of received default notification messages.

SCP has now significantly improved its topology learning process by incorporating the default notification URI and endpoint from both the NF profile and NF service level. As a result, SCP is now fully informed about these default notification callback URIs and possesses the capability to efficiently do routing and alternate routing of received default notification messages.

Managing Default Notification Message Routing

Enable

This feature is enabled by default in SCP at the time of deployment.

Configure

You can view or edit the default notification message subscription feature using the CNC Console. For information about this feature, see "Viewing or Editing the Default Notification Subscriptions section" in [Configuring NF Rule Profile](#) to view the default notification message subscriptions that are present under NF service level or NF profile level.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.13.1 Default Notification Routing Behavior at SCP

The expected behavior at SCP based on the presence of various headers is as follows:

Table 3-10 Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
"/scp-default-sub-notify-uri"	Absent	Present	Present	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	NA	SCP performs the discovery and gets the callback URI from the selected producer's profile for the given notification type. Note: Only apiversion is considered from 3gpp-Sbi-Callback.
"/scp-default-sub-notify-uri"	Absent	Present	Absent	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	NA	400 Bad request cause: query parameter missing (TS29.500)

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
"/scp-default-sub-notify-uri"	Absent	Absent	Present	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	NA	SCP performs the discovery and gets the callback URI from the selected producer's profile for the given notification type at the NF profile level defaultnotification/subscription parameters. This is the default notification due to the presence of the 3gpp-Sbi-Discovery-notification-type header. apiversion =1 Note: Only apiversion is considered from 3gpp-Sbi-Callback.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
										If not, the apiversion will be assumed to be 1.
"/scp-default-sub-notify-uri"	Absent	Absent	Absent	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	NA	400 Bad request cause: query parameter is missing (TS29.500)
non "/scp-default-sub-notify-uri"	Absent	Present	Present	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Absent	SCP performs the discovery and gets the callback URI from the selected producer's profile for the given notification type. SCP modifies the :path header as per the selected notification endpoint.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
non "/" scp-default-sub-notify-uri"	Absent	Present	Absent	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Absent	Selection of NF is dependent on the discovery response. The path will remain the same as it was when the consumer made their original request. Furthermore, the notification processing is non-default. For example, explicit notification since 3gpp-Sbi-Callback header is present. Note: The callbackUri from defaultNotificationSubscriptions section is not considered. It goes with the

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
										typical rule match. For more information, see Note at the bottom of this table if the notification message is without a service name.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
non "/" scp-default-sub-notify-uri"	Absent	Absent	Present	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Absent	SCP performs the discovery and gets the callback URI from the selected producer's profile for the given notification type at the NF profile level defaultnotification/subscription parameters. This is the default notification due to the presence of the 3gpp-Sbi-Discovery-notification-type header. apiversion =1

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
non "/" scp-default-sub-notify-uri"	Absent	Absent	Absent	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Absent	Selection of NF is dependent on the discovery response. The path will remain the same as it was when the consumer made their original request. Moreover, non-default notification processing, normal discovery, and NF selection processing is applied. Note: The callbackUri from defaultNotificationSubscriptions section is not considered. It goes with the typical rule match. For more information, see Note at

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
":path"										the bottom of this table if the notification message is without a service name.
non "/" scp-default-sub-notify-uri"	Absent	Present	Present	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Present	SCP does callback URI selection based on 3gpp-Sbi-Routing-Binding header information along with other received header information.
non "/" scp-default-sub-notify-uri"	Absent	Present	Absent	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Present	NF selection is based on 3gpp-Sbi-Routing-Binding header information along with other received header information.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
non "/" scp-default-sub-notify-uri"	Absent	Absent	Present	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Present	SCP does callback URI selection based on 3gpp-Sbi-Routing-Binding header information along with other received header information.
non "/" scp-default-sub-notify-uri"	Absent	Absent	Absent	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Present	NF selection is based on 3gpp-Sbi-Routing-Binding header information along with other received header information.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
"/scp-default-sub-notify-uri"	Present	Present	Present	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	NA	First Attempt: Normal route to the selected producer with the received :path header. Alternate Attempt: SCP performs the discovery and gets the callback URI from the selected producer's profile for the given notification type. SCP modifies the :path header as per the selected notification endpoint.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
"/scp-default-sub-notify-uri"	Present	Present	Absent	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	NA	First Attempt: Normal route to the selected producer with the received :path header. Alternate Attempt: Non-default notification processing , normal discovery, and NF selection processing Note: The callbackUri from defaultNotificationSubscriptions section is not considered. It goes with the typical rule match. For more information, see Note at the bottom of this table if the notification

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
":path"										message is without a service name
"/scp-default-sub-notify-uri"	Present	Absent	Present	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	NA	First Attempt: Normal route to the selected producer with the received :path header. Alternate Attempt: SCP performs the discovery and gets the callback URI from the selected producer's profile for the given notification type. SCP modifies the :path header as per the selected notification endpoint

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
"/scp-default-sub-notify-uri"	Present	Absent	Absent	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	NA	First Attempt: Normal route to the selected producer with the received :path header. Alternate Attempt: Non-default notification processing , normal discovery, and NF selection processing Note: The callbackUri from defaultNotificationSubscriptions section is not considered. It goes with the typical rule match. For more information, see Note at the bottom of this table if the notification

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
										message is without a service name.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
non "/" scp-default-sub-notify-uri"i.e., notification/callback URI	Present	Present	Present	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Absent	Model C Default notification processing with selected notification endpoint. First Attempt: Normal route to the selected producer with the received :path header. Alternate route: SCP performs the discovery and gets the callback URI from the selected consumer's profile for the given notification type and apiversion in the callback header. SCP modifies

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
										the :path header as per the selected notification endpoint.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
non "/" scp-default-sub-notify-uri"i.e., notification/callback URI	Present	Present	Absent	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Absent	Explicit notification processing The first routing is based on the 3gpp-target-apiRoot header. Alternate routing is based on normal discovery and NF selection processing, as applicable. Note: The callbackUri from defaultNotificationSubscriptions section is not considered. It goes with the typical rule match. For more information, see Note at the bottom of this table if the notification message

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
										is without a service name.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
non "/" scp-default-sub-notify-uri"i.e., notification/callback URI	Present	Absent	Present	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Absent	Model C Default notification processing with selected notification endpoint. First Attempt: Normal route to the selected producer with the received :path header. Alternate route: SCP performs the discovery and gets the callback URI from the selected producer's profile for the given notification type at the NF profile level defaultnotification/subscription parameter

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
										s. In the absence of a callback header, apiversion =1 is considered. This is the default notification due to the presence of the 3gpp-Sbi-Discovery-notification-type header. apiversion =1

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
non "/" scp-default-sub-notify-uri"i.e., notification/callback UR	Present	Absent	Absent	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Absent	Non Default notification processing The first routing is based on the 3gpp-target-apiRoot header. Alternate routing is based on normal discovery and NF selection processing , as applicable. Note: The callbackUri from defaultNotificationSubscriptions section is not considered. It goes with the typical rule match. For more information, see Note at the bottom of this table if the notification

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
										message is without a service name.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
non "/" scp-default-sub-notify-uri" i.e., notification/callback URI	Present	Present	Present	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Present	Model C Default notification processing with selected notification endpoint. First Attempt: Normal route to the selected producer with the received :path header. Alternate route: SCP does callback uri selection based on 3gpp-Sbi-Routing-Binding header information along with other received header information. SCP gets the callback URI from the selected consumer

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
										's profile for the given notification type and a version in the callback header. SCP modifies the :path header as per the selected notification endpoint. Note: 1. If servName is present in the 3gpp-sbi-Routing-binding header, then SCP gets the callback URI from the selected consumer's profile for the given notification type for that service name. 2. If servName

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
										is absent in the 3gpp-sbi-Routing-binding header, then SCP gets the callback URI from the selected consumer's profile for the given notification type from service level. If it is not found at the service level, it gets the callback URI from the profile level.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
non "/" scp-default-sub-notify-uri" i.e., notification/callback URI	Present	Present	Absent	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Present	Explicit notification processing The first routing is based on the 3gpp-target-apiRoot header. Alternate routing is based on 3gpp-Sbi-Routing-Binding header information along with other received header information and NF selection processing as applicable. Note: The callbackUri from defaultNotificationSubscriptions section is not considered. It goes with the typical rule match. For more information

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
										n, see Note at the bottom of this table if the notification message is without a service name. Note: Alternate routing is only performed when servName is present in the 3gpp-sbi-Routing-binding header.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
non "/" scp-default-sub-notify-uri" i.e., notification/callback URI	Present	Absent	Present	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Present	Model C Default notification processing with selected notification endpoint. First Attempt: Normal route to the selected producer with the received :path header. Alternate route: SCP does callback uri selection based on 3gpp-Sbi-Routing-Binding header info along with other received header info. SCP gets the callback URI from the selected producer's profile for the given

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
										notification type at the NF profile level defaultnotification/subscription parameters. In the absence of a callback header, apiversion =1 will be considered. This is the default notification due to the presence of the 3gpp-Sbi-Discovery-notification-type header. apiversion =1 Note: If servName is absent in the 3gpp-sbi-Routing-binding header, then SCP gets the callback

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
										URI from the selected consumer's profile for the given notification type from service level. If it is not found at the service level, it gets the callbackURI from the profile level.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
non "/" scp-default-sub-notify-uri" i.e., notification/callback URI	Present	Absent	Absent	Present	Present	Optional (don't care)	Optional (don't care)	Optional (don't care)	Present	Non-default notification processing The first routing is based on the 3GPP-target-apiRoot header. Alternate routing is based on 3gpp-Sbi-Routing-Binding header information along with other received header information and NF selection processing as applicable. Note: The callbackUri from defaultNotificationSubscriptions section is not considered. It goes with the typical rule match.

Table 3-10 (Cont.) Default Notification Routing Behavior

pseudo header	3GPP 5G SBI custom header									Expected behavior at SCP
":path"	3gpp-sbi-target-apiRoot Header	3gpp-Sbi-Callback Header	3gpp-Sbi-Discovery-notification-type(Mandatory) Header	3gpp-Sbi-Discovery-target-nf-type Header	3gpp-Sbi-Discovery-requester-nf-type Header	3gpp-Sbi-Discovery-n1-msg-class Header	3gpp-Sbi-Discovery-n2-info-class Header	3gpp-Sbi-Discovery-* Header	3gpp-sbi-Routing-binding Header	
										For more information, see Note at the bottom of this table if the notification message is without a service name. Note: Alternate Routing is only performed in case when servName is present in 3gpp-sbi-Routing-binding header.

Note

SCP behavior for Notification message without service name is as follows:

- SCP does the rule match or delegated discovery based on available information from the service request.
- SCP selects profile level FQDN for routing.
- SCP uses 80 as the default port for routing.
- SCP uses "http" as the default port for routing.

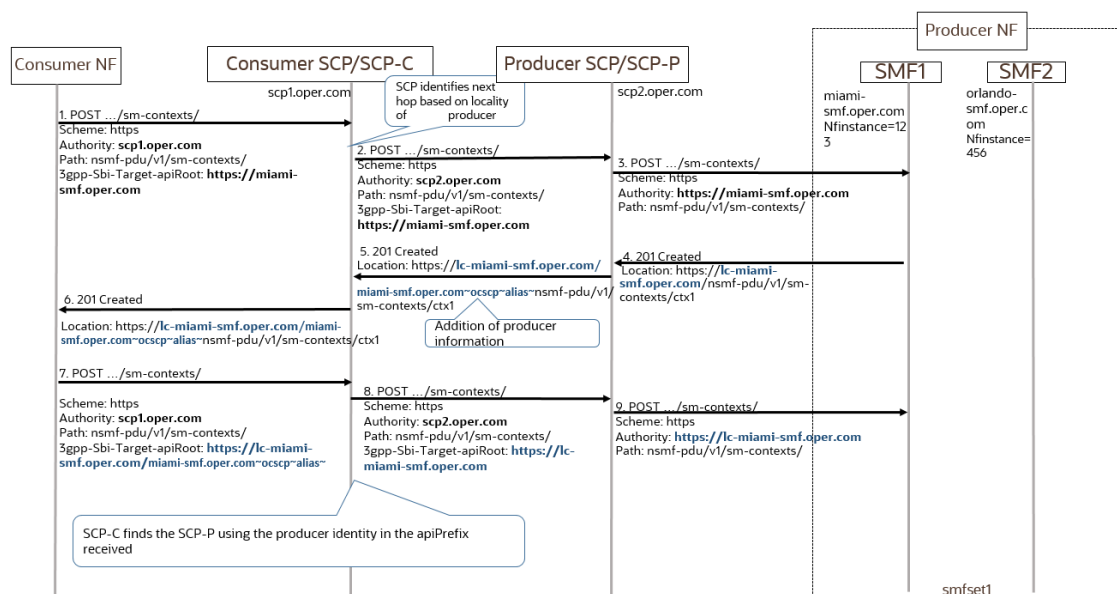
3.14 Location Header for Host Mismatch

According to 3GPP, a consumer NF is expected to use the received "Location" header content while sending subsequent message requests. After receiving a subsequent request, consumer SCP extracts the producer NF details from the API prefix in the "3gpp-Sbi-Target-Apiroot" header and selects an appropriate inter-SCP path to route the request.

SCP matches the authority (FQDN or IP) in "Location" header of the "201 Created" response with the producer's authority (FQDN or IP) where the message request was sent. If the FQDN or IP in the response is different, producer SCP provides the producer's NF details as an API prefix in the "Location" header of the "201 Created" response. If the FQDN or IP is the same, the "Location" header does not perform any modification.

This feature has been introduced to correct inter-SCP routing in the above scenario.

Figure 3-3 Initial and Subsequent Request Routing Supported at SCP



Managing Location Header for Host Mismatch

Enable

By default, this feature is disabled. To enable this feature, you must set the `location_hdr_update_for_host_mismatch` option to true as described in [Configuring SCP Features](#).

Configure

You can configure this feature using REST API and the CNC Console.

- **Configure using CNC Console:** For information about configuring this feature, see [Configuring SCP Features](#) to enable the location header update for host mismatch feature.
- **Configure using REST API:**
Previously, SCP used to support the above scenario through a Helm configuration in `values.yaml` using the `fqdnAliasingEnabled` field. Now a REST API parameter, `location_hdr_update_for_host_mismatch`, is introduced to enable this feature.

For information about configuring REST APIs, see "Configuring SCP Features" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.15 Support for the Client Credentials Assertion Header

The 3GPP introduced Client Credentials Assertion (CCA) enables the NRF or the producer NF to authenticate the consumer NF when using indirect communication through SCP. The CCAs are JSON web tokens and are secured with digital signatures based on JSON Web Signature (JWS). The consumer NF adds the 3gpp-Sbi-Client-Credentials header containing the CCA. The CCA contains the following:

- The NF instance ID of the consumer NF (subject)
- A timestamp (iat) and an expiration time (exp)
- The NF type of the expected audience, such as the NRF type or the NF type of the producer NF

The consumer NF digitally signs the generated CCA based on its private key. The signed CCA contains one of the following fields:

- The X.509 URL (x5u) to refer to a resource for the X.509 public key certificate or certificate chain used for signing the client authentication assertion
- The X.509 Certificate Chain (x5c) includes the X.509 public key certificate or certificate chain used for signing the client authentication assertion

Table 3-11 CCA Contents

Claim	Description
audience (aud)	NF type of the producer NF whose services are requested and/or NRF
expiration time (exp)	Timestamp of CCA expiration
subject (sub)	NF instance ID of consumer NF
timestamp (iat)	Timestamp of CCA generation
Security details used for signing CCA	• x5u - X.509 URL that refers to a resource for the X.509 public key certificate or certificate chain • x5c - X.509 Certificate Chain includes the X.509 public key certificate or certificate chain

The NRF, or producer NF, performs the CCA verification. It verifies that the public key certificate used to sign the CCA contains the same NF instance ID as the consumer's NF instance ID in the CCA. If the verification of the CCA fails at the receiving entity, a "403" response is returned with the cause attribute set to CCA_VERIFICATION_FAILURE.

If some rogue consumer NF acquires a valid CCA from some other consumer NF and uses it as its own CCA, then the CCA validation by the NRF or producer NF creates a security gap. Therefore, additional CCA validations on SCP are needed to mitigate this security impact. The NF instance ID of the NFs in the CCA must match the NF instance ID in the certificate used in the TLS connection by the consumer NF as part of the SCP validations.

Managing Client Credentials Assertion Header

Enable

By default, this feature is disabled. To enable this feature, you must complete the configuration as described in the following configuration section.

Configure

You can configure this feature using REST API and the CNC Console.

- **Configure using CNC Console:** For information about configuring this feature, see:
 - [Configuring SCP Features](#) to enable the CCA Header Validation feature
 - [Configuring Error Profile](#) to edit the error profiles
- **Configure using REST API:** For information about configuring REST APIs, see "Configuring SCP Features" and "Configuring Error Profile" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

The following metrics are added in the [Metrics](#) section:

- `ocscp_worker_cca_validation_failed_total`
- `ocscp_worker_cca_validation_feature_enabled_without_https`
- `ocscp_worker_cca_validation_performed_total`

Added the SCPCcaFeatureEnabledWithoutHttps alert in the [Alerts](#) section.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.16 Support for 5G SBI Message Correlation using the 3gpp-Sbi-Correlation-Info Header

The 3gpp-Sbi-Correlation-Info header contains correlation information, User Equipment (UE) identity that is used by an operator in various offline network management, performance analysis, and troubleshooting tools and applications to identify messages (requests, responses, subscriptions, and notifications) related to a particular subscriber.

In this header, the UE identifier used is Subscription Permanent Identifier (SUPI), and the SUPI type is an International Mobile Subscriber Identity (IMSI). For example, 3gpp-Sbi-Correlation-Info: imsi-345012123123123

SCP efficiently routes message requests, along with the 3gpp-Sbi-Correlation-Info header, to the designated producer NFs. In cases where the correlation header is absent in the response from the target producer NF, SCP forwards the response to the consumer NF while appending the same 3gpp-Sbi-Correlation-Info header received in the request message. Additionally, when the correlation header is present in the request message and SCP generates an error response, SCP includes the correlation header in the response to the consumer NF. This ensures smooth and consistent correlation of 5G SBI messages throughout the system.

SCP adds the correlation header in the following cases if the request from the consumer contains this header:

- Access token requests toward NRF received from Model-C consumer NF.
- Discovery requests or access token requests toward NRF received in service request for delegated discovery in Model-D.
- During alternate routing.
- All the transaction in SCP is forwarded with correlation header toward traffic feed server to save the data in Traffic Feed OCNADD.

Managing Support for 3gpp-Sbi-Correlation-Info Header

Enable

Not applicable

Configure

Not applicable

Observe

There is no specific metrics and alerts required for the Support for 3gpp-Sbi-Correlation-Info Header.

3.17 SCP Health Check API

In the previous release, consumer NFs used to send Service Based Interface (SBI) message requests to SCP without checking the status of SCP because there was no health check mechanism implemented at SCP.

Using the SCP Health Check Application Programming Interface (API) feature, consumer NFs can determine the health status of SCP before forwarding any SBI message request to SCP. This feature allows consumer NFs or SCP-C, which is a consumer SCP, to perform the following health check verification on SCP:

- SCP Reachable: Determines if SCP is reachable through the transport path.
- SCP Available: Determines if SCP is available to provide responses to SBI message requests.

Note

- This feature supports HTTP/2
- Health check fails if either of the above mentioned requirements is not achieved.

SCP provides the RESTful health check service Uniform Resource Identifier (URI) (`{scpapiRoot}/{apiVersion}/status`) to consumer NFs to confirm its availability to process the message requests. Consumer NFs can use the following URI to get the status of SCP:

- Method: OPTIONS
- Resource URIs:
 - `<scheme>://<Scp-worker Fqdn>:<Port>/{apiVersion}/status`
 - `<scheme>://<Scp-worker Fqdn>:<Port>/{apiPrefix}/{apiVersion}/status`

This feature facilitates consumer NFs to determine the health status of SCPs by sending the health query request to SCPs. Based on the health query response, consumer NFs decide whether to send SBI message requests to that SCP or perform alternate routing on other healthy SCP.

SCP uses the following process to provide its health status to a consumer NF:

1. When SCP receives a health query request, it calculates its overall average load by considering the load of each SCP-Worker instance present in that SCP.
2. SCP compares the overall average load with the configured values of the `avgScpLoadThresholdValue` REST API parameter as described in "Configuring SCP Features" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
3. If the overall average load value is more than the value of `avgScpLoadThresholdValue`, SCP is considered as unhealthy and SCP sends error responses to the consumer NF based on the `healthCheckErrorProfile` parameter as described in "Configuring Error Profiles" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
4. If the overall average load value is less than or equal to the value of `avgScpLoadThresholdValue`, SCP is considered as healthy and SCP sends one of the following responses to the consumer NF based on the configuration:
 - **200WithPayload:** Sends 200 OK response code with the overall average load information of SCP.
 - **200WithEmptyPayload:** Sends 200 OK response code without the overall average load information of SCP.
 - **204WithNoContent:** Send only 204 response code with no other information.

Inter-SCP Health Check

In inter-SCP health check scenarios, a consumer SCP-C performs health check query on peer SCP instances before sending any SBI message request.

When a consumer SCP-C receives a health query request from a consumer NF, it checks for the availability of a healthy SCP-P to which the SBI message request can be forwarded.

The following procedure describes how a consumer SCP-C checks the status of peer producer SCP-P instances:

1. SCPC-Notification sends the `InterScpList` to SCPC-Audit through REST APIs if the `isScpHealthCheckSvcEnabled` REST API parameter is enabled. For more information about this parameter, see "Configuring SCP Features" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
2. SCPC-Configuration sends the configuration information, such as `isScpHealthCheckSvcEnabled`, time interval to trigger periodic health query polls, number

of consecutive positive responses, number of negative responses, and so on, to SCPC-Audit.

3. SCPC-Audit periodically triggers health check query polls to SCP-P through SCP-Worker of SCP-C with 3gppapiroot: scp-pfqdn+ruriPath of health query if the isScpHealthCheckSvcEnabled REST API parameter is enabled.
4. SCP-Worker forward the health query request received from SCPC-Audit toward SCP-P.
5. SCP-P responds with its health query status.
6. SCPC-Audit obtains the response for the health query request from SCP-P through SCP-Worker.
7. SCPC-Configuration sends the configuration information, such as isScpHealthCheckSvcEnabled, time interval to trigger periodic health query polls, number of consecutive positive responses, number of negative responses, and so on, to SCP-Worker.
8. SCPC-Audit sends the health status of the inter-SCPs to SCP-Worker for the inter-SCP messages processing.

Managing SCP Health Check API

Enable

By default, this feature is enabled at the ingress side. However, in case of inter-SCP scenarios, this feature is disabled by default. To enable this feature, you must set the health_check option to true as described in [Configuring SCP Features](#).

Configure

You can configure this feature using REST API and the CNC Console.

- **Configure using CNC Console:** For information about CNC Console configurations, see [Configuring SCP Features](#).
- **Configure using REST API:** For information about configuring REST APIs, see "Configuring SCP Features" and "Configuring Error Profiles" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

The following metrics are added in the [Metrics](#) section:

- ocscp_worker_health_check_rx_req_total
- ocscp_worker_health_check_success_res_total
- ocscp_worker_health_check_error_res_total
- ocscp_interscp_health_check_status_success_total
- ocscp_interscp_health_check_status_failure_total
- ocscp_health_check_status_failed
- ocscp_interscp_health_check_status_failed
- ocscp_audit_interscp_health_check_tx_req_total
- ocscp_audit_interscp_health_check_rx_res_total

The following alerts are added in the [Alerts](#) section:

- [SCPHealthCheckFailedForPeerSCP](#)
- [SCPHealthCheckFailed](#)

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.18 Verbose Logging for SCP

SCP warn and error level logs are enhanced to display more information to ease the troubleshooting process for various issues in the production environment.

The Verbose Logging for SCP feature defines a common log format to include required information to enhance debugging process for WARN and ERROR logs. Using the following log format, ERROR and WARN level logs are modified to standardize the log format, add more details, and enhance readability:

The following is the log level format for SCP-Worker and SCP-nrfproxy microservices:

- WARN: Category: <<category>>, Event: <<Event>>, Details: <<Message_Details>>
- ERROR: Category: <<category>>, Event: <<Event>>, Details: <<Message_Details>>, Exception: {Cause:<<Exception_cause>>, Message: <<Exception_Message>>, StackTrace: <<Exception_Stacktrace>>}

The following is the log level format for SCP-notification microservice:

- WARN: Category: <<category>>, Event: <<Event>>, EventID: <<App-Event-ID>>, Details: <<Message_Details>>
- ERROR: Category: <<category>>, Event: <<Event>>, EventID: <<App-Event-ID>>, Details: <<Message_Details>>, Exception :{Cause:<<Exception_cause>>, Message: <<Exception_Message>>, StackTrace: <<Exception_Stacktrace>>}

Note

- This feature supports SCP-Worker and SCP-nrfproxy microservice with WARN and ERROR level logs.
- SCP is deployed with WARN level log in the production environment. This is applicable for SCP-Worker and SCP-nrfproxy microservice.
- SCP is deployed with INFO level log in the production environment. This is applicable for SCP-notification microservice.

The SCP log format is made verbose to follow a common format for WARN and ERROR log levels to determine and investigate issues in various events as described in the following table:

Table 3-12 Log Attributes

Log Attribute	Description
Category	Determines the category of log messages.

Table 3-12 (Cont.) Log Attributes

Log Attribute	Description
Event	Represents logged message event type.
EventID	Unique ID to identify associated category and sub category for logged message. EventID format: OSCP-XXX-YY-ZZZ, where - XXX: Represents Micro service codes - YY: Represents Category - ZZZ: Represents Event TYPE Note: This log attribute is only applicable for SCP-notification microservice.
Message_Details	Represents the actual message defining the error.
Exception_cause	Returns the cause of an error.
Exception_Message	Represents actual exception message.
Exception_Stacktrace	Contains complete stack trace which is causing the error.

The new log format provides comprehensive information about an issue comprising issue category, event name, and detailed description of the issue to help you identify and resolve errors.

The following table describes log levels for various event types:

Table 3-13 Log Levels for SCP-Worker

Category	Event	Log Level	Description
Up Stream	Connection Timeout	ERROR	This event is raised when the connection fails due to connect timeout expiry.
	Request Timeout	WARN	This event is raised when message request processing fails due to the message response not being received within the configured time.
	Response Timeout	WARN	This event is raised when the destination client response is timeout.
	RetryCount Exhausted	WARN	This event is raised when message request routing fails because all retry attempts are exhausted. All retry attempts lead to unsuccessful responses.
Overload	Pod OverLoad	WARN	This event specifies the transition of pod overload state from one state to another.
	Pending Transaction Overload	ERROR	This event is raised when the pending transaction count reaches the configured threshold level.
Health Check	Readiness Failure	ERROR	This event is raised when cache members or routing options are unavailable or the rule engine is down.
	Thread Status	ERROR	This event is raised when thread watchdog detects a thread is stuck.
Routing	Destination Search Exhausted	WARN	This event is raised when no further valid destination is found for routing, all eligible producer NFs are attempted.
	Rule Not Found	WARN	This event is raised when routing rules are not found for the selected producer NFs to route messages.

Table 3-13 (Cont.) Log Levels for SCP-Worker

Category	Event	Log Level	Description
	Invalid API Version in Request	WARN	This event is raised when no matching rule is found for the API version received in request.
	Release16 Validation Failed	WARN	This event is raised when any Rel16 message validation fails.
	Message Loop Detected	WARN	This event is raised when SCP detects a message loop for a request if it has a VIA header containing SCP's self FQDN.
	Destination Congested	WARN	This event is raised when SCP detects that the selected destination is congested.
	Destination Connection Failure	WARN	This event is raised when the upstream connection fails.
	Invalid or Empty Details	ERROR	This event is raised when any routing related data is empty or contains invalid data.
	Miscellaneous	WARN	This event is raised when any warning is raised due to miscellaneous reasons, such as failure in inter-plmn rule creation, handling routing failure responses, and so on.
Internal	Internal Error	ERROR	This event is raised when an internal error, such as monitoring of Kubernetes resources, failure to load or validate SSL certificates, object invocation, pegging of metrics, and so on, occur.
	Conversion Failure	ERROR	This event is raised due to unsupported formats, type cast, object formation, and parsing failures.
	Initialization Failed	ERROR	This event is raised when any class initialization fails, such as Jetty client and Kafka admin client creation.
	Miscellaneous	WARN	This event is raised when any failure in retrieval or updating of values to an object or due to unexpected event method execution.
	SSE Failure	ERROR	This event is raised due to connectivity issue with the SCPC-Configuration or SCPC-Notification microservice that impacts data retrieval.
	Config Data Parse	ERROR	This event is raised when parsing failed for the received config data.
Data Sync	Rule Engine	ERROR	This event is raised when routing rules fail to synchronize and update routing rules from the SCPC-Notification microservice.
	Keep Alive	WARN	This informational event is raised when the received Keep Alive and Config connectivity are up.
	Invalid Data	WARN	This event is raised when any incorrect configuration data is received due to missing parameters in any configuration.
	Config Update	WARN	This event is raised when any failure occurs while fetching or updating configuration values received from the SCPC-Configuration microservice.
Cache	Cache Initialization	ERROR	This event is raised when the initialization of the coherence cache fails.

Table 3-13 (Cont.) Log Levels for SCP-Worker

Category	Event	Log Level	Description
	Cache Read Write Failure	WARN	This event is raised when any read or write operation fails for the cache.
	Invalid or Empty Details	WARN	This event is raised when any incorrect or empty data from the cache is received.
Message Validation	CCA Header Validation Failed	WARN	This event is raised when the validation for the CCA header fails.

Table 3-14 Log Levels for SCP-nrfproxy

Category	Event	Log Level	Description
Up Stream	Connection Timeout	ERROR	This event is raised when the connection fails due to connect timeout expiry.
	Request Timeout	WARN	This event is raised when request processing fails due to response not received within configured time.
	Response Timeout	WARN	This event is raised when the destination client response is timeout.
	RetryCount Exhausted	WARN	This event is raised when message request routing fails because all retry attempts are exhausted. All retry attempts lead to unsuccessful responses.
Health Check	Readiness Failure	ERROR	This event is raised when cache members or routing options are unavailable or the rule engine is down.
	Thread Status	ERROR	This event is raised when thread watchdog detects a thread is stuck.
Routing	Destination Search Exhausted	WARN	This event is raised when no further valid destination is found for routing, all eligible producer NFs are attempted.
	Rule Not Found	WARN	This event is raised when routing rules are not found for the selected producer NFs to route messages.
	Destination webclient Connection Failure	WARN	This event is raised when upstream connection establishment fails.
	Invalid or Empty Details	ERROR	This event is raised when any routing related data is empty or contains invalid data.
	Miscellaneous	WARN	This event is raised when any warning is raised due to miscellaneous reasons, such as failure in inter-plmn rule creation, handling routing failure responses, and so on.
Internal	Internal Error	ERROR	This event is raised when an internal error occurs, such as a failure to read the application.properties file or a failure in the registration of NRF-specific details.
	Conversion Failure	ERROR	This event is raised due to unsupported formats, type cast, object formation, and parsing failures.
	Initialization Failed	ERROR	This event is raised when any class initialization fails, such as Jetty client and Kafka admin client creation.

Table 3-14 (Cont.) Log Levels for SCP-nrfproxy

Category	Event	Log Level	Description
	Miscellaneous	WARN	This event is raised when any failure in retrieval or updating of values to an object or due to unexpected event method execution.
Data Sync	SSE Failure	ERROR	This event is raised due to connectivity issue with the SCPC-Configuration or SCPC-Notification microservice that impacts data retrieval.
	Config Data Parse	ERROR	This event is raised when parsing failed for the received config data.
	Rule Engine	ERROR	This event is raised when routing rules fail to synchronize and update routing rules from the SCPC-Notification microservice.
	Keep Alive	WARN	This informational event is raised when the received Keep Alive and Config connectivity are up.
	Invalid Data	WARN	This event is raised when any incorrect configuration data is received due to missing parameters in any configuration.
	Config Update	WARN	This event is raised when any failure occurs while fetching or updating configuration values received from the SCPC-Configuration microservice.

Table 3-15 Log Levels for SCP-notification

Category	Event	EventID	Description
READINESS_FAILURE	DB_RECORDS_NOT_AVAILABLE	OSCP-NTF-RDYNES-EV001	This event is raised when default database records for the following are not created: - Canary Release - SCP Profile - NRF Details - Routing Options - System Options
	REQUIRED_DATA_NOT_AVAILABLE	OSCP-NTF-RDYNES-EV002	This event is raised if the data is not available, causing the health check to fail.
LIVENESS_FAILURE	THREAD_WATCHDOG_FAILURE	OSCP-NTF-LIVNES-EV001	This event is raised whenever any threads become stuck.
	QUEUE_FULL	OSCP-NTF-LIVNES-EV002	This event is raised whenever the liveness probing condition fails, resulting in application restarts. The probing condition is that the notification queue is full.
NF_RULE_CREATE_OR_UPDATE_FAILURE	NF_PROFILE_VALIDATION_FAILURE	WARN	This event is raised when NF profile validation fails due to the following reasons: - Invalid HOST - Invalid serviceName - Invalid IpAddress - Invalid NfServiceFqdn - Duplicate IpEndPoints/ServiceInstanceIDs

Table 3-15 (Cont.) Log Levels for SCP-notification

Category	Event	EventID	Description
	RULE_PROCESSING_FAILURE	WARN	This event is raised when rule creation fails due to an exception occurring, specifically when updating the destination rule fails.
	NF_PROFILE_REFINE_ERROR	WARN	This event is raised when: - IOException or ApiException occurs during data refinement in NFService - An exception is encountered due to incorrect configuration for the profile while creating a refined version of NFProfile.
	DB_UPDATE_FAILURE	ERROR	This event is raised when saving data to the database fails.
	DB_READ_FAILURE	OSCP-NTF-RULUPD-EV005	This event is raised when reading data from the database fails.
	DB_DELETE_FAILURE	OSCP-NTF-RULUPD-EV006	This event is raised when a database deletion operation fails.
	RULE_CREATE_OR_UPDATE_MISCELLANEOUS	OSCP-NTF-RULUPD-EV007	This event is raised when rule creation fails due to miscellaneous reasons.
	RULE_CREATE_OR_UPDATE_ABORTED	OSCP-NTF-RULUPD-EV008	This event is raised when rule creation is aborted due to failures.
NF_RULE_DELETION_FAILURE	RULE_DELETE_PROCESSING_FAILURE	OSCP-NTF-RULDEL-EV001	This event is raised when rule deletion fails due to certain errors or failures.
	RULE_DELETE_ALREADY_DEREGISTERED	OSCP-NTF-RULDEL-EV002	This event is raised when a deregistration request is received for an NF that has already been deregistered.
	RULE_DELETE_ABORTED	OSCP-NTF-RULDEL-EV003	This event is raised when the deletion of a rule is aborted due to miscellaneous reasons.
	RULE_DELETE_DB_READ_FAILURE	OSCP-NTF-RULDEL-EV004	This event is raised when there is a failure to delete a rule while reading from the database.
	RULE_DELETE_DB_DELETE_FAILURE	OSCP-NTF-RULDEL-EV005	This event is raised when there is a failure to delete a rule from the database.
NF_RULE_PROCESSOR_FAILURE	RULE_PROCESSOR_MISCELLANEOUS	OSCP-NTF-RULPRC-EV001	This event is raised when rule processing fails due to miscellaneous reasons.
	RULE_PROC_DB_READ_FAILURE	OSCP-NTF-RULPRC-EV002	This event is raised when rule processing fails due to a database read failure.
	RULE_PROC_DB_DELETE_FAILURE	OSCP-NTF-RULPRC-EV003	This event is raised when rule processing fails due to a database deletion failure.
	RULE_PROC_ABORTED	OSCP-NTF-RULPRC-EV004	This event is raised when rule processing fails due to receiving a notification from NRF for an NF configured as local.
INTERNAL	INTERNAL_ERROR	OSCP-NTF-INTRNL-EV001	This event is raised whenever there is an internal error such as monitoring of Kubernetes resources, object invocation, or pegging of metrics.
	CONVERSION_FAILURE	OSCP-NTF-INTRNL-EV002	This event is raised whenever there are failures in unsupported format/type casting, object formation, or parsing.
	INITIALIZATION_FAILURE	OSCP-NTF-INTRNL-EV003	This event is raised when there is a failure during class initialization.

Table 3-15 (Cont.) Log Levels for SCP-notification

Category	Event	EventID	Description
	MISCELLANEOUS	OSCP-NTF-INTRNL-EV004	This event is raised whenever there are failures in retrieving or updating values to an object, or during unexpected method execution.
	DB_READ_FAILURE_AT_STARTUP	OSCP-NTF-INTRNL-EV005	This event is raised whenever there is a database read failure during application startup.
	INTERNAL_DATA_NOT_FOUND	OSCP-NTF-INTRNL-EV006	This event is raised whenever required internal data is missing.
DATA_SYNC	CONFIG_DATA_PARSE_FAILURE	OSCP-NTF-DATSYN-EV001	This event is raised when parsing fails for received config data.
	CONFIG_UPDATE_FAILURE	OSCP-NTF-DATSYN-EV002	This event is raised whenever there is a failure while fetching or updating configuration values received from the Config MS
	DATA_SYNC_DB_READ_FAILURE	OSCP-NTF-DATSYN-EV003	This event is raised when reading the data from DB fails in SSE communications.
	CONFIG_DATA_NOT_FOUND	OSCP-NTF-DATSYN-EV004	This event is raised when the configured data is not found in DB sync.
	INVALID_DATA	OSCP-NTF-DATSYN-EV005	This event is raised when the configured data is not found during database synchronization.
DBSNAPSHOT_UPDATER	DB_SNAPSHOT_READ_FAILURE	OSCP-NTF-DBSNSEV001	This event is raised whenever an exception occurs while reading virtual services or serviceEntries from the database.

Managing Verbose Logging

Enable

This feature is enabled automatically at the time of SCP deployment.

Configure

There is no specific configuration required for this feature.

Support for the Trace log level has been introduced for the scp-notification microservice. For more information, please see the [Configuring Global Options](#) and the "Configuring Dynamic Logging" section in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

There is no specific metrics and alerts required for this feature.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

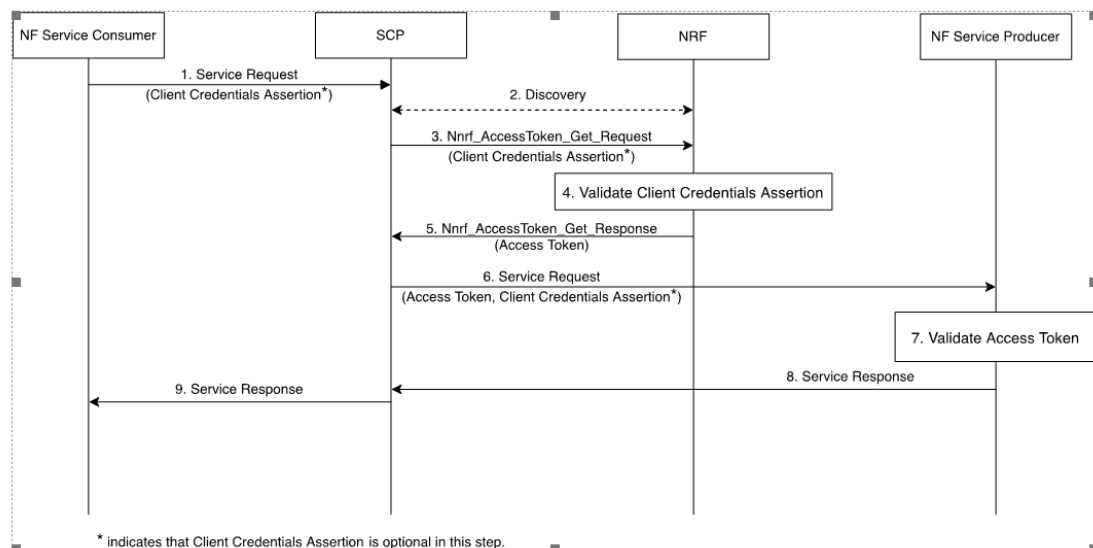
3.19 Support for OAuth2.0

Access tokens are credentials used by consumer NFs to access protected resources provided by producer NFs. These credentials are issued by NRF. These tokens represent specific scope and duration of access. SCP supports OAuth2.0 (Open Authorization) access tokens in 5G indirect communication models such as Model C and Model D.

In the 5G SBI Model D indirect communication (with delegated discovery) mode, SCP obtains access tokens from NRF for 5G SBI message requests except notification or callback messages. SCP uses the acquired access tokens to forward SBI message requests to the selected producer NFs.

The following call flow represents how SCP obtains access tokens from NRF:

Figure 3-4 SCP Obtains Access Tokens from NRF



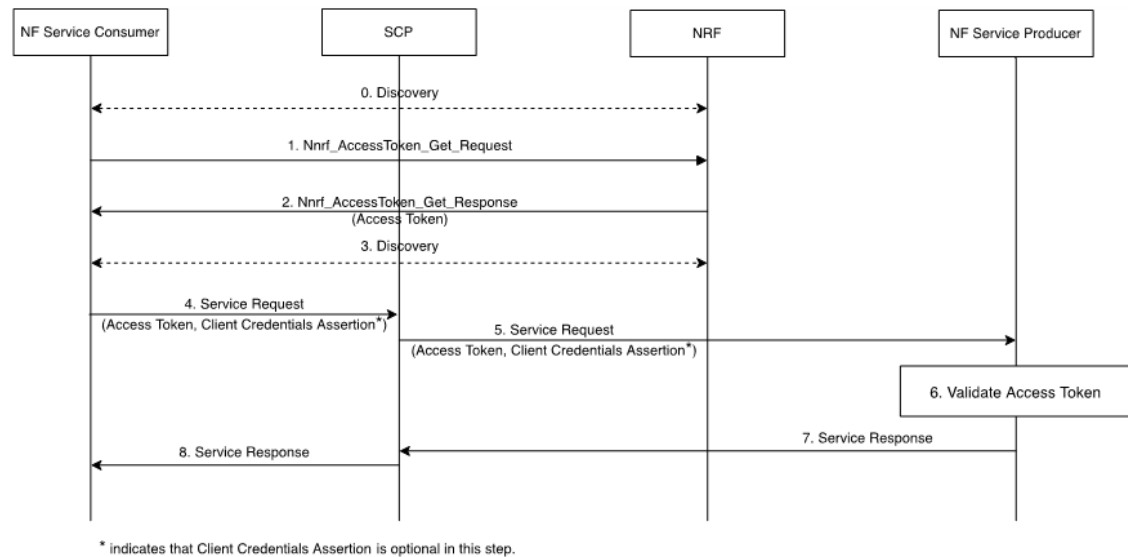
The above mentioned call flow describes the process to use the Discovery parameter to send a request to NRF for obtaining the access token:

1. NF Service Consumer sends a service request to SCP.
2. If SCP does not get an access token in the service request, it uses the Discovery parameter to send the access token request to NRF.
3. SCP sends Nnrf_AccessToken_Get_Request to NRF.
4. NRF validates the access token request using Client Credential Assertion (CCA).
5. NRF sends Nnrf_AccessToken_Get_Response to SCP.
6. SCP forwards the service request, along with CCA and access token, to NF Service Producer.
7. NF Service Producer validates the access token.
8. NF Service Producer sends the service response to SCP.
9. SCP relays the service response to NF Service Consumer.

In the 5G SBI Model C indirect communication (without delegated discovery) mode, consumer NFs obtain access tokens from NRF except notification or callback messages. SCP uses the access tokens received from consumer NFs and forward SBI message requests to the selected producer NFs.

The following call flow represents how NF Service Consumer obtains access tokens from NRF:

Figure 3-5 NF Consumer Obtains Access Tokens from NRF



The above mentioned call flow describes the process that NF Service Consumer uses to obtain access token from NRF and how SCP uses this access token to forward the service request.

1. When there is no mutual authentication between NF Service Consumer and NRF, NF Service Consumer sends Nnrf_AccessToken_Get_Request to NRF.
2. NRF sends Nnrf_AccessToken_Get_Response to NF Service Consumer.
3. NF Service Consumer sends Discovery service requests to NRF.
4. After receiving the discovery response with NF profiles, NF Service Consumer sends the service request to SCP.
5. SCP uses the access token received from NF Service Consumer and forwards the service request, along with CCA and access token, to NF Service Producer.
6. NF Service Producer validates the access token.
7. NF Service Producer sends the service response to SCP.
8. SCP relays the service response to NF Service Consumer.

Note

SCP supports 5G SBI Client Credentials Assertion (CCA) custom header for Model C and Model D indirect communication routing. It is an optional header that can be sent by consumer NFs, along with the service request for enabling consumer NFs authentication. For more information about CCA header, see [Support for the Client Credentials Assertion Header](#).

This feature enables SCP to perform the following tasks:

- SCP determines the status of OAuth2 support of producer NFs deployed in the network using the following:
 - Learnt 5G NF service topology from NRF
 - NF discovery response during delegated service request
 - Configurations at SCP
- In Model D indirect communication, SCP is responsible for access token lifecycle (initiate, maintain, and delete). SCP caches the access tokens and reuses them during initial, subsequent, or alternate routing.
- SCP validates access tokens as per the configuration.
- SCP returns the access token received from NRF to consumer NFs using the "3gpp-Sbi-Access-Token" header.
- SCP determines service level scope information in the access token from the "3gpp-Sbi-Access-Scope" header
- SCP regulates and avoids duplicate access token requests for the same consumer NF and producer NF pair.
- SCP supports OAuth2.0 in inter-SCP scenarios.
- SCP proactively attempts to refresh the access tokens when required.
- SCP supports NF type and NF instance type of access tokens.

SCP has introduced the SCP-nrfProxy-Oauth microservice to manage requests initiated to NRF for its nnrf-oauth2 service that considers various parameters as body and CCA header to provide an authentication token as response to the consumer NF. This microservice interacts with SCP-Worker when it receives request from SCP-Worker. When SCP-Worker initiates the access token request to NRF, Oauth2.0 service sends a request to SCP-Worker with NRF details in the routing binding header. Coherence cache is integrated with this service where the values of tokens are stored based on the cache configuration at SCP. To avoid duplicate requests to NRF, an internal cache is also maintained. This service is capable of handling various states of access tokens received, such as expiration, and acts accordingly.

Managing Support for OAuth2.0

Enable

By default, this feature is disabled. To enable this feature, do the following tasks:

1. Set the nrfProxyOauthService Helm parameter to true in the `ocscp_values.yaml` file.
2. Set the `oauth2_support` option to true as described in [Configuring SCP Features](#).

Configure

You can configure this feature using Helm, REST APIs, and the CNC Console.

- **Configure using CNC Console:** For information about CNC Console configurations, see [Configuring SCP Features](#) and [OAuth2.0 Configurations](#).
- **Configure using REST API:** For information about configuring REST APIs, see "Configuring SCP Features", "Configuring NRF", "Configuring OAuth2.0 Access Token Granularity", and "Configuring OAuth2.0 Local PLMN Required" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
- **Configure using Helm:** For information about configuring the SCP-nrfProxy-Oauth microservice, see "SCP-nrfProxy-oauth Parameters" in *Oracle Communications Cloud*

Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide.

Observe

The following metrics are added in the [Metrics](#) section:

- ocscp_oauth_access_token_cache_match_total
- ocscp_worker_access_token_forwarded_in_tx_res_total
- ocscp_worker_access_token_forwarded_in_tx_req_total
- ocscp_worker_access_token_cache_match_total
- ocscp_nrf_oauth_error_counts_total
- ocscp_oauth_access_token_tx_rsp_total
- ocscp_oauth_access_token_rx_res_total
- ocscp_oauth_access_token_tx_req_total
- ocscp_oauth_access_token_rx_req_total
- ocscp_worker_access_token_rx_res_total
- ocscp_worker_access_token_tx_req_total

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.20 Pod Overload Control

SCP processes 5G Service Based Interface (SBI) messages using its microservices. To protect scp-worker pods from the overload, SCP has introduced the Pod Overload Control feature.

This feature throttles only ingress 5G SBI message requests based on:

- CPU utilization of the scp-worker microservice pods.
- Pending Transactions for which responses from producer NFs are awaited by SCP.

SCP provides the following pod overload configurations for scp-worker:

- Pod Overload Action Policy: To configure action to be taken by scp-worker pods for CPU overload.
- Pod Overload Control Policy: To configure CPU and pending transactions onset threshold, abatement threshold, and abatement time for scp-worker.
- Pod Overload Discard Policy: To configure the percentage of messages to be discarded based on the discardPercentage parameter or discard messages based on the sbiMsgPriorityDiscardFrom parameter.

For more information about the aforementioned configurations, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

When CPU utilization or pending transactions of the scp-worker pod exceeds the configured threshold level, SCP responds based on the configured overload Action Policy as described in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Message Priority

SCP assigns and updates the "3gpp-Sbi-Message-Priority" header to prioritize ingress 5G SBI messages during the overload control task. For more information about message priority, see [Message Priority Assignment and Override](#).

Managing Pod Overload Control

Enable

This feature is enabled automatically in SCP at the time of deployment.

Configure

You can configure the Pod Overload Control feature using the CNC Console or REST API.

- **Configure using CNC Console:** For information about 5G SBI message priority and pod overload policies configurations, see [Configuring Pod Overload Control Policies](#) and [Configuring SBI Message Priority](#).
- **Configure using REST API:** For information about 5G SBI message priority and pod overload policies configurations, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about metrics, see [Metrics](#).

The following alerts are added in the [Alerts](#) section:

- [SCPCPUUsageFileReadFailed](#)
- [SCPGroupVersionDetectionFailed](#)

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.21 Host Preference for Egress Message Requests

Although Fully Qualified Domain Name (FQDN) is present in the ingress message requests, SCP uses IP as the host in the egress message request if IP is present in the learned Network Function (NF) profile of the producer NF.

Using the Host Preference for Egress Message Requests feature, SCP can determine the host type to be included in the egress messages. SCP uses one of the host types such as FQDN or IP, in the egress message request.

The host type in the egress message request is determined by the following:

- 3gpp-Sbi-Target-apiRoot header
- hostPreference of egress_host_preference as described in the "Configuring SCP Features" section of *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*
- Learned NF profile from NRF

If the 3gpp-Sbi-Target-apiRoot header is present in the ingress message request, the host type is determined by the `hostPreference.request.apiRootHdrPresent` REST API parameter. If this parameter value is set to `passThrough`, which is the default value, SCP uses the same type of host in the egress message request as received through the ingress message request. For more information about the `hostPreference.request.apiRootHdrPresent` REST API parameter, see "Configuring SCP Features" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

If the 3gpp-Sbi-Target-Apiroot header is absent in the ingress message request, SCP uses the `hostPreference.request.apiRootHdrNotPresent` REST API configuration to determine the host type. By default, SCP uses IP as the host type.

In case of alternate routing, SCP determines the host type of another equivalent producer NF by using the same method as used for the target producer NF.

If the selected host type is FQDN, SCP provides the `fqdnResolution.fqdn` REST API parameter to choose whether to resolve the FQDN in the egress message request with the IPs (from IP Endpoints) present in learned NF profiles of the target producer NF or trigger a Domain Name System (DNS) query.

SCP tries to get the same host type as configured in the `hostPreference` REST API parameter, however, if the corresponding information such as FQDN or IP, is unavailable, then SCP uses other available information for routing.

Managing Host Preference for Egress Message Requests

Enable

By default, this feature is disabled. To enable this feature, you must complete the configuration as described in the following configuration section.

Configure

You can configure this feature using REST API and the CNC Console.

- **Configure using CNC Console:** For information about CNC Console configurations, see [Configuring SCP Features](#).
- **Configure using REST API:** For information about configuring REST APIs, see "Configuring SCP Features" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

The following metrics are added in the [Metrics](#) section:

- `ocscp_worker_fqdn_resolution_total`
- `ocscp_worker_interplmn_fqdn_resolution_total`

- `ocscp_worker_http_req_host_type_total`

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.22 Enhanced 5G SBI Message Failure Handling

SCP adds server header to all SCP generated error responses as defined in 3GPP TS 29.500. SCP forwards error responses received from producer NFs to consumer NFs without modifying the server header. In case of alternate routing, SCP forwards the last received error response to the consumer NF.

In SCP 1.15.0, the 5G SBI Message Failure Handling functionality is enhanced to enable SCP to send information of all the routing attempts made on producer NFs by SCP. The server header format is enhanced to include all the routing information while forwarding error responses with server header to consumer NFs.

The server header format includes information of all the attempted producer NFs in the reverse order of attempts made. For example: If NF1 and NF2 are attempted in that order, the server header formats are:

- In forwarded responses: "Server: <NF2 Type>-<NF2 instance id> <NF1 Type>-<NF1 instance id>"
- In self-generated responses: "Server: SCP-<scp-fqdn> <NF2 Type>-<NF2 instance id> <NF1 Type>-<NF1 instance id>"

In SCP 23.1.0, when SCP does not receive a server header in an error response from a producer NF, SCP adds a server header with its FQDN while forwarding the response to the consumer NF.

SCP generates error responses in the following scenarios:

Note

The server header, using SCP's own FQDN, will apply to all error responses generated by SCP. This includes errors caused by discards due to ingress or egress rate limiting, pod overload control, destination exhausted, no host, and so on. For more information about error codes generated by SCP, see [Error Code Generated by SCP](#).

- **Connection issue with producer NF:** When the producer NF is unhealthy or SCP is unable to establish connection with the producer NF.
- **Timeout at SCP:** When the producer NF fails to reply to SCP within the specified time as configured in the `responseTimeout` REST API parameter or if the configured `totalTransactionLifetime` expires.
- **Service unavailable:** When the connection is unavailable between SCP and producer NF.

Note

If an IP address is provided in the SCP FQDN deployment field, the same is mentioned in the server header introduced by SCP for the above mentioned scenarios.

SCP adds a server header with its FQDN in the error responses received from the service mesh sidecar. The server header format is: "Server": "SCP-<SCP FQDN>".

SCP forwards the server header received in the error response from the producer NF as it is to the consumer NF.

When SCP receives an error response from sidecar or service mesh, then SCP adds a server header with its FQDN and relays them to the consumer NF. SCP detects an error generated by sidecar or service mesh failure when the received response has server header matching one of the configured server header strings in the `sideCarProxyServerHeader` REST API parameter. For example, "envoy", "istio-envoy", and custom string configured in SCP, and the status code in the received error response matches the status code configured in the `sideCarProxyStatusCode` REST API parameter.

SCP appends its FQDN in the server header if it has received a response from at least one producer NF (also known as the last error response), and SCP could not try other NFs due to the following:

- Connection issue, unknown host exception or unknown error
- Service mesh reported errors
- Timeout at SCP
- Received error response from the producer NF without server header

The error response for the above scenarios has the server header as follows: "Server": "SCP-<SCP Fqdn> NFTYPE-<NfInstanceId>".

In SCP 23.2.0, the 5G SBI Message Failure Handling functionality is enhanced. SCP forwards the last or latest error response received from producer NFs to consumer NFs, and server header content is populated with the details of the producer NF from which the error response is received.

SCP does not append SCP FQDN in returned server header to downstream if it has received an error response from at least one producer and SCP could not try other producer NFs due to the following:

- Connection issue, unknown host exception or unknown error
- Service mesh reported errors
- Timeout at SCP
- Received error response from the producer NF without server header

The error response for the above scenarios has the server header as follows: "Server": "NFTYPE-<NfInstanceId>".

The following table describes differences between the server header behavior in SCP 23.1.0 and SCP 23.2.0:

Table 3-16 Implemented Behavior

Scenario	Server Header in SCP 23.1.0	Server Header in SCP 23.2.0
When SCP sends a message request to UDM1, UDM1 sends an error response with a server header. Then, SCP performs alternate routing on UDM2, and a timeout occurs at SCP.	"Server": "SCP-<scp-fqdn> UDM-udm1-instance-id"	"Server": "UDM-udm1-instance-id"
When SCP sends a message request to UDM1, SCP receives an error response from the service mesh or sidecar. Then, SCP performs alternate routing on UDM2 and receives an error response without a server header, and when SCP performs another alternate routing on UDM3, it receives an error response with a server header.	"Server": "SCP-<scp-fqdn> UDM-udm3-instance-id"	"Server": "UDM-udm3-instance-id"

The following table describes the server header behavior in different error scenarios between SCP and producer NFs.

Note

In all the scenarios, SCP attempts to establish connections with producer NFs (UDM1, UDM2, and UDM3) as mentioned in the scenarios.

Example Scenario 19: UDM1 sends an error response with server header. Timeout has occurred at UDM2, and UDM3 has connection issue, and the pod is down.

In the above mentioned scenario and as specified in the note, SCP attempts to establish connection with UDM1 first followed by UDM2, and then UDM3.

Table 3-17 Server Header Behavior

Serial No.	Scenario	Server Header Behavior
1	UDM1 and UDM2 services are unavailable.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: – true: "Server" : "SCP-<SCP FQDN>" – false: "Server" : "SCP-<SCP FQDN>" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "SCP-<SCP FQDN>" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"
2	Timeout has occurred at UDM1 and UDM2.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: – true: "Server" : "SCP-<SCP FQDN>" – false: "Server" : "SCP-<SCP FQDN>" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "SCP-<SCP FQDN> UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"

Table 3-17 (Cont.) Server Header Behavior

Serial No.	Scenario	Server Header Behavior
3	UDM1 and UDM2 have generated error responses without server header.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "UDM-udm2-instanceid" — false: SCP won't add any server header - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"
4	UDM1 and UDM2 have generated error responses with server header.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "UDM-udm2-instanceid" — false: "Server" : "UDM-udm2-instanceid" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid"
5	UDM1 sends an error response without server header and timeout has occurred at UDM2.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "UDM-udm1-instanceid" — false: SCP won't add any server header - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"
6	UDM1 sends an error response with server header and timeout has occurred at UDM2.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "UDM-udm1-instanceid" — false: "Server" : "UDM-udm1-instanceid" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "UDM-udm1-instanceid"
7	UDM1 and UDM2 send error responses without server header.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "UDM-udm2-instanceid" — false: SCP won't add any server header - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"
8	UDM1 and UDM2 send error responses with server header.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "UDM-udm2-instanceid" — false: "Server" : "UDM-udm2-instanceid" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid"

Table 3-17 (Cont.) Server Header Behavior

Serial No.	Scenario	Server Header Behavior
9	UDM1 sends an error response with server header and UDM2 sends an error response without server header.	- Default behavior when addServerHeaderInProducerResponse is set to: — true: "Server" : "UDM-udm2-instanceid" — false: SCP won't add any server header - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "UDM-udm1-instanceid"
10	UDM1 sends an error response without server header.	- Default behavior when addServerHeaderInProducerResponse is set to: — true: "Server" : "UDM-udm1-instanceid" — false: SCP won't add any server header - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"
11	UDM1 sends an error response with server header.	- Default behavior when addServerHeaderInProducerResponse is set to: — true: "Server" : "UDM-udm1-instanceid" — false: "Server" : "UDM-udm1-instanceid" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "UDM-udm1-instanceid"
12	UDM1 sends an error response with server header and UDM2 SCP couldn't do Alternate route due to some issue.	- Default behavior when addServerHeaderInProducerResponse is set to: — true: "Server" : "UDM-udm1-instanceid" — false: "Server" : "UDM-udm1-instanceid" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "UDM-udm1-instanceid"
13	UDM1 sends an error response without server header and UDM2 SCP couldn't do Alternate route due to some issue.	- Default behavior when addServerHeaderInProducerResponse is set to: — true: "Server" : "UDM-udm1-instanceid" — false: SCP won't add any server header - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"
14	SCP is unable to connect with UDM1 and UDM2 due to connection issue.	- Default behavior when addServerHeaderInProducerResponse is set to: — true: "Server" : "SCP-<SCP FQDN>" — false: "Server" : "SCP-<SCP FQDN>" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "SCP-<SCP FQDN>" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"

Table 3-17 (Cont.) Server Header Behavior

Serial No.	Scenario	Server Header Behavior
15	Timeout has occurred at UDM1.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "SCP-<SCP FQDN>" — false: "Server" : "SCP-<SCP FQDN>" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "SCP-<SCP FQDN> UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"
16	UDM1, UDM2, and UDM3 send error responses without server header.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "UDM-udm3-instanceid" — false: SCP won't add any server header - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm3-instanceid UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"
17	UDM1, UDM2, and UDM3 send error responses with server header.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "UDM-udm3-instanceid" — false: "Server" : "UDM-udm3-instanceid" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm3-instanceid UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "UDM-udm3-instanceid UDM-udm2-instanceid UDM-udm1-instanceid"
18	UDM1 and UDM3 have connection issues and the pods are down. UDM2 sends an error response without server header.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "UDM-udm2-instanceid" — false: SCP won't add any server header - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm2-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"
19	UDM1 and UDM3 have connection issues and the pods are down. UDM2 sends an error response with server header.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "UDM-udm2-instanceid" — false: "Server" : "UDM-udm2-instanceid" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm2-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "UDM-udm2-instanceid"
20	UDM1 sends an error response with server header. Timeout has occurred at UDM2, and UDM3 has connection issue and the pod is down.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "UDM-udm1-instanceid" — false: "Server" : "UDM-udm1-instanceid" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "UDM-udm1-instanceid"

Table 3-17 (Cont.) Server Header Behavior

Serial No.	Scenario	Server Header Behavior
21	UDM1 sends an error response without server header. Timeout has occurred at UDM2, and UDM3 has connection issue and the pod is down.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "UDM-udm1-instanceid" — false: SCP won't add any server header - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"
22	Loop detection (508) occurred.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "SCP-<SCP FQDN>" — false: "Server" : "SCP-<SCP FQDN>" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "SCP-<SCP FQDN>" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"
23	UDM1 sends an error response with server header. Timeout has occurred at UDM2, and UDM3 has connection issue and the pod is down.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "UDM-udm1-instanceid" — false: "Server" : "UDM-udm1-instanceid" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "UDM-udm1-instanceid"
24	UDM1 Sidecar returns special server headers.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: SCP forwards the special server headers as it is to the Consumer. — false: SCP forwards the special server headers as it is to the Consumer. - Behavior in SCP 23.1.0 (Enhancement 1): SCP forwards the special server headers as it is to the Consumer. - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"
25	Ingress Rate Limiting: SCP Generates an error message because the ingress rate was more than the allowed rate configured for the consumer as specified by the rules configured in the Ingress Rate Limiter REST API.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "SCP-<SCP FQDN>" — false: "Server" : "SCP-<SCP FQDN>" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "SCP-<SCP FQDN>" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"
26	Pod Overload: SCP-worker pod is overloaded due to CPU overload or pending transactions.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: — true: "Server" : "SCP-<SCP FQDN>" — false: "Server" : "SCP-<SCP FQDN>" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "SCP-<SCP FQDN>" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN>"

Table 3-17 (Cont.) Server Header Behavior

Serial No.	Scenario	Server Header Behavior
27	OCI: UDM1 SCP doesn't attempt on UDM1 due to 100% OCI UDM2 SCP receives error response with server header.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: – true: "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid" – false: "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "UDM-udm2-instanceid UDM-udm1-instanceid"
28	OCI: UDM1 SCP doesn't attempt on UDM1 due to 100% OCI. UDM2 SCP doesn't attempt on UDM2 due to 100% OCI.	- Default behavior when <code>addServerHeaderInProducerResponse</code> is set to: – true: "Server" : "SCP-<SCP FQDN> UDM-udm2-instanceid UDM-udm1-instanceid" – false: "Server" : "SCP-<SCP FQDN> UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.1.0 (Enhancement 1): "Server" : "SCP-<SCP FQDN> UDM-udm2-instanceid UDM-udm1-instanceid" - Behavior in SCP 23.2.0 (Enhancement 2): "Server" : "SCP-<SCP FQDN> UDM-udm2-instanceid UDM-udm1-instanceid"

Managing Enhanced 5G SBI Message Failure Handling

Enable

This default behavior, which is specified in 3GPP TS 29.500, of this feature is enabled automatically in SCP at the time of deployment. To enable the enhanced server header behavior introduced in SCP 1.15.0, set **Enable Enhance Server Header Behavior** to true. To enable the enhanced server header behavior introduced in SCP 23.1.0 and later, set **Enable Enhance Server Header Behavior V2** to true. For more information, see [Configuring Server Header](#).

Note

The enhanced server header behavior of SCP 1.15.0 and SCP 23.1.0 are mutually exclusive.

Configure

You can configure the Enhanced 5G SBI Message Failure Handling feature using the CNC Console or REST API.

- **Configure using CNC Console:** For information about server header parameters configuration, see [Configuring Server Header](#).
- **Configure using REST API:** For information about server header configurations, see "Configuring Server Header" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

Updated the `ocscp_metric_sidecarproxy_failures_total` metric with the `ocscp_server_header` dimension in the [Metrics](#) section.

Added the [SCPServiceMeshFailure](#) alert in the [Alerts](#) section.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.23 Extension of HTTP Status Code List for Rerouting

SCP can reroute a request when a producer responds with an error code. Prior to this release, rerouting took place based on the configurable response codes. However, the list of configurable error codes was limited to a smaller set of enumerations (ENUMs), such as `GATEWAY_ERROR`, few other status codes and ENUMs.

With this enhancement, SCP now supports rerouting a request based on an extended list of configurable HTTP Status codes. The user-provided configurable list of error codes serves as the basis for rerouting. A wider set of numerical response codes is supported with the introduction of the new attribute "reRouteConditionList," which replaced the old attribute "reRouteOnResponseCodeList" in the Routing Options REST API. When a received error code matches a status code in the configurable list, the SCP reroutes the request based on the configured alternate routing mechanism.

Managing Configurable HTTP Status Codes for Rerouting

Enable

You can enable this feature using REST API by setting the `reRouteConditionList` routing option parameter. For more information about this parameter, see "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Configure

You can configure the HTTP Status Codes for Rerouting functionality using the CNC Console or REST API.

- **Configure using CNC Console:** For information about how to configure HTTP Status Codes for Rerouting, see [Configuring Routing Options](#).
- **Configure using REST API:** For information about how to configure HTTP Status Codes for Rerouting, see "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

There is no specific metrics and alerts required for the Extension of HTTP Status Code List for Rerouting functionality.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.24 Load Balancing

The 5G core network is a service-based architecture that does not lend itself to an efficient load balancing of provider NFs. By introducing Service Communication Proxy (SCP) in the midst, the load balancing across available NFs can be significantly improved. SCP has a complete view of all the messages arriving for a given NF type and supports traffic distribution patterns, such as round-robin, based on its capacity and availability by enabling uniform load balancing strategy across the network.

SCP supports Load Balancing Algorithm: Priority and Capacity. Load balancing across the equivalent NF or NF Services is based on the priority and capacity.

Managing Load Balancing

Enable

The Load Balancing feature is a core functionality of SCP.

You can enable this feature using REST API by setting the `routePolicy` routing option parameter to `Load_Balance`. For more information about this parameter, see "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Note

This feature is enabled automatically if you are deploying SCP with Release 16 of Model C. For more information about Model C, see [Support for Model C Indirect 5G SBI Communication](#).

Configure

You can configure the Load Balancing functionality using the CNC Console or REST API.

- **Configure using CNC Console:** For information about how to configure load balancing, see [Configuring Routing Options](#).
- **Configure using REST API:** For information about how to configure load balancing, see "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

There is no specific metrics and alerts required for the Load Balancing functionality.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.25 5G SBI Message Routing

Service Communication Proxy (SCP) provides 5G Service Based Interface (SBI) message routing control and brings resiliency to the network. It relieves user NFs from remembering and interpreting complex routing rules associated with the next hop selection. Concurrently, it makes rerouting decisions based on the load conditions and health status of NF providers within the configured time period.

SCP acts as a proxy and forwards and routes any 5G ingress service request to the host (5G NF) present in the request's URI.

For Release 15, SCP supports:

Note

Release 15 deployment model is not supported from SCP 23.4.0.

- Routing based on International Mobile Subscriber Identity (IMSI) and Mobile Station International Subscriber Directory Number (MSISDN)
 - Routing based on Subscription Permanent Identifier (SUPI) and Generic Public Subscription Identifier (GPSI) are supported based on the start and end attributes as published in NF Profile and routing based on the pattern attribute is not supported.
- Only IPv4 IP family
- Only IPv6 IP family
- Every producer NFs are required to publish IpEndpoints for each NFServices in NFProfile while registering with NRF.
- Routing Scope: Site
- DB Sync Status: No Sync, Site Wide
 - Load Balancing and alternate routing is based on DB Sync Status within the Site

For Release 16, SCP supports:

- 5G SBI Model C indirect communication
- 5G SBI Model D indirect communication
- Learns 5G NF topology from NRF
- Static configuration of 5G NF topology
- Load Control based on the Load Control Information Header and so on.

- IPv4 IP family
- Every NF deployed in the network is expected to register with NRF as SCP learns the deployed network's NF topology from NRF.

Alternate Routing

SCP performs alternate routing by forwarding the message request to the next available producer NF if the target producer NF:

- Is unavailable
- Doesn't respond due to a timeout
- Responds with an error

An alternate routing attempt is considered when a connection is established and a message request is sent out. If the SCP fails to establish a connection, it is not counted as an attempt.

After receiving a message request, if SCP does not find any routing rules, SCP uses a default routing rule to route the request directly to the target producer NF. In such scenarios, alternate routing and load balancing functionalities are not applicable.

SCP provides circuit breaking functionality by rejecting message requests for failed or degraded NFs when alternate routing is unavailable. This process prevents consumer NFs from using their resources because they wait for responses from producer NFs. SCP attempts to get a response from producer NFs on behalf of consumer NFs by allowing the service user to focus on the application..

SCP also reroutes message requests with one of the configured HTTP status codes. For more information about the status codes, see "HTTP Status Code and Applicability for Rerouting" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

When one of the following nexthop SCP/SEPP routing failure conditions occurs, SCP performs alternative routing:

- Error received from nextHop SCP/SEPP
- Request timeout at the SCP
- Transport error

Types of Routing Options Supported by SCP

The types of routing options are:

- Pod Level Routing: Controls routing across the pods of selected NF
- Service Level Routing: Controls routing of selected NF services

InterSCP Routing

SCP supports Inter-SCP routing for scenarios where routing across regions is required.

The network can be divided into different geographical regions, with each region deploying redundant NF instances. Each of these regions can be deployed with their own georedundant NRFs. There are scenarios when traffic routing is required across the regions.

In such cases, SCP has to learn the network topology by interacting with NRFs in different regions. SCP also discovers other SCPs in the network. To enable routing to producer NF instances in other regions, the consumer's SCP in a region is aware of the producer's SCP in other regions.

SCP defines its serving Locality in the servingScope parameter in its profile registered with NRF and learns the NF topology from NRF. SCP learns other SCPs from NRF in the same way as it learns from any other producer.

When consumer NF's SCP receives a request, it finds the locality of the selected producer NF instance.

If the producer NF locality is not served by the consumer SCP, the consumer NF's SCP finds the SCP serving the producer NF's locality and routes the request to that SCP. The producer's SCP forwards the request to the producer NF within its served locality.

If the consumer SCP serves the producer locality, it is routed by the consumer SCP. If not, the request is forwarded to the producer SCP serving the NF locality for routing in its served region.

"User-Agent" Header Formats

SCP adds the "User-Agent" header to SCP originated message requests to NRF.

SCP uses the userAgentHeaderFormat REST API parameter to define the format of the "User-Agent" header that is added in the SCP originated message requests towards NRF. SCP supports the following formats of the "User-Agent" header:

- NFTYPE-NFINSTANCEID FQDN
- NFTYPE-NFINSTANCEID-FQDN
- NFTYPE-FQDN NFINSTANCEID
- NFTYPE-FQDN-NFINSTANCEID
- NFTYPE-NFINSTANCEID
- NFTYPE-FQDN
- NFTYPE
- NFTYPE-UNIQUEID

This enhancement enables SCP to support above mentioned "User-Agent" formats and includes consumer NFs identity as dimensions in SCP metrics, alerts, and logs. The following dimension are introduced as part of this enhancement to filter SCP metrics based on the consumer NF information: ocscp_consumer_host, ocscp_consumer_nf_instance_id, and ocscp_consumer_nf_type.

SCP uses the scp_user_agent_info REST API parameter to enable or disable the addition of the "User-Agent" header in the SCP originated messages towards NRF. For more information about the scp_user_agent_info parameter, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Loop Detection

The loop detection feature is used to identify the loops in the 5G control plane network. SCP detects loops using the via header in the incoming request. SCP looks for its own FQDN presence in via header. If such a case occurs, this implies that the same SCP is invoked twice in the same request sequence, which means there is a loop. Hence, SCP rejects such a request. The loop detection feature is enabled by default.

Managing 5G SBI Message Routing

Enable

SCP determines 5G NF topology information from NRF. You can enable the 5G SBI Message Routing functionality by providing all the NRF profiles' information in the YAML file as described

in the global parameters section in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

① Note

- This feature is enabled automatically if you are deploying SCP with Release 16 of Model C. For more information about Model C, see [Support for Model C Indirect 5G SBI Communication](#).
- You can configure the required Helm parameters for this feature if you have performed a fresh installation of SCP or upgraded SCP within the same release version, for example, 1.15.1 to 1.15.1.

Configure

You can configure the 5G SBI Message Routing functionality using the CNC Console or REST API.

- **Configure using CNC Console:** For information about how to configure routing options, see [Configuring Routing Options](#).
- **Configure using REST API:** For information about configuring routing options, see "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.25.1 5G SBI Notification Requests Processing

The following table describes ApiPrefix usage in notification messages at SCP:

Table 3-18 5G SBI Notification Requests Processing

Notification Message Received at SCP			Forwarded Notification Message to NextHop SCP (SCP-P)			Forwarded Notification Message to Target NF			Description
3gpp-sbi-target-apiRoot Header	:authority Header	:path Header	3gpp-sbi-target-apiRoot Header	:authority Header	:path Header If the next hop SCP has published ApiPrefix in the SCP profile.	Routing Attempt	:authority Header	:path Header	-
scheme://NF-C1.com:port/NF-C1ApiPrefix	scp.com:port	scpApiPrefix/notification Uri	scheme://NF-C1.com:port/NF-C1ApiPrefix	nextHopScp.com:port	{nextHops cpApiPrefix}/notification Uri	Primary routing	scheme://NF-C1.com:port	NF-C1ApiPrefix/notification Uri	While registering with NRF, the notification handling service published apiPrefix in NFProfile.
						Alternate routing	scheme://NF-C2.com:port	{NF-C2ApiPrefix}/notification Uri	NF-C2ApiPrefix is used as per deployment and the apiPrefix value in the NFProfile for the service.
scheme://NF-C1.com:port/	scp.com:port	scpApiPrefix/notification Uri	scheme://NF-C1.com:port/	nextHopScp.com:port	{nextHops cpApiPrefix}/notification Uri	Primary routing	scheme://NF-C1.com:port	/notification Uri	NF-C2ApiPrefix is used as per deployment and the apiPrefix value in the NFProfile for the service.

Table 3-18 (Cont.) 5G SBI Notification Requests Processing

Notification Message Received at SCP			Forwarded Notification Message to NextHop SCP (SCP-P)			Forwarded Notification Message to Target NF			Description
3gpp-sbi-target-apiRoot Header	:authority Header	:path Header	3gpp-sbi-target-apiRoot Header	:authority Header	:path Header If the next hop SCP has published ApiPrefix in the SCP profile.	Routing Attempt	:authority Header	:path Header	-
						Alternate routing	scheme://NF-C2.com:port	{NF-C2ApiPrefix}/notification Uri	NF-C2ApiPrefix is used as per deployment and the apiPrefix value in the NFProfile for the service.
scheme://NF-C1.com:port/NF-C1ApiPrefix	scp.com:port	/notification Uri	scheme://NF-C1.com:port/	nextHopScp.com:port	{nextHopScpApiPrefix}/notification Uri	Primary routing	scheme://NF-C1.com:port	NF-C1ApiPrefix/notification Uri	SCP is deployed without apiPrefix in SCP profile.
						Alternate routing	scheme://NF-C2.com:port	{NF-C2ApiPrefix}/notification Uri	NF-C2ApiPrefix is used as per deployment and the apiPrefix value in the NFProfile for the service.
scheme://NF-C1.com:port/	scp.com:port	/notification Uri	scheme://NF-C1.com:port/	nextHopScp.com:port	{nextHopScpApiPrefix}/notification Uri	Primary routing	scheme://NF-C1.com:port	/notification Uri	SCP and the notification handling service are deployed without an apiPrefix in the NFProfile.

Table 3-18 (Cont.) 5G SBI Notification Requests Processing

Notification Message Received at SCP			Forwarded Notification Message to NextHop SCP (SCP-P)			Forwarded Notification Message to Target NF			Description
3gpp-sbi-target-apiRoot Header	:authority Header	:path Header	3gpp-sbi-target-apiRoot Header	:authority Header	:path Header If the next hop SCP has published ApiPrefix in the SCP profile.	Routing Attempt	:authority Header	:path Header	-
						Alternate routing	scheme://NF-C2.com:port	{NF-C2ApiPrefix}/notificationUri	NF-C2ApiPrefix is used as per deployment and the apiPrefix value in the NFProfile for the service.
Delegated Discovery notification request	scp.com:port	{scpApiPrefix}/scp-default-sub-notify-uri	scheme://NF-C1.com:port/ • Select ed target notification URI	nextHopScp.com:port	{nextHops cpApiPrefix}/notificationUri	Primary routing	scheme://NF-C1.com:port	{NF-C1ApiPrefix}/notificationUri	Notification URI is used according to the published NFProfile based on discovery responses .
						Alternate routing	scheme://NF-C2.com:port	{NF-C2ApiPrefix}/notificationUri	Notification URI is used according to the published NFProfile based on discovery responses .

Table 3-18 (Cont.) 5G SBI Notification Requests Processing

Notification Message Received at SCP			Forwarded Notification Message to NextHop SCP (SCP-P)			Forwarded Notification Message to Target NF			Description
3gpp-sbi-target-apiRoot Header	:authority Header	:path Header	3gpp-sbi-target-apiRoot Header	:authority Header	:path Header If the next hop SCP has published ApiPrefix in the SCP profile.	Routing Attempt	:authority Header	:path Header	-
Delegated Discovery notification request	scp.com:port	/scp-default-sub-notify-uri	scheme://NF-C1.com:port/ • Selected target notification URI	nextHopScp.com:port	{nextHopScpApiPrefix}/notification Uri	Primary routing	scheme://NF-C1.com:port	{NF-C1ApiPrefix}/notification Uri	Notification URI is used according to the published NFProfile based on discovery responses.
						Alternate routing	scheme://NF-C2.com:port	{NF-C2ApiPrefix}/notification Uri	Notification URI is used according to the published NFProfile based on discovery responses.

Obtaining NF Information from Notification Requests

SCP obtains the target producer NF information from `nfset` and `nfinst` attributes of the notification request even if the `3gpp-Sbi-Routing-Binding` header does not contain the `servname` attribute. In this case, SCP pegs the exact values of the producer NF type in `ocscp_nf_type` and `ocscp_nf_service_type` dimensions of the respective metrics. The format of dimension values are as follows:

- `ocscp_nf_type=<extracted_nftype>`
- `ocscp_nf_service_type=<extracted_nftype>-unknown`

Where,

- `<extracted_nftype>` is the target producer NF type.
- `unknown` indicates that the `servname` attribute is absent in the `3gpp-Sbi-Routing-Binding` header.

In case the `servname` attribute is present in the `3gpp-Sbi-Routing-Binding` header, SCP pegs the exact values of the producer NF type in `ocscp_nf_type` and `ocscp_nf_service_type` dimensions of the respective metrics. The format of dimension values are as follows:

- `ocscp_nf_type=<extracted_nftype_fromservname>`
- `ocscp_nf_service_type=<extracted_nftype_fromservname>-<service name>`

Where,

- `<extracted_nftype_fromservname>` is the target producer NF type extracted from the `servname` attribute.
- `<service name>` is the service name present in the 3gpp-Sbi-Routing-Binding header.

3.25.2 5G SBI Service Requests Processing

The ApiPrefix usage in service request messages at SCP are as follows:

Table 3-19 5G SBI Service Requests Processing

5G SBI Request Message Received at SCP			SBI Service Request to NextHop SCP			Forwarded SBI Service Request Message to Target NF			Description
3gpp-sbi-target-apiRoot Header	:authority Header	:path Header	3gpp-sbi-target-apiRoot Header	:authority Header	:path Header	Routing Attempt	:authority Header	:path Header	-
scheme://NF-P1.com:port/NF-P1ApiPrefix	scp.com:port	scpApiPrefix/<apiName>/<apiVersion>apiSpecificResourceURI	scheme://NF-P1.com:port/NF-P1ApiPrefix	nextHopScp.com:port	{nextHopScpApiPrefix}/<apiName>/<apiVersion>/apiSpecificResourceURI If the next hop SCP has published ApiPrefix in the SCP profile.	Primary routing	scheme://NF-P1.com:port	NF-P1ApiPrefix/<apiName>/<apiVersion>/apiSpecificResourceURI	While registering with NRF, the NF service published apiPrefix in NFProfile.
						Alternate routing	scheme://NF-P2.com:port	{NF-P2ApiPrefix}/<apiName>/<apiVersion>/apiSpecificResourceURI	NF-P2ApiPrefix is used according to deployment and the apiPrefix value in the NFProfile for the service.
scheme://NF-P1.com:port/	scp.com:port	scpApiPrefix/<apiName>/<apiVersion>/apiSpecificResourceURI	scheme://NF-P1.com:port/	nextHopScp.com:port	{nextHopScpApiPrefix}/<apiName>/<apiVersion>/apiSpecificResourceURI If the next hop SCP	Primary routing	scheme://NF-P1.com:port	/<apiName>/<apiVersion>/apiSpecificResourceURI	NF service is not supporting apiPrefix and not published in NF Profile.

Table 3-19 (Cont.) 5G SBI Service Requests Processing

5G SBI Request Message Received at SCP			SBI Service Request to NextHop SCP			Forwarded SBI Service Request Message to Target NF			Description
3gpp-sbi-target-apiRoot Header	:authority Header	:path Header	3gpp-sbi-target-apiRoot Header	:authority Header	:path Header	Routing Attempt	:authority Header	:path Header	-
					has published ApiPrefix in the SCP profile.	Alternate routing	scheme://NF-P2.com:port	{NF-P2ApiPrefix}/<apiName>/<apiVersion>/apiSpecificResource URI	NF-P2ApiPrefix is used according to deployment and the apiPrefix value in the NFProfile for the service.
scheme://NF-P1.com:port/NF-P1ApiPrefix	scp.com:port	/<apiName>/<apiVersion>/apiSpecificResource URI	scheme://NF-P1.com:port/	nextHopScp.com:port	{nextHopScpApiPrefix}/<apiName>/<apiVersion>/apiSpecificResource URI If the next hop SCP has published ApiPrefix in the SCP profile.	Primary routing	scheme://NF-P1.com:port	NF-P1ApiPrefix/<apiName>/<apiVersion>/apiSpecificResource URI	SCP is deployed without apiPrefix in SCP profile.
						Alternate routing	scheme://NF-P2.com:port	{NF-P2ApiPrefix}/<apiName>/<apiVersion>/apiSpecificResource URI	NF-P2ApiPrefix is used according to deployment and the apiPrefix value in the NFProfile for the service.
scheme://NF-P1.com:port/	scp.com:port	/<apiName>/<apiVersion>/apiSpecificResource URI	scheme://NF-P1.com:port/	nextHopScp.com:port	{nextHopScpApiPrefix}/<apiName>/<apiVersion>/apiSpecificResource URI If the next hop SCP	Primary routing	scheme://NF-C1.com:port	/<apiName>/<apiVersion>/apiSpecificResource URI	SCP and NF service are deployed without an api prefix in the NFProfile.

Table 3-19 (Cont.) 5G SBI Service Requests Processing

5G SBI Request Message Received at SCP			SBI Service Request to NextHop SCP			Forwarded SBI Service Request Message to Target NF			Description
3gpp-sbi-target-apiRoot Header	:authority Header	:path Header	3gpp-sbi-target-apiRoot Header	:authority Header	:path Header	Routing Attempt	:authority Header	:path Header	-
					has published ApiPrefix in the SCP profile.	Alternate routing	scheme://NF-C2.com:port	{NF-P2ApiPrefix}/<apiName>/<apiVersion>/apiSpecificResource URI	NF-P2ApiPrefix is used according to deployment and the apiPrefix value in the NFProfile for the service.
Delegated Discovery 5G SBI request (Not Present)	scp.com:port	{scpApiPrefix}/<apiName>/<apiVersion>/apiSpecificResource URI	scheme://NF-P1.com:port/Selected target notification URI	nextHopScp.com:port	{nextHops cpApiPrefix}/<apiName>/<apiVersion>/apiSpecificResource URI If the next hop SCP has published ApiPrefix in the SCP profile.	Primary routing	scheme://NF-P1.com:port	{NF-P1ApiPrefix}/<apiName>/<apiVersion>/apiSpecificResource URI	apiRoot is created according to the published NFProfile based on discovery responses .
						Alternate routing	scheme://NF-P2.com:port	{NF-P2ApiPrefix}/<apiName>/<apiVersion>/apiSpecificResource URI	apiRoot is created according to the published NFProfile based on discovery responses .
Delegated Discovery 5G SBI request (Not Present)	scp.com:port	/scp-default-sub-notify-uri	scheme://NF-P1.com:port/Selected target notification URI	nextHopScp.com:port	{nextHops cpApiPrefix}/<apiName>/<apiVersion>/apiSpecificResource URI If the next hop SCP has published ApiPrefix	Primary routing	scheme://NF-P1.com:port	{NF-P1ApiPrefix}/<apiName>/<apiVersion>/apiSpecificResource URI	apiRoot is created according to the published NFProfile based on discovery responses .

Table 3-19 (Cont.) 5G SBI Service Requests Processing

5G SBI Request Message Received at SCP			SBI Service Request to NextHop SCP			Forwarded SBI Service Request Message to Target NF			Description
3gpp-sbi-target-apiRoot Header	:authority Header	:path Header	3gpp-sbi-target-apiRoot Header	:authority Header	:path Header	Routing Attempt	:authority Header	:path Header	-
					in the SCP profile.	Alternate routing	scheme://NF-P2.com:port	{NF-P2ApiPrefix}/<apiName>/<apiVersion>/apiSpecificResource URI	apiRoot is created according to the published NFProfile based on discovery responses.

3.26 5G SBI Message Mediation Support

SCP supports Mediation as an independent service to modify 5G Service Based Interface (SBI) message content, which includes HTTP2 header values and JSON message body, based on the user-defined mediation rule sets.

SCP uses microservice, scp-mediation, that applies user configured message mediation policy rules to ingress 5G SBI messages to perform message mediation transformation. The scp-worker microservice applies mediation invocation filter criteria according to the mediation trigger point configuration and forwards 5G SBI messages to the scp-mediation microservice.

The following parameters are configured to forward SBI messages to Mediation from the trigger points:

- **nfType:** It routes all message requests for this nfType to the scp-mediation microservice.
- **serviceName:** It routes all message requests for this serviceName to the scp-mediation microservice.
- **httpMethods:** It routes all messages of a particular HTTP method type, which corresponds to the methods specified in this list, to the scp-mediation microservice.
- **triggerPoints:** It contains the list of trigger points that must be enabled to forward message requests from SCP to the scp-mediation microservice from the enabled trigger point. It has four values, requestIngress, requestEgress, responseIngress, and responseEgress.
- **match:** This block specifies the criteria required to match HTTP headers or JSON body or the both to route to the scp-mediation microservice.

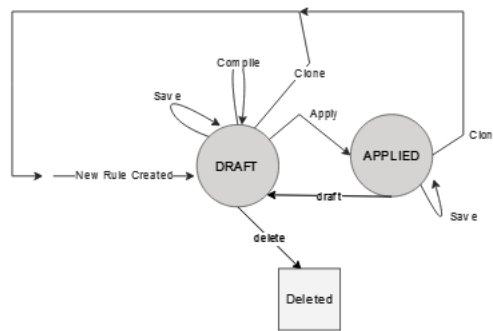
Note

- Mediation is supported only in Release 16 SCP deployment.
- Mediation does not support multipart messages.

3.26.1 Mediation Rules Configuration

Mediation rules stored in the database can be retrieved, new rules can be added, and existing rules can be modified using the CCNC GUI and RESTful API. In addition, the storage capacity has improved.

Figure 3-6 Rule Status Flow



The following table lists the state of rules and its description

Table 3-20 Types of State

State	Description
Clone	To create a new rule, the values of the rule already created and stored in the database are used. This newly created rule has a name followed by a term draft. A new name must be provided as a rule.
Compile	This state precompiles, takes the rule source, and builds an executable rule base.
Draft	This state moves a rule from APPLIED status to DRAFT status.
Apply	This state moves a rule from DRAFT status to APPLY status.
Save	This state saves the changes in database.

The table lists the modifications allowed to be configured in rule status:

Table 3-21 Rule Status

Field Name	DRAFT status	Applied status
Rule Name	Not Allowed	Not Allowed
Format	Not Allowed	Not Allowed
Status	Not Allowed	Not Allowed
Mediation Mode	Allowed	Allowed
Code	Allowed	Not Allowed
State	Allowed	Allowed

Note

The Rule Name must be unique. The status of a rule can be DRAFT or APPLIED. Only DRAFT status can be deleted, edited, or viewed.

This section describes APIs that are used to define Mediation rules.

Header Rule APIs

The following are the header APIs used to create header rules supported in Mediation.

Table 3-22 Header Rule APIs

API Name	Return Type	Description
get(String key)	String	Returns the header value for the key specified. Example: Request(headers.get("x-number"))
has(String key)	Boolean	Returns true or false depending on whether the header specified by the key attribute is present or absent. Example: Request(headers.has("3gpp-Sbi-Producer-Id"))
has(String key, String value)	Boolean	Returns true or false depending on whether the header specified by the key attribute is present. Example: Request(headers.has("3gpp-Sbi-Producer-Id","nfinst=1faf1bbc-6e4a-3994-a507-a14ef8e1bc23"))
put(String key, String value)	Boolean	Creates the header with the specified key value pair if the key is absent. If the key is present, it updates the value of the header with the specified value. Example: req.headers.put("x-original-authority", "10.172.19.110:8080")
add(String key, String value);	Boolean	Adds the header with the provided key and value if it does not find the key. Example: req.headers.add("Content-type","application/json")
set(String key, String value);	Boolean	Updates the header with the value if the provided key is found. Example: req.headers.set("3gpp-Sbi-Producer-Id","nfinst=1faf1bbc-6e4a-3994-a507-a14ef8e1bc25")
set(String key, String oldValue, String newValue);	Boolean	Updates the header with a new value if the existing value of the header matches with the value specified by the oldValue attribute. Example: req.headers.set("x-number", "2", "3")
del(String key)	Boolean	Removes the header specified by the key attribute. Example: req.headers.del("3gpp-Sbi-Message-Priority")
del(String key, String value)	Boolean	Removes the header if it matches with the provided key and value. Example: req.headers.del("x-forwarded-NF","NRF")
count();	Integer	Returns the total count of all the headers. Example: Request(headers.count())

Table 3-22 (Cont.) Header Rule APIs

API Name	Return Type	Description
count(String key);	Integer	Returns the total count of headers with the key. Example: Request(headers.count("Accept"))
count(String key, String value)	Integer	Returns the total count of headers with the key and value. Example: Request(headers.count("Accept", "application/json"))

APIs for Modifying Request Body

The following table describes the rule APIs to update and delete HTTP message body that are supported in mediation:

Note

The paths in all the parameters must be in the JSONPath format.

Table 3-23 APIs for Modifying Request Body

API Name	Return Type	Description
get(String path)	Object	Returns the value present at the specified path. Example: body.get("\$.serviceName")
getAll(String path)	List<Object>	Returns the list of objects present at the specified path. Example: body.getAll("\$.serviceName")
has(String path);	boolean	Checks whether the key is present at the path or not. Example: body.has("\$.serviceName")
has(String path, Object value)	boolean	Checks whether the key has the value present at the path or not. Example: body.has("\$.serviceName", "value")
absPath(String arrayPath, String elem, Object value)	Object	Returns the whole path from the head to the element if the value is present. Example: req.body.absPath("\$.payload", "iePath", "/a")
indexOf(String arrayPath, String elem, Object value)	Integer	Returns the index of the value in that JSONArray. Example: req.body.indexOf("\$.dataToIntegrityProtectBlock.payload", "value", 100)
put(String path, String key, T value)	boolean	Adds the body with the provided key and value at the path if the key is absent. Otherwise, update the body with the value at the path. Example: req.body.put("\$", "key", "value") Note: It is recommended to use the PUT method while trying to add a new key.

Table 3-23 (Cont.) APIs for Modifying Request Body

API Name	Return Type	Description
add(String path, T value)	boolean	Adds the key embedded in the path with the provided value if the element is absent. Example: req.body.add("\$.ipEndPoints", "value") Note: When adding a new key, the PUT method is always preferred over the ADD method.
set(String path, T value)	boolean	Updates the key embedded in the path with the provided value if the element is present. Example: rsp.body.add("\$.ipEndPoints", "value")
del(String path, T value)	boolean	Removes the element at that path with the value present. Not supported. Example: req.body.del("\$.path", "value")
del(String path)	boolean	Removes the element at that path. Example: req.body.del("\$.path")

Http Message Common APIs

The following are the common Http message APIs supported by both request and response body.

Table 3-24 Http Message Common APIs

API Name	Return Type	Description
getUri()	String	Returns the URI of the request. Example: req.getUri()
setUri(String URI)	void	Sets the URI in the body of the request. Example: req.setUri("http://oracle.com")
getHttpStatusCode()	Integer	Returns the StatusCode from the body of the response. Example: rsp.getHttpStatusCode()
setHttpStatusCode(Integer httpStatusCode)	void	Sets the StatusCode in the body of the response. Example: rsp.setHttpStatusCode(201)
getRejectMessage()	String	Returns the Reject Message from the body of the response. Example: rsp.getRejectMessage()
setRejectMessage(String rejectMessage)	void	Sets the Reject Message in the body of the response. Example: rsp.setRejectMessage("Message")

Mediation Service Rule Configuration with Scenarios

This section captures various types of mediation rule configuration with scenarios.

Use Case 1

To update the headers for specific requests.

Scenario

The mediation rule is configured when headers get ("pduSessionId"="23476") and add the header (X-Service-Name=PCF) value to the request.

Mediation Rule

```
rule "Rule2_PCF"
  agenda-group "CUSTOM_ORACLE_SCP-requestEgress"
  when
    req : Request(headers.get("pduSessionId") == "23476")
  then
    req.headers.put("x-service-name", "PCF")
  end
```

Use Case 2

To add, update, or delete any attribute in the JSON payload in the request or response.

Scenario

Mediation rule is configured as when incoming response payload got servingNetworkName="5G:nmc310.mcc310.3gppnetwork.org" then delete the "traceData" attribute and add TP="ResIngressAusf" in json payload.

Mediation Rule

```
rule "Rule7_AUSF"
  agenda-group "CUSTOM_ORACLE_SCP-responseIngress"
  when
    res : Response(body.get("$.servingNetworkName")=="5G:nmc310.mcc
310.3gppnetwork.org")
  then
    res.body.del("$.traceData")
    res.body.put("$", "TP", "ResIngressAusf")
  end
```

Use Case 3

To update URI components such as apiRoot, apiName, apiVersion, and apiSpecificResourceUriPart in the request or response.

Scenario

Mediation rule is configured as when request payload got "3gpp-sbi-target-apiroot"="http://ausf1svc.default.svc.cluster.local:8080/USEast" then update the URI to "http://ausf2svc.scpsvc.svc.cluster.local:8080/USEast/nausf-auth/v1/ue-authentications".

Mediation Rule

```
rule "Rule5_AUSF"
  agenda-group "CUSTOM_ORACLE_SCP-requestEgress"
  when
    req : Request(headers.get("3gpp-sbi-target-apiroot") == "http://
ausf1svc.default.svc.cluster.local:8080/USEast")
  then
    req.setUri("http://ausf2svc.default.svc.cluster.local:8080/USEast/nausf-
```



```
auth/v1/ue-authentications")
end
```

Use Case 4

To update the authority in apiRoot coming in the request or response.

Scenario

Mediation rule is configured as when request header got 3gpp-Sbi-Target-apiRoot="http://pcf1svc.default.svc.cluster.local:8080/USEast" update the authority to pcf3 from pcf1.

Mediation Rule

```
rule "Rule6_PCF"
  agenda-group "CUSTOM_ORACLE_SCP-requestEgress"
  when
    req : Request(headers.get("3gpp-sbi-target-apiroot") == "http://
pcf1svc.scpsvc.svc.cluster.local:8080/USEast")
  then
    req.setUri("http://pcf3svc.scpsvc.svc.cluster.local:8080/USEast/npcf-
smpolicycontrol/v1/sm-policies")
  end
```

Use Case 5

To add any vendor-specific header.

Scenario

Mediation rule is configured as when response payload got "sliceInfo"=="adfg127" then add the vendor-specific header ("x-service-name","PCF").

Mediation Rule

```
rule "Rule2_PCF"
  agenda-group "CUSTOM_ORACLE_SCP_requestEgress"
  when
    req : Request(body.get("$.sliceInfo") == "adfg127")
  then
    req.headers.put("x-service-name","PCF")
  end
```

Use Case 6

To convert the response status code that it received to any specific status code..

Scenario

The mediation rule is configured; if mediation receives an HTTP response code of 401, it should update the response code to 402.

Mediation Rule

```
rule "status rule_responseEgress"
  agenda-group "CUSTOM_ORACLE_SCP_responseEgress"
  when
    rsp : Response(getHttpStatusCode() == "402")
  then
```

```
    rsp.setHttpStatusCode(401)
end
```

Use Case 7

To update authority in apiRoot based on a JSON-nested object from the body.

Scenario

The mediation rule is configured to segregate NF Producer based on MSISDN.

To retrieve the nested parameter, servedGPSI, from the below structure:

```
"pDUSessionChargingInformation":{
  "chargingId":747941933,
  "pduSessionInformation":{},
  "ueTimeZone":"+00:00+0",
  "userInformation":{
    "servedGPSI":"msisdn-447432982860",
    "servedPEI":"imeisv-3528878115855501",
    "unauthenticatedFlag":false
  },
  "
```

Mediation Rule

```
rule "G_N40_odd"
  agenda-group "G_N40_odd-requestIngress"
when
  req :
Request(body.get("$.pDUSessionChargingInformation.userInformation.servedGPSI"
)=="msisdn-447432982860")
then
  req.headers.put("3gpp-sbi-target-apiroot",http://10.251.132.183:8080)
end
```

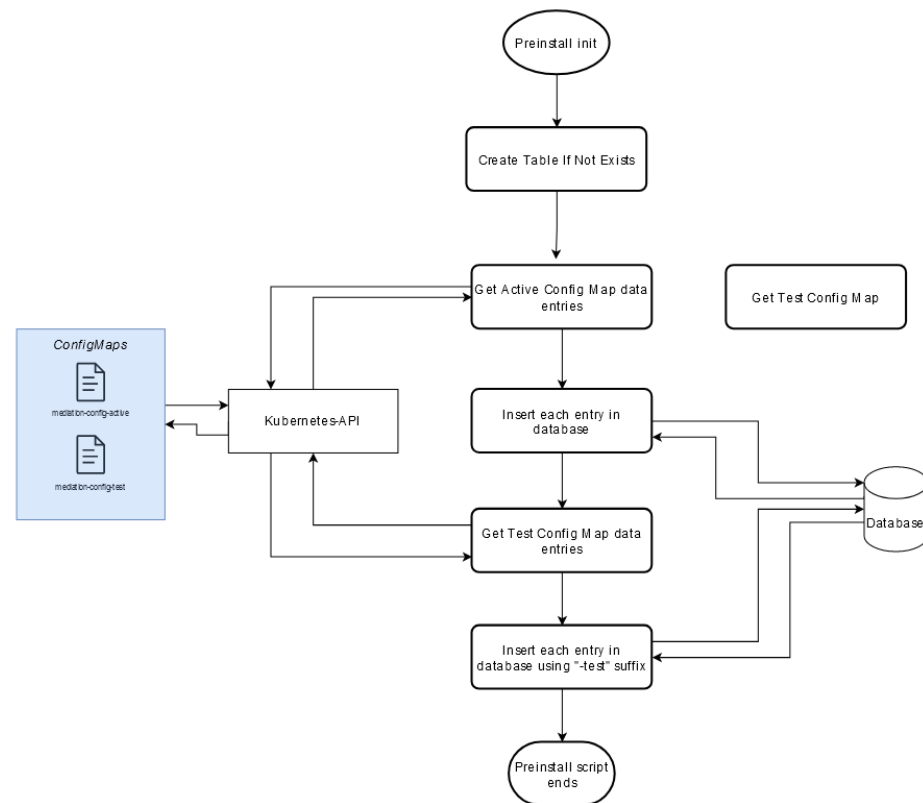
3.26.1.1 Mediation Rule Migration Process

Previously, rules were used from the Kubernetes config map. Currently, the rules present in the config map are migrated to the database table (mediation_rule table).

This section describes the procedure to execute the migration of rules from the existing config-map based implementation into the new REST API.

The following are the config-map related to mediation:

- ocsdp-nf-mediation-config-boot: It contains all the configuration related to the spring boot application and is used in both modes (test and active).
- ocsdp-nf-mediation-config-active: It contains rule entries and is only applicable to active mode.
- ocsdp-nf-mediation-config-test: It contains rule entries and is only applicable to test mode.
- The following image illustrates the migration process:

Figure 3-7 Migration to Database table**Note**

To successfully migrate to database table, all the rules in the Config map must be valid.

Generate Rule Entries

1. Read ocscp-nf-mediation-config-active configuration map.
2. For each entry in key-value map structure, create entry in Rule table with the following:
 - a. Rule.name: entry.key removing .drl extension
 - b. Rule.code: entry.value
 - c. Rule.status: APPLIED
 - d. Rule.mode: MEDIATION_ACTIVE
3. Read ocscp-nf-mediation-config-test configuration map.
4. Create entry in Rule table with the following:
 - a. Rule.name: entry.key removing .drl extension adding "-test" as suffix to avoid name conflicts with active configmap
 - b. Rule.code: entry.value
 - c. Rule.status: APPLIED
 - d. Rule.mode: MEDIATION_TEST

Migration of Rule Entries

The following examples details about the rule migrated from configmap to database table.

Example of ocscp-nf-mediation-config-active configuration map:

```
apiVersion: v1
kind: ConfigMap
metadata:
  creationTimestamp: "2022-11-11T13:15:14Z"
  name: nf-mediation-nf-mediation-config-active
  namespace: scpsvc
data:
  rule_1.drl: |
    package com.oracle.cgbu.ocmediation.nfmediation;

    import com.oracle.cgbu.ocmediation.nfruleengine.NFDroolsRuleEngine;
    import com.oracle.cgbu.ocmediation.factdetails.Request;
    import com.oracle.cgbu.ocmediation.factdetails.Response;
    import java.util.Map;
    import java.util.HashMap;

    dialect "mvel"

    rule "Rule_show_add_header"
    when
      req : Request(headers.has("header-test") == true)
    then
      req.headers.add("3gpp-Sbi-Message-Priority", "10")
      req.headers.del("x-test-req-header1")
    end

    rule "Rule_show_delete_header"
    when
      req : Request(headers.has("header-test2") == true)
    then
      req.headers.del("header-test2")
    end

  rule_2.drl: |
    package com.oracle.cgbu.ocmediation.nfmediation;

    import com.oracle.cgbu.ocmediation.nfruleengine.NFDroolsRuleEngine;
    import com.oracle.cgbu.ocmediation.factdetails.Request;
    import com.oracle.cgbu.ocmediation.factdetails.Response;
    import java.util.Map;
    import java.util.HashMap;

    dialect "mvel"

    rule "Rule_show_add_headerv2"
    when
      req : Request(headers.has("another-header") == true)
    then
```

```

        req.headers.add("new-header2", "10")
    end

```

Example of ocscp-nf-mediation-config-test configuration map:

```

apiVersion: v1
kind: ConfigMap
metadata:
  creationTimestamp: "2022-11-11T13:15:14Z"
  name: nf-mediation-nf-mediation-config-test
  namespace: scpsvc
data:
  rule_1.drl: |
    package com.oracle.cgbu.ocmediation.nfmediation;

    import com.oracle.cgbu.ocmediation.nfruleengine.NFDroolsRuleEngine;
    import com.oracle.cgbu.ocmediation.factdetails.Request;
    import com.oracle.cgbu.ocmediation.factdetails.Response;
    import java.util.Map;
    import java.util.HashMap;

    dialect "mvel"

    rule "Rule_show_add_header"
    when
        req : Request(headers.has("header-test") == true)
    then
        req.headers.add("3gpp-Sbi-Message-Priority", "10")
        req.headers.del("x-test-req-header1")
    end

    rule "Rule_show_delete_header"
    when
        req : Request(headers.has("header-test2") == true)
    then
        req.headers.del("header-test2")
    end

```

In this examples, the ocscp-nf-mediation-config-active configuration map has two different entries (rule_1.drl and rule_2.drl) and rule_1.drl has the definition of two different rules, Rule_show_add_header and Rule_show_delete_header, it is important to consider that rule_1 will have a single rule entry in the database containing all the code for its two rules.

In this case, the database entries are as follows:

Table 3-25 Database Entries

Name	Status	Mode	Code
rule_1	APPLIED	MEDIATION_ACTIVATIVE	<pre> package com.oracle.cgbu.ocmediation.nfmediation; import com.oracle.cgbu.ocmediation.nfruleengine.NFDrools RuleEngine; import com.oracle.cgbu.ocmediation.factdetails.Request import com.oracle.cgbu.ocmediation.factdetails.Response; import java.util.Map; import java.util.HashMap; dialect "mvel" rule "Rule_show_add_header" when req : Request(headers.has("header-test") == true) then req.headers.add("3gpp-Sbi-Message- Priority","10") req.headers.del("x-test-req-header1") end rule "Rule_show_delete_header" when req : Request(headers.has("header-test2") == true) then req.headers.del("header-test2") end </pre>

Table 3-25 (Cont.) Database Entries

Name	Status	Mode	Code
rule_2	APPLIED	MEDIATION_ACTIVATIVE	<pre> package com.oracle.cgbu.ocmediation.nfmediation; import com.oracle.cgbu.ocmediation.nfruleengine.NFDrools RuleEngine; import com.oracle.cgbu.ocmediation.factdetails.Request import com.oracle.cgbu.ocmediation.factdetails.Response; import java.util.Map; import java.util.HashMap; dialect "mvel" rule "Rule_show_add_headerv2" when req : Request(headers.has("another-header") == true) then req.headers.add("new-header2", "10") end </pre>

Table 3-25 (Cont.) Database Entries

Name	Status	Mode	Code
rule_1-test	APPLIED	MEDIATION_TEST	<pre> package com.oracle.cgbu.ocmediation.nfmediation; import com.oracle.cgbu.ocmediation.nfruleengine.NFDrools RuleEngine; import com.oracle.cgbu.ocmediation.factdetails.Request import com.oracle.cgbu.ocmediation.factdetails.Response; import java.util.Map; import java.util.HashMap; dialect "mvel" rule "Rule_show_add_header" when req : Request(headers.has("header-test") == true) then req.headers.add("3gpp-Sbi-Message- Priority","10") req.headers.del("x-test-req-header1") end rule "Rule_show_delete_header" when req : Request(headers.has("header-test2") == true) then req.headers.del("header-test2") end </pre>

3.26.2 Mediation Log Level Configuration

The Mediation Log Level Configuration feature allows users to fetch and update the log level settings of the mediation service. Through the REST API, users can control the mediation log level, enhancing the operational efficiency of the mediation service.

Managing Mediation Support Feature

Enable

To enable 5G SBI Message Mediation Support feature, set the `mediationService` parameter to `true` in the `global parameters` section of the `custom-values.yaml` file. For more information about this parameter, see *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Note

You can configure the required Helm parameters for this feature if you have performed a fresh installation of SCP or upgraded SCP within the same release version.

Configure

You can configure the Mediation features using REST API and the CNC Console.

- **Configure using CNC Console:** You can enable the 5G SBI Message Mediation Support feature by setting `mediation` to `true`. For more information, see [Configuring SCP Features](#). To configure trigger points, see [Configuring Mediation](#) and to configure mediation routing options, see [Configuring Routing Options for Mediation](#).
To configure the mediation rules, see [Configuring Mediation Rules](#).
To configure mediation at the log level, see [Configuring Mediation Log Levels](#) section.
- **Configure using REST API:** For information about configuring trigger points, see "Configuring Mediation Trigger Point", and for configuring mediation routing options, see "Configuring App Routing Options".
To configure mediation rules using the REST API, see "Configuring Mediation Rules".
To configure mediation at the log level using the REST API, see "Configuring Mediation Rules" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.27 Message Priority Assignment and Override

The "3gpp-Sbi-Message-Priority" header is defined to carry the message priority of 5G messages. Service Communication Proxy (SCP) includes or modifies the header based on the configuration parameters.

The operator can assign or override the message priority of 5G messages as per their network requirements. SCP uses the message priority in prioritising the message in congestion or overload features.

Managing Message Priority Assignment and Override**Enable**

Not applicable.

Configure

You can configure the Message Priority Assignment and Override functionality using the CNC Console or REST API.

- **Configure using CNC Console:** For information about how to configure message priority options, see [Configuring SBI Message Priority](#).
- **Configure using REST API:** For information about how to configure message priority options, see "Configuring SBI Message Priority for Pod Overload Control" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.28 Outlier Detection

The Outlier Detection feature monitors the status of each endpoint of the producer NF or NF services. Upstream producer endpoints that continuously return the service responses with non-2xx status codes are removed from the routing pool for a predefined period based on the configured non-2xx status codes, counts, and ejection time.

In SCP 22.4.0, this feature is enhanced to support outlier detection for upstream NFs, which includes local producer NFs, next-hop SCP, and next-hop SEPP.

When SCP receives a message request from a consumer NF, it routes the message request to the target upstream NFs if the upstream NF is healthy. When SCP receives non-2xx status codes consecutively from the target upstream NF, SCP marks the upstream NF unhealthy after the number of error responses exceeds the predefined value. For example, if the `consecutiveErrors` parameter range is set to 5, after receiving five consecutive errors from upstream NF1, SCP marks the upstream NF1 as degraded or unhealthy. When SCP receives the sixth message request from the consumer NF, it routes the message request to upstream NF2 or any other available upstream NF without considering upstream NF1, which is marked unhealthy. SCP starts forwarding message requests to upstream NF1 after the configured time period as defined in the `baseEjectionTime` parameter. For more information about `consecutiveErrors` and `baseEjectionTime` parameters, see "Configuring Outlier Detection" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

For the next hop, SCP or SEPP, these error codes are SCP or SEPP-generated errors. These error responses contain a server header that includes the FQDN of SCP or SEPP.

① Note

SCP resets outlier detection data for producer NF profile updates.

Managing Outlier Detection

Enable

To enable the Outlier Detection feature, complete the following configurations as described in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*:

- For specific NF service, set `odEnabled` to `true`, and then configure `odRuleName` as described in the "Configuring Routing Options" section.
- For Inter SCP scenarios, set `odEnabled` to `true` and configure `odScpRuleName` as described in the "Configuring System Options" section.
- For Inter PLMN routing scenarios, set `odSeppEnabled` to `true` and configure `odSeppRuleName` as described in the "Configuring System Options" section.

Configure

You can configure Outlier Detection feature using the CNC Console and REST API.

- **Configure using CNC Console:** For information about CNC Console configurations, see [Configuring Global Options](#) and [Configuring Outlier Detection](#).
- **Configure using REST API:** For information about REST API configurations, see "Outlier Detection Configuration", "Configuring System Options", and "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.29 Circuit Breaking

The Circuit Breaking feature is triggered on a per NF service instance (including all of the service's IP endpoints or FQDNs) when its outstanding transactions exceed a preconfigured value. If outstanding transaction exceeds the preconfigured value and alternate routing is configured, the message requests are routed to alternate NFs. When Circuit Breaking is activated, message requests are routed alternatively or rejected.

Managing Circuit Breaking

Enable

This feature is enabled using REST API parameters as per the following scenarios:

- For routing SBI messages directly to producer NFs, set `cbEnabled` to `true` in Routing Options REST APIs as described in "Configuring Routing Options" in *Oracle*

Communications Cloud Native Core, Service Communication Proxy REST Specification Guide.

- For routing notification messages directly to producer NFs, set `cbEnabled` to true in Routing Config Set REST APIs as described in "Routing Config Set" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
- For InterPLMN and InterSCP routing, set `cbScpEnabled` and `cbSeppEnabled` to true in System Options REST APIs as described in "Configuring System Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Configure

You can configure this feature using the CNC Console and REST APIs.

- **Configure using CNC Console:** For information about CNC Console configurations, see [Configuring Global Options](#) and [Circuit Breaking Configurations](#).
- **Configure using REST API:** For information about REST API configurations, see "Circuit Breaking Configurations", "Configuring System Options", "Routing Config Set", and "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.30 Canary Release

The Canary Release feature inspects the API version attribute of the NF service profile published by the NFs during NF registration or update. In case there is a new version of the API, Service Communication Proxy (SCP) identifies the version as a canary version if the version matches the configured value.

The older version is considered as the Production version and the new version of the service instance is considered as the Canary version. SCP distributes traffic between Production version and Canary version based on the operator configuration.

When the canary version's deployment is complete, the production version is upgraded to the canary version.

Managing Canary Release

Enable

You can enable the Canary Release functionality using the CNC Console or REST API.

- **Enable using CNC Console:** To enable the Canary Release feature, set the **Canary Release Flag** switch to **Enabled** as described in [Configuring Canary Release](#).
- **Enable using REST API:** To enable the Canary Release feature using REST API, set the `canaryReleaseFlag` parameter to `true` as described in "Configuring Canary Release Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Note

Canary Release must be enabled for each service.

Configure

You can configure the Canary Release functionality using the CNC Console or REST API.

- **Configure using CNC Console:** For information about how to configure Canary Release, see [Configuring Canary Release](#).
- **Configure using REST API:** For information about how to configure Canary Release, see "Configuring Canary Release Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.31 Static Configuration of NF Profiles

The Static Configuration Fallback feature provides a static configuration of NF profiles when georedundant NRFs in the network fail. Under failure conditions, when NRF is not configured, or when static configurable routing is preferred, Service Communication Proxy (SCP) uses the user configured or updated NF profiles. SCP retains the 5G topology information from NRF for further operations. It allows fallback to a static mode of operation for modifying the previously discovered NF producer profiles through NRF and adds any new NF producer profile while the NRFs are unavailable.

The following configurations are used for the Static Configuration Fallback feature:

- **TopologySourceInfo:** This is the configuration used by SCP to determine the 5G topology source. SCP learns and creates the routing rules accordingly.
- **5G NF topology information:** This information is available with SCP after learning the 5G topology from the defined source as per the previous section configuration either from NRF or a static method.
- You can create or modify the 5G NF profile information using REST APIs.

Managing Static Configuration

Enable

You can enable the Static Configuration feature by setting the `scpcconfiguration.defaultTopologySource` parameter to `LOCAL` in the `scpc` configuration section of the `values.yaml` file. For more information about this parameter, see *Oracle Communications Cloud Native Core, Cloud Native Environment Installation, Upgrade, and Fault Recovery Guide*.

Note

You can configure the required Helm parameters for this feature if you have performed a fresh installation of SCP or upgraded SCP within the same release version, for example, 1.15.1 to 1.15.1.

Configure

You can configure the Static Configuration feature using the CNC Console or REST API.

- **Configure using CNC Console:** For information about how to configure the Static Configuration feature, see [Configuring Topology Source Info](#).
- **Configure using REST API:** For information about how to configure the Static Configuration feature, see "Configuring NF Topology Groups" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.32 Preventive Audit on Last NF Instance

SCP determines the network topology from NRF by subscribing to each NF type. When a new NF instance is registered in the network, SCP receives a registration notification from the NRF and creates a routing rule based on the NF profile in the notification.

SCP removes the routing rule for an NF instance after receiving a deregistration notification from NRF.

When the primary NRF communication breaks, the producer NF registers with the configured secondary NRF. In case the primary NRF is not notified about this reregistration, then it sends a deregistration notification to SCP. This results in the removal of the routing rule at SCP. Also, it causes network failure if repeated for multiple NF instances.

With the introduction of the Preventive Audit on Last NF Instance feature, SCP can control NRF failure scenarios in the network by performing the preventive audit (discovery). This audit is performed with all the configured regional NRFs before removing the last NF instance of a particular NF type. If any NRF returns the success discovery response with NF profiles, SCP creates the routing rules based on the discovery response. SCP raises an alert to notify operators about such NRF failure scenarios.

Managing Preventive Audit on Last NF Instance

Enable

You can enable the Preventive Audit on Last NF Instance feature by setting the `preventiveAuditOnLastNFInstanceDeletion` parameter to `true` in the global parameters section of the `values.yaml` file. For more information about this parameter, see *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Note

You can configure the required Helm parameters for this feature if you have performed a fresh installation of SCP or upgraded SCP within the same release version, for example, 1.15.1 to 1.15.1.

Configure

There is no specific configuration required for the Preventive Audit on Last NF Instance feature.

Observe

There is no specific metrics and alerts required for the Preventive Audit on Last NF Instance feature.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.33 Rate Limiting

Service Communication Proxy (SCP) provides the Rate Limiting feature to:

- protect the network from consumer NFs that might overload the network with service requests.
- protect the producer NF instance from getting overloaded.

3.33.1 Local Rate Limit Feature

SCP uses the token bucket algorithm in the Rate Limit feature functionality to track the allowed and consumed tokens in a given time period as per rate limit configuration.

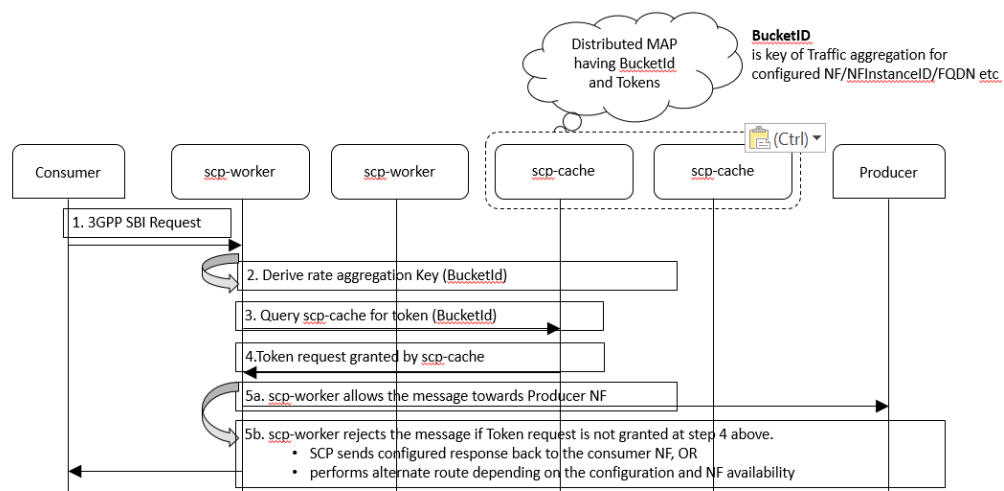
In the following ways SCP uses the distributed cache to aggregate the token consumption across scp-worker pods of an SCP instance:

1. Over a 1 second time period, the rate limit framework divides the configured rate into smaller slots of 100milliseconds each.
2. SCP monitors the difference between the actual request message rate received and the total request messages, i.e., allowed to send in the 100 milliseconds window as calculated.
3. Each SCP worker pod also calculates its own share of the total request message sent and accordingly decides on a batch of tokens to be preconsumed.
4. If at any time the scp-worker POD is not able to get a token, the request message is treated with rate limit treatment as per rate limit configuration.

Note

- The prediction of token batch count depends on the traffic rate detected in scp-worker PODs; hence, irregular traffic may lead to more token consumption by scp-workers. This may result in a difference between the overall message rate allowed and the configured rate.
- The rate limit framework expects a uniform traffic pattern over a second period for a configured rate.

Figure 3-8 Local Rate Limit Feature Functional View



3.33.2 Global Rate Limit Feature

The global rate limit feature aggregates traffic rate data across multiple SCP instances and applies traffic rate control based on the configured aggregated rate.

The aggregated rate is configured at each SCP instance in the following way:

1. Each SCP instance monitors its own traffic routed count against the configured NF, NFInstanceID, FQDN, and so on, and reports the same to all other SCP instances.
2. Each SCP instance aggregates the total traffic routed against the configured NF, NFInstanceID, and FQDN from all SCP instances.
3. Each SCP also calculates its own share of routed traffic in the aggregated total routed traffic count, as well as the status of the difference between total routed traffic and the configured aggregated rate.
4. Global Rate Limit Control actions such as allowing or disallowing traffic are decided based on the data collected in the steps above.

Note

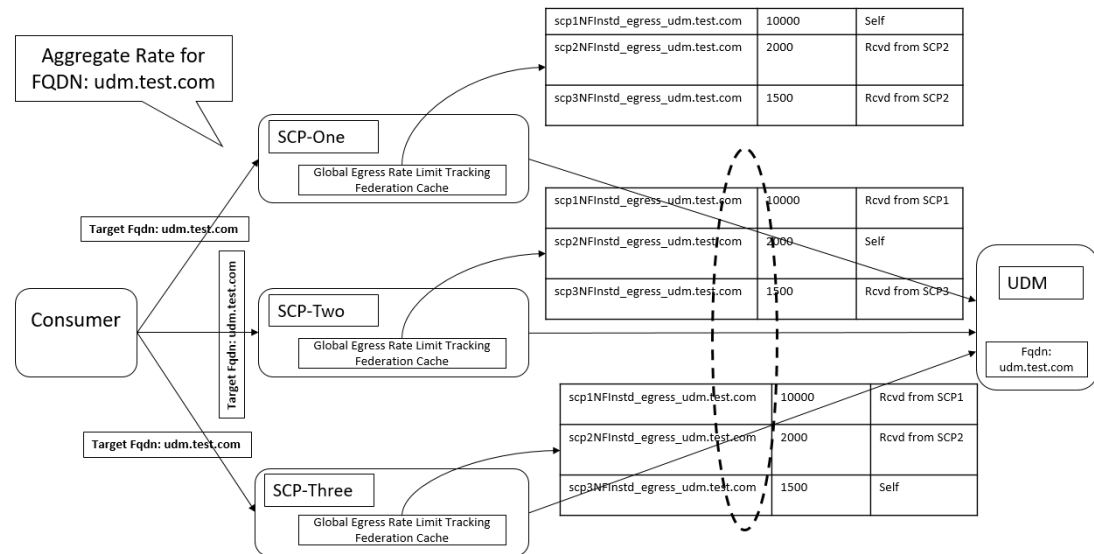
Convergence of global rate data aggregation depends on network latency among the SCP instances, and higher latency can impact rate-limiting performance.

Global Rate Working Principle

The working principle are as follows:

1. Each SCP instance has an aggregation cache, which syncs its own traffic route count with other SCP instances.
2. In this aggregation cache, a routed traffic rate count is maintained for each SCP instance.
3. Each SCP instance calculates the total traffic routed towards the configured NF, NFInstanceID, FQDN, and so on.
4. For example, the diagram depicts the aggregated routed traffic count for a FQDN, "udm.test.com," across all 3 SCP instances.
 - Sample configuration 1: A global rate limit is configured in 3 SCP instances for FQDN "udm.test.com" with a rate of 15000 per sec.
 - Sample configuration 2: A global rate limit is configured in 3 SCP instances for FQDN "udm.test.com" with a rate of 13000 per sec.
5. The global rate limit control action, such as allowing or disallowing the traffic, is decided based on the aggregated routed traffic count collected.
 - Action for configuration 1: Traffic is allowed since the total traffic routed rate is less than 15000 per sec.
 - Action for configuration 2: Traffic is throttled since the total traffic routed rate is greater than 13000 per sec.

Figure 3-9 Global Rate Limit Working Principle



SCP supports rate limiting at the ingress and egress side.

If the SCP-cache service is unavailable, then SCP-worker continues to process service requests, however, rate limiting is not applied. If any SCP-worker is unavailable, it cannot move to the Ready state until SCP-cache initializes the rate limiting.

Note

- The Rate Limit framework depends on the ingress traffic pattern at SCP, and its variation reflects in traffic rate detection and applies throttling of messages in a distributed environment since it involves convergence of message rate for a specific rate limit configuration, for example, a configured allowed rate value for a specific FQDN of an NF service instance across multiple scp-worker service pods.
- The average allowed traffic rate with rate control may be achieved as per the configured allowed traffic rate over a few seconds. The instantaneous allowed traffic rate with rate control may have some variation with respect to the configured allowed traffic rate on a short time scale.

The following sections explain how the SCP Rate Limiting feature controls ingress and egress traffic.

3.33.3 Ingress Rate Limiting

The Ingress Rate Limiting feature protects the network from unsecured consumer NFs that might overload the network with message requests. SCP secures the network when aggregated ingress traffic from any registered NF instance exceeds the allowed traffic rate limit. If the traffic exceeds the allowed traffic rate limit, SCP does not process the traffic and responds with an error code. Operators can configure the error code. Otherwise, the default error code 429 is set.

With the enhanced Ingress Rate Limiting functionality, SCP can use the User-Agent or XFCC header in the ingress message request to identify the consumer NF and apply ingress rate limiting at the ingress side.

The traffic rate limit is set at the NF type level that is applicable for all NF instances. SCP obtains the FQDN of consumer NF from the x-forwarded-client-cert (XFCC) header in the incoming message request. SCP provides the configuration for ingress "allowed requests/sec" per NF type. This configuration is applicable for all NF instances of the NF type.

As most of the network do not provide XFCC header in the ingress message requests, SCP cannot rely on the XFCC header to fetch consumer NFs' identity. Therefore, as recommended by 3GPP, the User-Agent header can be added to service request by the consumer NF to communicate its identity. The User-Agent header is added to each ingress message request irrespective of the network type. This header contains consumer NF identity in the different formats.

SCP supports the following types of User-Agent header formats to retrieve the consumer NF identity:

- **Format1:** NFTYPE-NFINSTANCEID FQDN or NFTYPE-NFINSTANCEID-FQDN
- **Format2:** NFTYPE-FQDN NFINSTANCEID or NFTYPE-FQDN-NFINSTANCEID
- **Format3:** NFTYPE-NFINSTANCEID
- **Format4:** NFTYPE-FQDN

Where,

- NFTYPE indicates the type of consumer NF.
- NFINSTANCEID indicates the instance ID of the consumer NF.
- FQDN indicates the FQDN of the consumer NF.

Using the aforementioned parameters of User-Agent header, SCP obtains the consumer NF information and applies ingress rate limiting at the ingress side. SCP uses the ConsumerInfo REST API parameter to determine the header type such as XFCC or User-Agent, in the ingress message request. For more information about the ConsumerInfo REST API parameter, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

SCP determines consumer NF identity such as NFTYPE, NFINSTANCEID, or FQDN, from the User-Agent header and applies the ingress rate limiting configurations in the following sequence:

1. NFINSTANCEID
2. FQDN
3. NFTYPE

If SCP determines any one of the configurations as per the order, SCP stops searching for the remaining configurations and uses the retrieved configurations for ingress rate limiting.

Example of Ingress Rate Limiting Configuration

You can configure ingress request rate for all NF instances of an NFType so that this configuration is applicable for all NF instances. You can configure MPS for a single NF instance identified by FQDN.

There are 10 Session Management Functions (SMFs) determined by SCP. The maximum traffic limit for all SMFs except for SMF3, SMF4, and SMF7 is 1000 MPS.

1. The maximum ingress requests limit from SMF3 is 600 MPS.

2. The maximum ingress requests limit from SMF4 with instance ID 26740918-e9cd-0205-aada-71a76214d33c is 200 MPS.
3. The maximum ingress requests limit from SMF7 is 400 MPS.

Table 3-26 Example of Ingress Rate Limiting Configuration

Exception	NFType	FQDN	NF Instance ID	MPS
NA	SMF	-	-	1000 MPS
1	-	smf3.example.com	-	600 MPS
2	-	smf7.example.com	-	400 MPS
3	-	-	26740918-e9cd-0205-aada-71a76214d33c	200 MPS

Note

The NF Instance ID parameter is used only in the User-Agent header.

Ingress Rate Limiting Process Using the User-Agent or XFCC Header Parameter

SCP uses the values of `primaryHeaderName` and `secondaryHeaderName` in the `ConsumerInfo` REST API parameter to identify the consumer NF in an ingress message request. When SCP receives an ingress message request, it searches for the `primaryHeaderName` value and follows the following methods to apply ingress rate limiting:

- If SCP finds `primaryHeaderName`, it considers the configured value such as XFCC header or User-Agent header, to identify the consumer NF and uses the configurations that match the consumer NF identity to apply ingress rate limiting. If SCP does not find any configurations for consumer NF, it does not apply ingress rate limiting.
- If SCP does not find `primaryHeaderName`, it searches for `secondaryHeaderName` and considers the configured value such as XFCC header or User-Agent header, to obtain the identity of the consumer NF and uses the configurations that match the consumer NF identity to apply ingress rate limiting. If SCP does not find any configurations for consumer NF, it does not apply ingress rate limiting.
- If SCP does not find `secondaryHeaderName`, it does not apply ingress rate limiting.

Managing Ingress Rate Limiting**Enable**

The Ingress Rate Limiting feature is a core functionality of SCP. This feature is automatically enabled in SCP at the time of deployment.

Configure

You can configure the Ingress Rate Limiting feature using the CNC Console or REST API.

- **Configure using CNC Console:** For information about configuring the Ingress Rate Limiting feature, see [Configuring Ingress Rate Limiter Consumer Info](#) and [Configuring Ingress Rate Limiting](#).
- **Configure using REST API:** For information about configuring the Ingress Rate Limiting feature, see "Configuring Ingress Rate Limiting" and "Configuring Consumer NF Info" in

Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.33.4 Egress Rate Limiting

The Egress Rate Limiting feature protects the registered producer NF instance from getting overloaded. With this feature, SCP secures the network when aggregated egress traffic towards an NF instance exceeds the allowed traffic rate limit.

If the traffic exceeds the allowed limit, then SCP performs alternate routing by default. In case alternate routing is not configured, SCP responds with an error code.

Operators can configure the error code. Otherwise, the default error code 503 is set.

To determine the destination of the request, SCP checks the API root and authority information in the request header. SCP also checks the service name and path to identify the service type in the request header. Considering this information, SCP provides the configuration for "egress allowed requests/sec" per NF type. This configuration is applicable for all NF instances of the NF type.

Example of Egress Rate Limiting Configuration

You can configure Messages Per Second (MPS) for all NF instances of an NFType so that this configuration is applicable for all NF instances. You can configure MPS for single NF instance identified by FQDN.

There are 4 UDMs determined by SCP. The maximum traffic limit for all UDMs with the following exceptions is 400 MPS:

1. The maximum traffic limit for UDM3 is 500 MPS.
2. The maximum traffic limit for SDM service for all UDMs is 200 MPS.
3. The maximum traffic limit for UMD1 UECM service is 100 MPS.
4. The maximum traffic limit for UDM4 UEAU service is 600 MPS.

Table 3-27 Example of Egress Rate Limiting Configuration

Exception	NFType	FQDN	Service Name	MPS
NA	UDM	-	-	400
1	-	Udm3.example.com	-	500
2	-	-	Nudm_sdm	200

Table 3-27 (Cont.) Example of Egress Rate Limiting Configuration

Exception	NFType	FQDN	Service Name	MPS
3	-	Udm1.example.com	Nudm_uecm	100
4	-	Udm4.example.com	Nudm_ueau	600

Managing Egress Rate Limiting

Enable

The Egress Rate Limiting feature is a core functionality of SCP. This feature is automatically enabled in SCP at the time of deployment. To allow rate limit to occur, you must enable either per NF rate limit or make entries in the Egress table using the CNC Console.

Configure

You can configure the Egress Rate Limiting feature using the CNC Console or REST API.

- **Configure using CNC Console:** For information about configuring the Egress Rate Limiting feature, see [Configuring Egress Rate Limiting](#).
- **Configure using REST API:** For information about configuring the Egress Rate Limiting feature, see "Configuring Egress Rate Limiting" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.33.4.1 Global Egress Rate Limiting

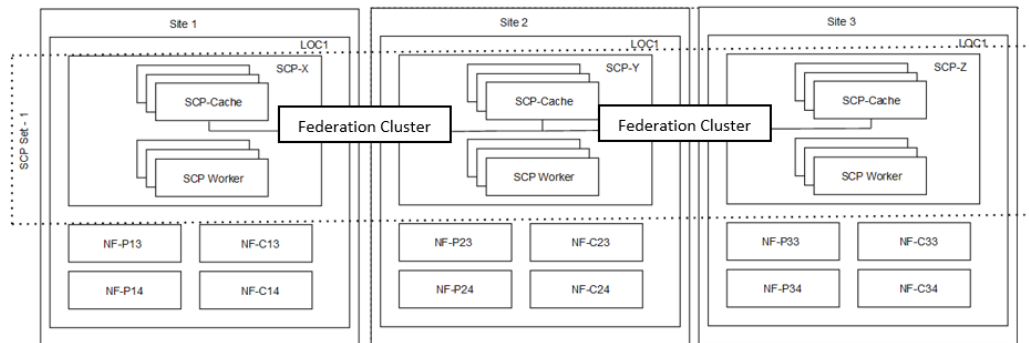
The existing egress rate limiting functionality aggregates traffic rate for a target producer NF within one SCP instance across multiple pods of SCP-worker service. It does not aggregate the overall traffic rate towards the producer NF across multiple SCP instances. Therefore, the Global Rate Limiting feature is implemented so that a group of SCP instances coordinates among each other to ensure traffic rates for common destinations, that is, producer NFs, are aggregated and the egress rate limit is applied to the overall traffic rate.

The Global Egress Rate Limiting feature synchronizes and aggregates the egress rate data for multiple producer NFs among multiple SCP instances (up to three SCP instances). Based on the aggregated rate data, each SCP controls its traffic to ensure the global configured egress rate is achieved.

The SCP-cache microservice synchronizes egress rate data among multiple SCP instances. The participating SCP instances for the egress rate limiting aggregation details are configured

using Helm as described in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*. SCP uses an Oracle product called Oracle Coherence which offers federated caching to link multiple clusters so that cache data is automatically synchronized between clusters.

Figure 3-10 Role of SCP-Cache



By default, the egress rate aggregation is disabled. After enabling the Global Egress Limiting feature using the `global_egress_ratelimit` parameter on the CNC Console, the egress rate aggregation uses `remoteScpOneEnabled` and `remoteScpTwoEnabled` parameters to enable rate data synchronization and aggregation with the corresponding remote instances of SCP. If the rate aggregation and synchronization are disabled or the communication fails with a remote SCP instance, the Global Egress Rate Limiting feature does not consider that remote SCP as part of the Global Federation cluster. Therefore, Global Rate Limit Algorithm does not consider any message rate related updates from that remote SCP while applying the rate limit.

Managing Global Egress Rate Limiting

Enable

You can enable this feature by setting `global_egress_ratelimit` to true. For more information, see [Configuring SCP Features](#).

Configuring Participating SCP for Global Egress Rate Limit

Perform the following configuration while deploying SCP.

1. If the following Helm parameters of the participating SCP instances are available, add the required Helm parameters in case a single instance of SCP is used or add all the parameters in case both the instances of SCP are used to the `custom-values.yaml` file:
 - `coherence.federation.remoteScpOne.fqdnOrIp`
 - `coherence.federation.remoteScpOne.port`
 - `coherence.federation.remoteScpOne.clusterName`
 - `coherence.federation.remoteScpOne.nfInstanceId`
 - `coherence.federation.remoteScpTwo.fqdnOrIp`
 - `coherence.federation.remoteScpTwo.port`
 - `coherence.federation.remoteScpTwo.clusterName`

- `coherence.federation.remoteScpTwo.nfInstanceId`
2. Deploy SCP as described in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.
 3. Enable global egress rate aggregation and control with the participating SCP instances using the CNC Console as described in [Configuring SCP Features](#).

Perform the following configuration if SCP is deployed.

1. Perform the same release upgrade using the following Helm parameters for the participating SCP instances in the `custom-values.yaml` file. Add the required parameters if one instance of SCP is used, otherwise add all the parameters if both the instances of SCP are used.
 - `coherence.federation.remoteScp0ne.fqdn0rIp`
 - `coherence.federation.remoteScp0ne.port`
 - `coherence.federation.remoteScp0ne.clusterName`
 - `coherence.federation.remoteScp0ne.nfInstanceId`
 - `coherence.federation.remoteScpTwo.fqdn0rIp`
 - `coherence.federation.remoteScpTwo.port`
 - `coherence.federation.remoteScpTwo.clusterName`
 - `coherence.federation.remoteScpTwo.nfInstanceId`

Only the `scp-cache` microservice restarts.

2. Enable global egress rate aggregation and control with the participating SCP instances using the CNC Console as described in [Configuring SCP Features](#).

Configure

You can configure this feature using Helm and REST API.

- **Configure using Helm:** For information about Helm configurations, see "global Parameters" in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Note

You can configure the required Helm parameters for this feature if you have performed a fresh installation of SCP or upgraded SCP within the same release version, for example, 1.15.1 to 1.15.1.

- **Configure using REST API:** For information about REST API configurations, see "Configuring Egress Rate Limiting and Global Egress Rate Limiting" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.33.5 Egress and Ingress Key Combinations

This section provides information about different key combinations and key formats required to configure ingress and egress rate limiting entries.

Operators can use these key combinations for aggregating traffic to apply rate limit. The key format is obtained from the JSON body of the Rate Limiting configuration request.

Note

Operators must perform the search in the same sequence as mentioned in the following tables.

Table 3-28 Egress Key Combination

Sequence	Key Combination	Key Format
1	FQDN & ServiceType	FQDN_ServiceType
2	FQDN & NFType	FQDN_NFType
3	FQDN	FQDN
4	ServiceType	NFInstanceId_serviceType Note: Look up NFInstanceId from Learnt NFProfiles using FQDN.
5	NFType	NFInstanceId Note: Look up NFInstanceId from Learnt NFProfiles using FQDN.

Table 3-29 Ingress Key Combination

Sequence	Key Combination	Key Format
1	NfInstanceId	NfInstanceId of the consumer NF.
2	FQDN	FQDN
3	NFType	NFInstanceId Note: Look up NFInstanceId from Learnt NFProfiles using FQDN.

3.34 Support for Multus Container Network Interface

Multus Container Network Interface (CNI) is a plug-in that allows you to connect multiple network interfaces to pods in Kubernetes.

Each Kubernetes pod contains only one network interface. After integrating Multus CNI with SCP, you can create multiple interfaces in a pod and perform the following tasks:

- Create multiple signaling interfaces for routing the signaling traffic
- Create different interfaces for Operations, Administration, and Maintenance (OAM) and Signaling (SIG) traffic to enable traffic segregation

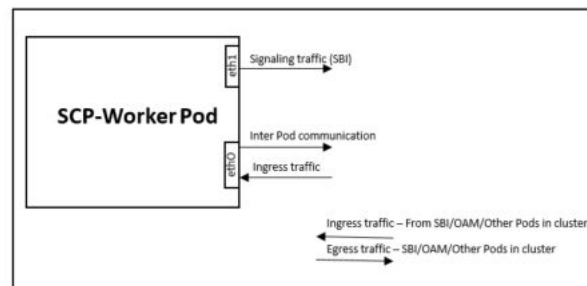
This feature configures the pods to use different network interfaces for different types of traffic. These interface IPs and routing information, such as range of IPs and gateways, are provided by Network Attachment Definition (NAD) Kubernetes custom resource. Multus CNI configures the IP routing rules based on this routing information.

For more information about Multus CNI, see <https://github.com/k8snetworkplumbingwg/multus-cni>.

Note

Default Pod IPs assigned by Kubernetes are used for establishing pod-to-pod and egress communication. When Multus CNI adds other network interfaces to pods, these interface IPs can be used for egress communication.

Figure 3-11 Networking-Multus Support



The diagram shows how support for Multus helps SCP to support multiple interfaces on egress traffic from the scp-worker pod.

Note

The Multus requirement is not applicable for SCP Control Pods.

The scp-worker pod is handling the following types of traffic:

- 5G SBI traffic
- Inter-pod communication traffic
- OAM traffic is communication traffic between observability components and scp-worker

The directions of this traffic are to and fro via SCP, and hence they are present in ingress and egress directions. There is no segregated ingress traffic. Only egress traffic is separated over multiple interfaces by dedicating Virtual network interface controllers (vNICs) to manage each type of traffic:

- Eth0 vNIC manages Egress (inter-pod communication traffic).

- Eth1 vNIC manages Egress 5G SBI traffic.

It is not required to separate OAM traffic from signaling traffic.

Managing Multus Container Network Interface

Enable

The Multus CNI plug-in can be integrated with Service Communication Proxy (SCP) at the time of SCP installation.

Configure

To use this feature, complete the configuration as described in "Configuring Multus Container Network Interface" in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

① Note

You can configure the required Helm parameters for this feature if you have performed a fresh installation of SCP or upgraded SCP within the same release version, for example, 1.15.1 to 1.15.1.

Observe

There is no specific metrics and alerts required for Multus Container Network Interface.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.35 Support for Model C Indirect 5G SBI Communication

5G Service Based Interface (SBI) communication models are used to establish connections among NFs and NF services. These communication models facilitate consumer NFs to route service requests to producer NFs either directly or indirectly through Service Communication Proxy (SCP). For more information about different communication models, such as Model A, Model B, Model C, and Model D, see 3GPP TS 23.501 version 16.6.0 Release 16.

With the introduction of the Model C Indirect 5G SBI Communication format in SCP, the consumer NF sends NF discovery service requests to NRF. After receiving the discovery response with NF profiles, the consumer NF does the following:

- Selects an NF Set or a specific NF instance from the NF Set
- Sends a service request to SCP with the address of the selected service producer NF
- Creates multiple network interfaces for different subnet IPs

Where,

- NF Set is a group of interchangeable NF instances of the same type that supports the same services and the same Network Slices. NF instances in the same NF Set may be geographically distributed but have access to the same context data.
- NF instance is an instance of the NF that can be identified.
- NF Set Identifier (NF Set ID) is a globally unique identifier of a set of equivalent and interchangeable CP NFs from a given network that provides distribution, redundancy, and scalability.

Then, SCP selects an NF service instance and routes the service requests to the selected producer NF. In case of failure if reselection is required, SCP selects the producer NF instance based on the NF Set. The reselection is based on the binding header, if available.

Support for Binding Headers in Model C Indirect 5G SBI Communication

SCP supports the following 5G SBI custom headers in Model C Indirect 5G SBI Communication:

- 3gpp-Sbi-Binding: This header contains a comma-delimited list of Binding Indications from an HTTP server for storage and use of HTTP clients. The absence of this parameter in a Binding Indication in a service request is interpreted as "callback".
- 3gpp-Sbi-Routing-Binding: This header contains a Routing Binding Indication to direct service request to an HTTP server which has the targeted NF service resource context.

SCP does not update the value of the "3gpp-Sbi-Binding" header in the HTTP/2 request or response. It supports reading value of the "3gpp-Sbi-Routing-Binding" header in the service request and uses the binding level in the "3gpp-Sbi-Routing-Binding" header to perform routing.

Note

Binding level NF service Set and NF service instance are not supported.

Support for Load Balancing and Alternate Routing in Model C Indirect 5G SBI Communication

When the target producer NF is unavailable, then SCP determines the NF Set ID of the target producer NF and performs alternate routing on the NF ID set instances in a sequence until SCP completes all the configured routing attempts.

SCP uses the NF set or NF instance for load balancing and alternate routing.

Support for Reverse Lookup in Model C Indirect 5G SBI Communication

When forward routing fails due to unavailability of the producer NF or the message does not have the NF Set ID, SCP performs reverse lookup in learned NF profiles. It performs alternate routing by obtaining destination information from the Set IDs. Reverse Lookup occurs when "3gpp-Sbi-Discovery-Target-Nfset" and "3gpp-Sbi-Routing-Binding" headers are absent in the message.

Model C Indirect 5G SBI Communication Response Headers

Additional response headers supported by Model C Indirect 5G SBI Communication in case of alternate routing are as follows:

- "3gpp-Sbi-Producer-Id"
- Either "3gpp-Sbi-Target-apiRoot" or Location header

Note

Load Balancing or Forward Routing from routing options can not be configured with Release 16 of Model C Indirect 5G SBI Communication because forward routing occurs based on the "3gpp-Sbi-Target-apiRoot" header. If the "3gpp-Sbi-Target-apiRoot" header is unavailable, then only load balancing occurs based on NF Set ID in the binding header.

Managing Model C Indirect 5G SBI Communication**Enable**

You can enable the Model C feature by adding `- rel16` under `releaseVersion` in the `ocscp-custom-values-<scp_release_number>.yaml` file. For more information about enabling this feature, see *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Note

You can configure the required Helm parameters for this feature if you have performed a fresh installation of SCP or upgraded SCP within the same release version, for example, 1.15.1 to 1.15.1.

Configure

You must complete the fetching routing information configuration as described in "Fetching Release 16 Routing Rules" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.36 Multiple SCP Deployment in the Same Kubernetes Cluster

This feature allows operators to deploy multiple SCP instances in the same Kubernetes cluster. Each SCP instance is deployed in a different namespace and is independent of each other. These SCP instances can use the same cnDBTier with a unique database name or a dedicated cnDBTier during the installation. `scpFqdn` is included as part of the SCP logs, metrics, and alerts to identify and filter logs, metrics, and alerts at Grafana or Kibana when multiple SCP instances are deployed in the same Kubernetes cluster.

Managing Multiple SCP Deployment in the Same Kubernetes Cluster

Enable

Not applicable.

Configure

There is no specific configuration required for this feature. The following new dimensions are added to the respective metrics in the [Metrics](#) section:

- scpFqdn (for data plane)
- scp_fqdn (for control plane)

Observe

There is no specific metrics and alerts required for this feature.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.37 Support for SCP Triplet Deployment

Using the Service Communication Proxy (SCP) Triplet Deployment model, operators can configure three SCP instances as mates for georedundancy. Each SCP instance in a triplet is configured to manage the overall traffic expected for the triplet so that when two SCP instances are unavailable, the single SCP instance can manage the overall traffic. For example, if the first and the second SCP instances are unavailable, then the third instance is dimensioned to manage the load of the first two instances.

With this feature, SCP can:

- configure two mated SCP.
- support forward routing, load balancing, and alternate routing to producers in its serving localities and serving localities of its mates.
- route the service request directly to any producer in its serving localities and serving localities of its mates.
- support inter-SCP routing to producers which are not in its serving localities and serving localities of its mates.
 - When SCP serving a particular locality is unavailable, then service requests are routed to mated SCPs based on priority.
 - When multiple SCPs with the same priority are serving the producer's locality, SCP supports load balancing.

Managing SCP Triplet Deployment

Enable

Not applicable.

Configure

To use this feature, complete the configuration as described in "global Parameters" in *Oracle Communications Cloud Native Core, Cloud Native Environment Installation, Upgrade, and Fault Recovery Guide*.

Note

You can configure the required Helm parameters for this feature if you have performed a fresh installation of SCP or upgraded SCP within the same release version, for example, 1.15.1 to 1.15.1.

Observe

For information about metrics, see [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.38 Alternate Routing based on the DNS SRV Records

Using the Alternate Routing based on the Domain Name System (DNS) Service (SRV) Records feature, SCP performs alternate routing by querying SRV records with the DNS server to find an alternate producer NF and route the service request.

Note

- SCP uses either NF Set or DNS SRV Records to perform alternate routing.
- This feature is supported only in SCP Release 16 deployment with Model C Indirect 5G SBI Communication. For more information about Release 16, see 3GPP TS 23.501.

SCP performs the DNS SRV query for the FQDNs determined from NF profiles to find an alternate NF instance and route the services requests to the newly discovered alternate producer NF. These FQDNs are sourced from Network Repository Function (NRF) notifications received at SCP and from operator's configuration, which is statically configured through *scpc-configuration* as described in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

The DNS SRV Resource record known as Service Proto Name (SPN) is used to identify the target NF type. If there are multiple instances of the same alternate NF profiles in a DNS resolution response for handling the same service request, then the appropriate target producer NF is selected using priority and weight attributes provided in the *DNSSRVRecord*

data model as described in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

This feature introduces the `scpc-alternate-resolution` microservice to perform the following tasks:

- Find alternate NF services by sending DNS SRV queries to the DNS server and receive the DNS SRV responses.
- Refresh DNS SRV records to get update in DNS SRV records.
- Provide DNS SRV records to `scp-worker` when required.

DNS SRV Default Scheme

SCP obtains FQDNs from the NF profiles and creates alternate routing records for each FQDN. Alternate routes are obtained by querying the configured DNS server. To query the DNS server, SCP creates an SPN entry for each FQDN by determining the scheme, such as `http` or `https`, for each FQDN.

Example of an SPN entry is as follows:

```
_http._tcp.udm1svc.scpsvc.svc.cluster.local
```

In an NF profile, FQDNs are available at the profile level and service level. For service level FQDNs, the scheme is defined at the service level. However, for profile level FQDNs, the scheme is not defined in the profile. Earlier, SCP used "`http`" as the default scheme for profile level FQDNs when creating SPNs. However, for operators who use "`https`", this results in not getting alternate routes for the profile level FQDN from the DNS server.

Therefore, the following Helm configuration is introduced to determine the scheme to be used for SPN creation for profile level FQDNs:

```
dnsSrvSchemeConfig:  
  defaultScheme: "https"  
  exceptionList: [""]
```

`defaultScheme` is the default scheme used for creating Domain Name System (DNS) Service (SRV) Service Proto Name (SPN) from NF profile level FQDN. SCP also supports per NF type scheme selection. This is achieved by `exceptionList`, which defines the NF types that use a non-default scheme. For example, if `defaultScheme` is "`https`" and `exceptionList` has ["UDM", "AUSF"], then SPNs created from profile level FQDNs of UDM and AUSF will have "`http`" as the scheme, and all other NF types will have "`https`" as the scheme. For more information about `defaultScheme` and `exceptionList`, see "Global Parameters" in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Note

The default value in the Helm configuration indicates that the previously used value of "`http`" has been changed to "`https`".

Managing Alternate Routing based on the DNS SRV Records

Enable

This feature is enabled by setting the `dnsSRVAlternateRouting` parameter to `true` in the `custom-values.yaml` file. For more information, see *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Note

You can configure the required Helm parameters for this feature if you have performed a fresh installation of SCP or upgraded SCP within the same release version, for example, 1.15.1 to 1.15.1.

Configure

You can configure the Alternate Routing based on the DNS SRV Records feature using Helm, REST API, and the CNC Console.

- **Configure using Helm:**
 - For information about configuring `defaultScheme` and `exceptionList`, see "Global Parameters" in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.
 - For information about configuring the `scpc-alternate-resolution` microservice parameters, see "scpc-alternate-resolution Parameter" in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.
- **Configure using CNC Console:** For information about the **Routing Options** tab to enable alternate routing configuration and set alternate route modes, see [Configuring DNS SRV based Alternate Routing](#).
- **Configure using REST API:** For information about the `alternatenfgroup` parameter to manage the alternate NF group configuration data that is configured for the corresponding query parameters, see "Configuring Alternate NF Group" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.39 Support for Multiple Signaling Service IPs

In the previous release, SCP supported only one signaling IP from a given network. This IP was exposed to all NFs to establish 5G Service Based Interface (SBI) communications.

With the support for multiple signaling IPs, SCP can expose multiple signaling service IPs from the same or a different network to all NFs to accept 5G SBI signaling traffic (ingress connections).

Note

SCP supports 8 signaling IPs.

This feature provides the flexibility to achieve maximum redundancy in the IP path. 5G NFs can have more than one SCP's destination IPs for the transport network to select different paths for each of the IPs. Therefore, a single SCP instance is available through more than one IP path if one of the transport network paths fails.

Using this feature, you can do the following:

- Improve resiliency in 5G core network during transport failure by enabling multiple signaling paths.
- Create dedicated signaling service paths by configuring multiple service signaling IPs.

Managing Multiple Signaling IPs

Enable

This feature is enabled automatically in SCP at the time of deployment.

Configure

The `serviceSpecifications` Helm parameter is introduced as part of this feature. To use this feature, complete the configuration as described in "Configuration Parameters" in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Note

You can configure the required Helm parameters for this feature if you have performed a fresh installation of SCP or upgraded SCP within the same release version, for example, 1.15.1 to 1.15.1.

To add or remove signaling services, see *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Observe

For information about metrics, see [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.40 Egress Congestion Control Support at SCP

When Service Communication Proxy (SCP) receives a 5G Service Based Interface (SBI) message request from a consumer NF, SCP forwards the message request to a target producer NF. If the producer NF is overloaded, then SCP performs alternate routing to find another equivalent producer NF. In case no equivalent producer NF is available, then SCP sends an error response to the consumer NF with the configured error code. For more information about configuring error codes, see [Configuring SBI Message Priority and Error Profiles for Egress Congestion Control](#).

Support for 3GPP SBI Message Priority

With the introduction of this feature, SCP uses the SBI message priority value present in the "3gpp-Sbi-Message-Priority" header to categorize a 5G SBI message request as high or low priority. The message request is prioritized based on the value of the "3gpp-Sbi-Message-Priority" header. By default, SCP considers 24 as the priority for the ingress message request if the message does not have the "3gpp-Sbi-Message-Priority" header. If the priority of the ingress message request is less than the value of the SBI Message Priority Discard From parameter, which is defined in [Configuring SBI Message Priority and Error Profiles for Egress Congestion Control](#), then these messages are considered as high priority messages.

When SCP receives a high priority message request and the target producer NF is overloaded, SCP performs alternate routing to find an equivalent producer NF. If no equivalent producer NF is available, then SCP forwards the high priority message to the same target producer NF that is overloaded.

When SCP receives a low priority message request, SCP forwards the message request to a target producer NF. If the producer NF is overloaded, then SCP performs alternate routing to find another equivalent producer NF. In case no equivalent producer NF is available, then SCP sends an error response to the consumer NF with the configured error code.

Egress Congestion Control for Producer NF

The following subsections describe the egress congestion control for producer NFs.

Egress Congestion Control based on Busy Producer NF

SCP, based on the response status codes 503, 429, and 307 with the Retry-After header in the response, removes the producer NF from its routing list for a period specified in the Retry-After header. This behavior is applicable if the Retry-After header is present in the response.

Egress Congestion Control based on Reported Producer NF Service Load

The NRF reports the load of a producer NF through a notification. After detecting that the producer NF load is at or above the alternate routing onset threshold but below the throttle onset threshold, SCP attempts to alternate route message requests designated for the producer NFs to other NFs wherever possible.

If message requests cannot be alternate routed, they are routed to the producer NF. The behavior continues until the producer NF load reported decreases below the abatement threshold. If the load reported by the NF exceeds the throttle onset threshold, all requests are alternate routed wherever possible and if requests cannot be alternate routed, they are rejected with a configurable error response code. SCP supports a configurable error response code that is to be sent when requests are being throttled and cannot be alternate routed. The default code is 500. For more information on utilizing LCI for obtaining producer load, refer the [Load Control based on the Load Control Information Header](#) section.

Note

Currently, if load is not present at service level, it is not picked from the NF profile level.

Egress Congestion Control for Reported Load of NextHop SCP or SEPP

SCP protects the next-hop SCP or SEPP from congestion. SCP monitors the load reported by nexthop SCP or SEPP in the LCI header. When the load exceeds the threshold, congestion control steps are taken. Alternate routes or traffic throttling are utilized based on the request's priority.

Managing Egress Congestion Control Support at SCP**Enable**

This feature is enabled automatically in SCP at the time of deployment.

Configure

You can configure this feature using the CNC Console or REST API.

- **Configure using CNC Console:** For information about SBI message priority and error profile configurations, see [Configuring SBI Message Priority and Error Profiles for Egress Congestion Control](#).
- **Configure using REST API:** For information about message priority discard threshold and error code configurations in the `nfServiceLoadBasedCongestionControl` parameter, see "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at the system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.41 Support for Model D Indirect 5G SBI Communication with Delegated Discovery

5G Service Based Interface (SBI) communication models are used to establish connections among NFs and NF services. These communication models facilitate consumer NFs to route service requests to producer NFs either directly or indirectly through Service Communication Proxy (SCP). For more information about different communication models, such as Model A, Model B, Model C, and Model D, see 3GPP TS 23.501 version 16.6.0 Release 16.

Using the Model D Indirect 5G SBI Communication with Delegated Discovery feature, consumer NFs delegate the producer NF discovery procedure to SCP by adding discovery parameters to the service requests.

SCP performs NF discovery with NRF using the received discovery parameters. The NRF indicated in the `preferredNrfForOnDemandDiscovery` parameter is used for the discovery procedure, which specifies the NRF preferred by `scp-nrfproxy` for delegated discovery. When the indicated NRF is not available, SCP does alternate routing to the `NRFSet`, which has the `preferredNrfForOnDemandDiscovery` parameter.

When SCP receives service requests with NF discovery parameters by including "3gpp-Sbi-Discovery" headers for the target producer NF selection, then SCP:

- extracts "3gpp-Sbi-Discovery" headers from the ingress service requests and initiates the discovery procedure based on these headers. For more information about the "3gpp-Sbi-Discovery" header, see 3GPP TS 29.500 version 16.6.0.
- selects the producer NF instance from the discovery response based on priority and capacity.
- stores NF discovery responses using the cache-control directives from NRF in NF discovery response messages.
- removes all "3gpp-Sbi-Discovery" headers from the service requests and forwards the service requests to the selected target producer NF. In case of inter-SCP routing, SCP forwards all the "3gpp-Sbi-Discovery" headers to the next hop SCP.
- reselects with the use of the discovery headers and results from delegated discovery responses are used to do initial selection and reselection.

Service Specific Discovery Request in Model D

The Model D discovery request that SCP sends to NRF has the `3gpp-sbi-discovery-service-names` header. The path URI of the ingress message is used to extract the service name, such as `nudm_uecm`. As a result, the rule creating takes less time since SCP just needs to parse a smaller NF profile. The NF profile returned by NRF contains the requested NF service-specific information only.

NF Selection based on Preferred Locality

A consumer NF selects the target producer NF closest to it by setting the preferred locality to the value of the locality of the producer NF. When the preferred locality query parameter is present in the NF discovery service request, NRF prefers NF profiles with a locality attribute that matches the preferred locality. NRF returns additional NFs in the response that does not match the preferred target NF location. NRF sets a lower priority for any additional NFs on the response that does not match the preferred target NF location than those matching the preferred target NF location.

SCP configures preferred locality query parameters based on the received "3gpp-Sbi-Discovery-preferred-locality" header in the service request. SCP assigns its own locality as a preferred-locality for NF discovery if the `assignPreferredLocality` parameter is enabled and the "3gpp-Sbi-Discovery-preferred-locality" header is absent in ingress service requests. SCP overrides the "3gpp-Sbi-Discovery-preferred-locality" header received in the service request and uses its own locality as preferred locality for NF discovery if the `overridePreferredLocality` parameter is enabled and the "3gpp-Sbi-Discovery-preferred-locality" header is present in ingress service requests. SCP controls the forwarding of assigned or overridden value of "3gpp-Sbi-Discovery-preferred-locality" headers to next hop SCP.

Model D Indirect 5G SBI Communication supports the following routing functionalities as described in [Support for Model C Indirect 5G SBI Communication](#) :

- Binding Headers
- Load Balancing and Alternate Routing
- Reverse Lookup

Load Balancing Support for Delegated Discovery

Users can change the preferred NRF instance ID or NRF Set ID for delegated discovery at runtime using the REST API. If only NRF nfSetId is configured, then SCP does load balancing for the matching NRFs selected by nfSetId. For more information, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Managing Model D Indirect 5G SBI Communication

Enable

You can enable the Model D feature by setting modeld_routing to true. For more information, see [Configuring SCP Features](#).

Note

You can enable this feature if SCP is deployed with Release 16.

Configure

You can configure this feature using Helm, REST API, and the CNC Console.

- **Configure using Helm:** For information about configuring the scp-nrfproxy microservice parameters, nrfProxyService, and nrfProfiles.customInfo.preferredNrfForOnDemandDiscovery, see *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Note

You can configure the required Helm parameters for this feature if you have performed a fresh installation of SCP or upgraded SCP within the same release version, for example, 1.15.1 to 1.15.1.

- **Configure using CNC Console:** For information about configuring this feature, see:
 - [Configuring SCP Features](#) to enable the feature and cache discovery responses.
 - [Configuring Global Options](#) to verify the **Forward Nf Discovery Headers** status.
 - [Configuring Routing Options](#) to manage the "3gpp-Sbi-Discovery-preferred-locality" header.
- **Configure using REST API:** For information about the REST API configurations, see "Configuring System Options" and "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about metrics, see [Metrics](#).

Maintain

If you encounter alerts at the system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.41.1 Managing Delegated Discovery Response Caching Based on Discovery Query Parameters

In Model D, the responsibility of NF discovery has been delegated to the Service Communication Proxy (SCP). When Model D cache is enabled at SCP, SCP caches all the NF discovery responses received from NRF. These cached responses are then used for further Model D delegated discovery service requests, where applicable.

Currently, when caching is enabled at SCP, all NF discovery responses received from NRF are cached by SCP. The service request often contains query parameters that change frequently, such as subscriber IDs, for example, Subscription Permanent Identifier (SUPI), Global Permanent Subscriber Identity (GPSI), and so on. This dynamic nature of query parameters may prevent the reuse of cached responses at SCP. Since SCP maintains a defined cache size, frequent cache filling could lead to inefficient utilization of the discovery response cache.

With this feature enhancement, SCP can do selective caching of the discovery responses based on the user-configurable discovery query parameter list, which will be avoided or ignored for discovery response caching at SCP. Thus, the selective caching control helps the user reduce the risk of cache overflow. This caching mechanism helps to optimize the performance while reducing latency.

Managing Delegated Discovery Response Caching

Prerequisite

This feature will be enabled if Model D feature and caching in Model D are both enabled.

Enable

You can enable the Managing Delegated Discovery Response Caching Based on Query Parameters feature by setting `modeld_routing` to true and `Caching` is set to true to enable caching to store discovery responses. For more information, see [Support for Model D Indirect 5G SBI Communication with Delegated Discovery](#).

Configure

You can configure this feature using REST API and the CNC Console.

- **Configure using CNC Console:** For information about CNC Console configurations, see
 - [Get NF Discovery Response Cache Configuration Data](#)
 - [Configuring NF Delegated Discovery Response Cache](#)
- **Configure using REST API:** For information about configuring REST APIs, see the "Discovery Cache Response Configuration" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

The following metrics are added in the [Metrics](#) section:

- ocscp_worker_ondemand_discovery_response_cache_skipped_total
- ocscp_worker_ondemand_discovery_response_cache_added_total
- ocscp_worker_ondemand_discovery_response_cache_removed_total
- ocscp_nrfproxy_ondemand_discovery_response_cache_skipped_total
- ocscp_nrfproxy_ondemand_discovery_response_cache_added_total
- ocscp_nrfproxy_ondemand_discovery_response_cache_removed_total

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

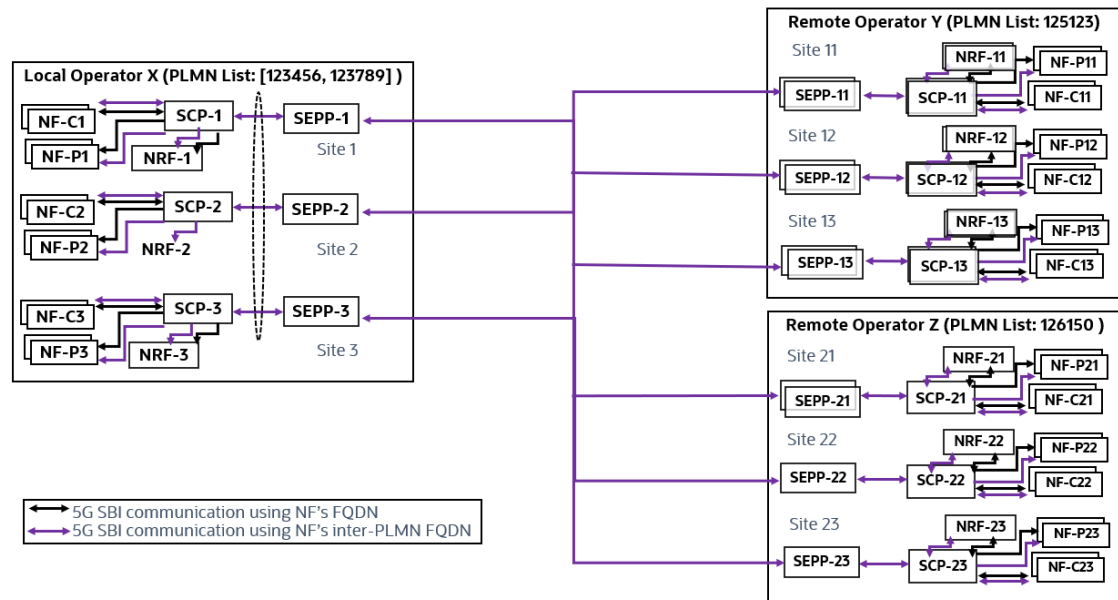
In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.42 Support for 5G SBI Roaming

3GPP specifies Local Breakout (LBO) and Home Routing (HR) roaming architectures. SCP can be used for indirect communications such as Model C or Model D between NFs within Visited Public Land Mobile Network (VPLMN) or Home Public Land Mobile Network (HPLMN) or within both VPLMN and HPLMN.

The following image depicts the 5G system roaming architecture in the case of a home routed scenario with service-based interfaces within the Control Plane. For more information, see 3GPP TS 23.501 v16.6.0.

Figure 3-12 Local and Remote Routing

Local PLMN consumer NFs can be V-SMF, V-AMF, or V-PCF, and so on. Remote PLMN producer NFs can be H-SMF, H-UDM, or H-PCF, and so on.

This feature enables 5G SBI indirect communication for 5G SBI roaming signaling traffic, which includes roaming ingress traffic to local operator's network and egress traffic to remote operator's network.

The PLMN network ID for a mobile network identifies the roaming location uniquely. The ID contains Mobile Country Code (MCC) and Mobile Network Code (MNC) to identify a mobile network operator.

If an NF wants to be discovered by other NFs in a different PLMN, it registers inter PLMN FQDN in NRF. The format of inter PLMN FQDN is as follows:

<labels separated by dots>.5gc.mnc<MNC>.mcc<MCC>.3gppnetwork.org

Example of of Inter PLMN FQDN UDM belonging to PLMN MNC 12 and MCC 345:

udm.5gc.mnc012.mcc345.3gppnetwork.org

Note

Telescopic FQDN is not supported in this release.

Intra-PLMN (non-roaming) traffic is routed based on 5G NF topology determined from NRF using FQDN (local) present in NFs. 5G SBI roaming traffic received from SEPP, that is, ingress to local operator network site and with inter-PLMN or self constructed FQDN, is routed using inter-PLMN FQDN present in NFs.

If SCP receives a "3gpp-Sbi-Target-apiRoot" HTTP header in the ingress service request, it determines the target PLMN from the inter PLMN FQDN specified in this header. If the "3gpp-Sbi-Target-apiRoot" HTTP header is not present, SCP performs delegated discovery for NF selection based on the received "3gpp-Sbi-Discovery" headers.

To support the 5G SBI roaming subscriber inter-PLMN traffic routing for different remote PLMNs, SCP identifies whether the ingress 5G SBI service request is for local PLMN IDs or remote PLMN IDs. To differentiate between intra-PLMN and inter-PLMN service requests, SCP uses local supported PLMN IDs.

For a network, local supported PLMN IDs include SCP supported PLMN IDs. Other than locally supported PLMN IDs, all PLMN IDs are considered as remote PLMN IDs by SCP. A network consists of multiple PLMN IDs, and network deployment is based on PLMN IDs.

SCP determines the SEPP topology from NRF or locally configured SEPP profile at SCP. SCP selects a SEPP instance based on PLMN serving the target NF. SCP performs load balancing and alternate routing between SEPP instances based on priority and capacity available in the SEPP profile.

To use InterPLMN routing with Oracle NRF, you must enable the `scp_user_agent_info` REST API parameter to add the "User-Agent" header in the format starting with the keyword "SCP-" so that SCP originated messages towards NRF include the "User-Agent" header required by Oracle NRF.

Managing Support for 5G SBI Roaming

Enable

You can enable this feature by setting `interplmn_routing` to true. For more information, see [Configuring SCP Features](#).

Configure

You can configure this feature using Helm, REST API, and the CNC Console.

- **Configure using Helm:** For information about Helm configurations, see *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Note

You can configure the required Helm parameters for this feature if you have performed a fresh installation of SCP or upgraded SCP within the same release version, for example, 1.15.1 to 1.15.1.

- **Configure using CNC Console:** For information about configuring NF instance IDs, see [Configuring SEPP Inter PLMN Routing](#).
- **Configure using REST API:** For information about the REST API configurations, see "Configuring SCP Features" and "Configuring SEPP InterPlmn Info" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.43 Routing Options Enhancements for Next Hop SCP and Notification or Callback Messages

The existing configuration of routing options supports 3GPP defined 5G NF services types, such as nudm-uecm, nudm-sdm, nsmsf-sms, and so on. However, there is no routing option defined for the following:

- 5G SBI notification or callback requests
- Inter-SCP or next hop SCP routing

With the introduction of this feature, the existing 5G SBI message routing option is enhanced to support:

- 5G SBI notification or callback message requests: SCP uses the "3gpp-sbi-callback" header to identify whether a 5G SBI request is a notification or callback.
- Inter-SCP or next hop SCP routing: When a consumer SCP routes a 5G SBI service request to the next hop SCP, the consumer SCP must have configurable routing options for SCP to SCP interface for each 3GPP defined 5G NF Service types, such as nudm-uecm, nudm-sdm, nsmsf-sms, and so on, if more than one SCP instance is involved in routing the 5G SBI service request. A new parameter known as nextHopSCP is introduced to the existing routing configurations to support this functionality.

This feature provides flexibility to configure inter-SCP or next hop SCP routing options for all services.

SCP derives an incoming message type from both "3gpp-sbi-callback" header and servicename in path to identify if it is a notification or callback request, or service request.

SCP uses the nextHopSCP routing configuration when routing messages towards the next hop SCP. The nextHopSCP routing parameters are set in SCP to define the message processing time to get a response from the producer NF instance in the same network or from another SCP instance in a different region.

Identification of 5G SBI Notification or Callback Messages

SCP identifies whether the ingress message request is a notification or callback, or service specific request based on the "3gpp-sbi-callback" header and the servicename in path present in the Uniform Resource Identifier (URI).

The following table describes different scenarios to identify a message request based on the availability of the "3gpp-sbi-callback" header and servicename in path.

Table 3-30 Identifying a Message Request

3gpp-sbi-callback	servicename in path	Description	Expected Routing at SCP
Present	Absent	The message request is considered as a 5G SBI notification or callback. Routing option configured for "5g-sbi-notification" is applicable. For more information about "5g-sbi-notification", see 3GPP TS 29.500 v16.6.0.	The expected notification message routing is as follows: • If the 3gpp-Sbi-Routing-Binding header is present, then the alternate routing as per Model C, Model D, and interPLMN traffic routing feature support at SCP. • If an alternate routing configuration is configured at SCP, then the alternate routing as per DNSSRV or route group configuration is supported at SCP. • If both alternate notification endpoint selection information is unavailable, SCP cannot perform notification message alternate routing.
Present	Present	The message request is considered as a 5G SBI notification or callback. Routing option configured for "5g-sbi-notification" is applicable.	The expected notification message routing is as follows: • If the 3gpp-Sbi-Routing-Binding header is present, then the alternate routing as per Model C, Model D, and interPLMN traffic routing feature support at SCP. • If an alternate routing configuration is configured at SCP, then the alternate routing as per DNSSRV or route group configuration is supported at SCP. • If both alternate notification endpoint selection information is unavailable, SCP cannot perform notification message alternate routing.
Absent	Absent	The message request is considered as a 5G SBI notification or callback. Routing option configured for "5g-sbi-notification" is applicable.	The expected notification message routing is as follows: • If the 3gpp-Sbi-Routing-Binding header is present, then the alternate routing as per Model C, Model D, and interPLMN traffic routing feature support at SCP. • If an alternate routing configuration is configured at SCP, then the alternate routing as per DNSSRV or route group configuration is supported at SCP. • If both alternate notification endpoint selection information is unavailable, SCP cannot perform notification message alternate routing.

Table 3-30 (Cont.) Identifying a Message Request

3gpp-sbi-callback	servicename in path	Description	Expected Routing at SCP
Absent	Present	The message request is considered as a 5G SBI service specific request. Routing option configured to specific 5G NF service is applicable.	The expected service request message routing is as follows: • If the 3gpp-Sbi-Routing-Binding header is present, then the alternate routing as per Model C, Model D, and interPLMN traffic routing feature support at SCP. • If an alternate routing configuration is configured at SCP, then the alternate routing feature as per DNSSRV or route group configuration is supported at SCP. • If both alternate notification endpoint selection information is unavailable, SCP will be unable to perform alternate routing based on reverse lookup feature support at SCP.

Managing Routing Options Enhancements for Next Hop SCP and Notification or Callback Messages

Enable

This feature is enabled automatically at the time of SCP deployment.

Configure

You can configure this feature using REST API and the CNC Console.

- **Configure using REST API:** For information about the REST API configurations, see "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
- **Configure using CNC Console:** For information about configuring NF instance IDs, see [Configuring the nextHopSCP Routing Option](#).

Observe

For information about metrics, see [Metrics](#).

Maintain

If you encounter alerts at the system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.44 Support for Alternate Routing Using Static Configuration

5G Service Based Interface (SBI) communication models establish connections among NFs and NF services. These communication models facilitate consumer NFs to route service requests to producer NFs directly or indirectly through Service Communication Proxy (SCP).

If the target producer NF is unavailable, SCP determines the NF Set ID of the target producer NF and performs alternate routing on the NF Set ID instances in a sequence until SCP completes all the configured routing attempts. In the absence of NF Set, SCP allows operators to configure FQDNs and use the Domain Name System (DNS) Service (SRV) method to find the equivalent NFs. In case the DNS SRV method is not availed by operators, then static configuration is used for alternate routing.

With the introduction of the static configuration, SCP allows operators to configure a pair or combination of primary NFs and their alternate NFs to define alternate producer NFs, which can be used in alternate NF selection when routing attempt to the primary NF fails.

The static alternate route feature prioritizes the settings in the static routing configuration over any NF profile for the alternate destination.

Managing Support for Alternate Routing Using Static Configuration

Enable

By default, this feature is disabled. To enable alternate routing using static configuration, you must complete the configuration as described in the following configuration section.

Configure

You can configure this feature using REST API and the CNC Console.

- **Configure using CNC Console:** For information about CNC Console configurations, see [Configuring Routing Options](#) and [Configuring Route Groups](#).
- **Configure using REST API:** For information about configuring REST APIs, see "Configuring Route Groups" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.45 Support for Thread Watchdog

The Thread Watchdog feature enables SCP to monitor whether any SCP microservice threads are stuck or nonresponsive at regular intervals. If the thread watchdog reports any thread as nonresponsive or stuck, then the pod's liveness fails. If the pod's liveness failure count

exceeds the failure threshold value, then the pod is restarted. If a thread doesn't respond within a specified duration, it is marked as nonresponsive or stuck. This time interval is configurable.

The failure threshold for the thread watchdog is also configurable. It determines how often the watchdog attempts to retry in case the watchdog detects a thread stuck or nonresponsiveness before declaring whether a thread is stuck or nonresponsive. Pod liveness fails when the watchdog failure count exceeds the configured threshold. These values are tuned and are not recommended to be changed.

Managing Support for Thread Watchdog

Enable

This feature is enabled automatically in SCP at the time of deployment. To disable this feature, set the `enableThreadWatchDog` parameter to `false` as described in "Configuring SCP Services" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Configure

You can configure this feature using REST API parameters as described in "Configuring SCP Services" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. To view thread watchdog configurations on the CNC Console, see [Configuring SCP Services](#).

Observe

There is no specific metric and alert required for this feature.

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.46 Message Feed Enhancements

The Message Feed feature allows copying of both request and response messages routed through SCP toward Oracle Communications Network Analytics Data Director (OCNADD) based on the configuration at SCP. Along with request and response messages, metadata, as described in [Table 3-31](#), is also sent to OCNADD.

The Message Copy feature provides SSL and SASL as a security mechanism for encryption and authentication. SSL can be configured for encryption, and it also serves as two way authentication between the scp-worker and the Kafka broker. SASL can be configured to achieve authentication of connections to brokers from scp-worker. Configuration of both SSL and SASL can be done on SCP with secret creation and specifying them under the `ddSslConfiguration` and `ddSaslConfiguration` sections of the SCP deployment file.

From SCP 23.4.1, SCP is enhanced to select Kafka partitions based on the `kafkaPartitionSelectionLogic` REST API configuration that provides the following methods to distribute the messages:

- **RoundRobin:** All the messages are distributed across all the available partitions in a round-robin order.

- **KeyBased:** Messages with the same correlation-id are routed to the same partition. This ensures the same transaction messages are routed to the same partition. For more information about `kafkaPartitionSelectionLogic`, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

OCNADD aggregates the data for monitoring and analytical purposes. For more information about OCNADD, see *Oracle Communications Network Analytics Data Director User Guide*.

The following table describes metadata sent to OCNADD:

Table 3-31 Metadata Table

Metadata	Description
correlation-id	Identifies transactions such as ingress downstream request, all corresponding upstream requests and responses, and downstream response.
consumer-id	Indicates the 5G NF instance ID of the originator NF, which is retrieved from the User-Agent header in the received request message. Conditions: • For SCP initiated request and response messages, for example, delegated Discovery and OAuth access token requests, consumer-id metadata must be present, and the value should be the same as the feed-source->nf-instance-id metadata value. Note: As this is an SCP instance-id, there should not be any dependency on the User-Agent header to retrieve the nf-instance-id information. • For consumer NF initiated messages, consumer-id metadata depends on the presence of the User-Agent header in the received service request. The recommended User-Agent header format is <NF Type>-<NF Instance ID> <NF FQDN>.
producer-id	Indicates the 5G NF instance ID of the destination NF.
consumer-fqdn	Indicates FQDN of the originator NF, which is retrieved from the User-Agent header in the received request message. Conditions: • For SCP initiated request and response messages, for example, delegated Discovery and OAuth access token requests, consumer-fqdn metadata must be present, and the value should be the same as the feed-source->nf-fqdn metadata value. Note: As this is an SCP FQDN, there should not be any dependency on the User-Agent header to retrieve the FDQN information. • For consumer NF initiated messages, consumer-id metadata depends on the presence of the User-Agent header in the received service request. The recommended User-Agent header format is <NF Type>-<NF Instance ID> <NF FQDN>.
producer-fqdn	Indicates FQDN of the destination NF.

Table 3-31 (Cont.) Metadata Table

Metadata	Description
hop-by-hop-id	Uniquely identifies a request-response pairs within a transaction. The hop-by-hop-id uses a maximum substring length of 30 characters for the following fields as part of the identifier: • producerFqdn: Use the first 30 characters • consumerFqdn: Use the first 30 characters • podInstanceID: Use the last 30 characters The hop-by-hop-id uses the following format to identify request-response pairs: • Consumer NF to SCP or SCP to Consumer NF (RxRequest/TxResponse) – When the consumer NF is specified in the request: Format: <First 30 characters of consumerFqdn>_<Last 30 characters of worker-pod-instance-Id> Example: udm12.oracle.com_cp-scp-worker-56df9944b6-kxthp – When the consumer NF is not specified: Format: NA_<Last 30 characters of worker-pod-instanceId> Where, NA indicates the absence of a consumer NF FQDN in the request. Example: NA_cp-scp-worker-56df9944b6-kxthp • SCP to Producer NF or Producer NF to SCP (TxRequest/RxResponse): Format: <Last 30 characters of worker-pod-instance-Id>_<First 30 characters of producerFqdn>_Suffix Where, Suffix indicates an incrementing integer that increases with each hop. Example: cp-scp-worker-56df9944b6-kxthp_udm1svc.scpsvc.svc.cluster.loc_2

Table 3-31 (Cont.) Metadata Table

Metadata	Description
reroute-cause	Indicates the reroute cause. It contains one of the following parameters: • Circuit breaking: Indicates that a message is an alternate attempt due to circuit breaking functionality at SCP. • Outlier detection: Indicates that a message is an alternate attempt due to outlier detection functionality at SCP. • Egress-rate-limit: Indicates that a message is an alternate attempt due to egress rate limiting functionality at SCP • producer-nf-congestion: Indicates that a message is an alternate attempt due to producer NF congestion • Error received • Timeout • Not Applicable
timestamp	Indicates timestamp (in nanoseconds) when message is received or forwarded by SCP.
message-direction	Indicates whether a message is an ingress message sent to SCP or an egress message forwarded by SCP. It can be indicated by putting the following traffic feed trigger point name: • RxRequest • TxRequest • RxResponse • TxResponse
feed-source	Indicates the source of this traffic feed. This contains the key-value of different identity of the node sending the traffic feed. • Feed-source • nf-type = SCP • nf-fqdn = SCP's FQDN • nf-instance-id = SCP's NF instance id • pod-instance-id = SCP-worker's pod instance id

Sample traffic feed generated by SCP:

```
{
  "version": "1.0.0",
  "metadata-list": {
    "correlation-id": "19645510-4698-4303-84f9-41c5414822e0",
    "consumer-id": "26740918-e9cd-0205-aada-71a76214d33c",
    "consumer-fqdn": "udm121.oracle.com",
    "producer-fqdn": "udm1svc.default.svc.cluster.local",
    "hop-by-hop-id": "1",
    "reroute-cause": "ErrorReceived",
    "timestamp": "1659959278583551367",
    "message-direction": "RESPONSE_EGRESS",
    "feed-source": {
      "nf-type": "SCP",
      "nf-fqdn": "ocscp-scp-worker.scpsvc.svc.cluster.local",
      "nf-instance-id": "6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e",
      "pod-instance-id": "ocscp-scp-worker-65df786b4d-qv7hs"
    }
  }
},
```

```

"header-list": {
  "server": [
    "SCP-ocscp-scp-worker.scpsvc.svc.cluster.local"
  ],
  "content-type": [
    "application/problem+json"
  ]
},
"5g-sbi-message": "{\"title\":\"Service
Unavailable\", \"status\":\"503\", \"detail\":\"Service Unavailable:: Service
Unavailable\"}"
}

```

Managing Message Copy

Enable

By default, this feature is disabled. To enable this feature, you must complete the configuration as described in the following configuration section.

Configure

You can configure this feature using REST API and the CNC Console.

- **Configure using CNC Console:** For information about CNC Console configurations, see [Configuring SCP Features](#) and [Configuring Traffic Feed](#).
- **Configure using REST API:** For information about configuring REST APIs, see "Configuring SCP Features", "Configuring Traffic Feed Data Director", and "Configuring Traffic Feed Trigger Point Config" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

The following metrics are added in the [Metrics](#) section:

- ocscp_metric_trafficfeed_attempted_total
- ocscp_metric_trafficfeed_failed_total
- ocscp_metric_trafficfeed_not_attempted_total
- ocscp_metric_trafficfeed_partition_unhealthy
- ocscp_metric_trafficfeed_success_total
- ocscp_metric_trafficfeed_cluster_unhealthy

The following alerts are added in the [Alerts](#) section:

- [SCPTrafficFeedSendFailed](#)
- [SCPTrafficFeedPartitionUnhealthy](#)
- [SCPTrafficFeedKafkaClusterUnhealthy](#)

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.

2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.47 Observability Enhancements for SCP Inter-microservice Connectivity

In the previous releases, SCP microservices communicated with each other to perform various tasks, however, there was no mechanism to determine communication failures between any two SCP microservices.

This enhancement enables SCP to improve the SCP observability by introducing observability parameters to monitor inter-microservices communication failures.

SCP microservices communicate with each other through Hypertext Transfer Protocol (HTTP). To monitor any communication failure between SCP microservices, SCP observability services are enhanced with a new metric and an alert. This observability enhancement checks inter-microservice connectivity at regular intervals. If there is any communication failure, it raises an alert based on the threshold configurations. The alert is automatically cleared when the connection is restored.

Managing Observability Enhancements for SCP Inter-microservice Connectivity

Enable

This enhancement is automatically enabled in SCP at the time of deployment.

Configure

You can configure this feature using REST API and the CNC Console.

- **Configure using CNC Console:** For information about CNC Console configurations, see [Configuring SCP Services](#).
- **Configure using REST API:** For information about configuring using REST APIs, see "Configuring SCP Services" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Observe

For information about alerts and metrics, see [Alerts](#) and [Metrics](#).

Maintain

If you encounter alerts at system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.48 Load Control based on the Load Control Information Header

In the previous release, SCP was only relying on the producer NF's load information obtained from Network Repository Function (NRF) through NF status notifications, as SCP has subscribed for load status with NRF. This process might delay the availability of the instantaneous load information value at SCP.

With the introduction of Load Control based on the Load Control Information (LCI) Header feature, SCP has enhanced the existing load control functionality by enabling a direct exchange of the load information between 5G NFs and SCP using the "3gpp-Sbi-Lci" HTTP header.

When Service Communication Proxy (SCP) receives a "3gpp-Sbi-Lci" HTTP header, SCP calculates the effective available capacity based on the load reported by the peer NFs in LCI header or NF status notification from NRF. When SCP receives a 5G Service Based Interface(SBI) message request from a consumer NF, then SCP forwards the message request to a selected target NF based on the priority and effective available capacity. If the load of the producer NF is equal to or above the throttle onset threshold percent, SCP doesn't consider that producer NF for routing; instead, it goes for alternate routing options to other equivalent producer NFs.

SCP can also convey its own load control information to its peers so that the 5G NFs can take appropriate action based on its implementation. Thus, Load Control Information assists the receiver 5G NF in target NF reselection and load balancing.

Managing Load Control based on the Load Control Information Header

Enable

To enable the Load Control based on the Load Control Information Header feature, set the **Ici** option to true on the CNC Console. For more information, see [Configuring SCP Features](#).

Configure

You can configure this feature using REST API and the CNC Console.

- **Configure using REST API:** For information about the REST API configurations, see "Configuring Load Control Information" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
- **Configure using CNC Console:** For information about configuring NF instance IDs, see [Configuring SCP Features](#).

Observe

For information about metrics, see [Metrics](#).

Maintain

If you encounter alerts at the system level, see the [Alerts](#) section for resolution steps.

In case the alerts persist, perform the following tasks:

1. **Collect the logs:** For information about how to collect logs, see *Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.
2. **Raise a service request:** For information about how to raise a service request, see [My Oracle Support](#).

3.49 Support for Three-site Georedundancy

Geographical Redundant (Georedundant) deployments ensure high availability and redundancy of NFs. SCP supports a three-site georedundancy deployment of cnDBTier to ensure service availability when one of the SCP sites is down.

When the primary SCP is available, NFs send service requests to the primary SCP. When SCP at the primary site is unavailable, NFs redirect service requests to the secondary SCP or tertiary SCP until the primary SCP's active status is restored.

The SCP database consists of the following:

- **Configuration Data:** The configuration data is exclusive for a given SCP instance. Therefore, an exclusive logical database is created and used by an SCP instance to store its configuration data. You can configure SCP instance specific configurations using RESTful config API exposed by SCPC-Configuration microservice through the CNC Console.
- **Routing Data:** This is the routing rules data that SCP determines from NRF in 5G Core network topology.
- **Status Data:** This data provides the status of upgrade or rollback.

In a three-site deployment, each SCP instance uses its database. The data gets replicated to all other sites of the cnDBTier cluster so that the data is available on all the cnDBTier cluster sites in case of a cnDBTier instance failure. SCP data gets replicated between the georedundant sites by using cnDBTier replication services. cnDBTier can be supported in the following modes:

- **Two-site cnDBTier georeplication mode:** The database backup from one of the sites can be used for fault recovery of SCP.
- **Three-site cnDBTier georeplication mode:** The database backup from one of the sites can be used for fault recovery of SCP.
- **One-site cnDBTier deployment mode:** The georeplication is unavailable.

For more information about database replication in a georedundant environment, see *Oracle Communications Cloud Native Core cnDBTier User Guide*.

3.50 Automated Test Suite Support

SCP provides Automated Test Suite (ATS) for validating the functionalities. ATS allows you to execute SCP test cases using an automated testing tool, and then compares the actual results with the expected or predicted results. In this process, there is no intervention from the user. For more information about installing and configuring ATS, see *Oracle Communications Cloud Native Core, Automated Test Suite Guide*.

4

Configuring Service Communication Proxy using the CNC Console

This chapter provides information about how to configure and modify different services in Service Communication Proxy (SCP) using the Oracle Communications Cloud Native Configuration Console (CNC Console).

The REST API configurations can also be performed using the CNC Console.

4.1 Support for Multiclustert Deployment

The CNC Console supports both single and multiple cluster deployments.

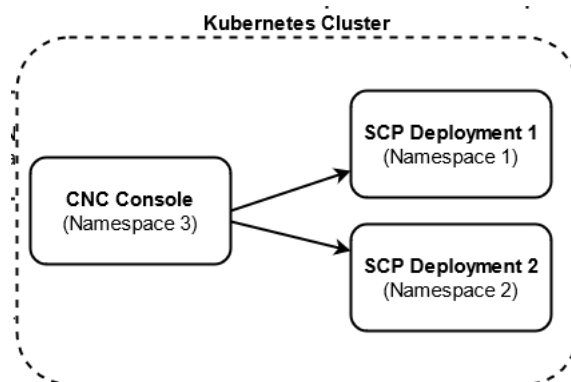
In a single cluster deployment, the CNC Console can manage NFs and Oracle Communications Cloud Native Environment (OCCNE) common services deployed in the local Kubernetes clusters.

In a multiclustert deployment, the CNC Console can manage NFs and OCCNE common services deployed in the remote Kubernetes clusters. For more information about single and multiple cluster deployments, see *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide*.

With the support of multiclustert deployment, a single instance of the CNC Console can configure two or multiple instances of SCP deployments if both CNC Console and SCP instances are deployed in the same Kubernetes cluster with different namespaces.

The following image represents a Kubernetes cluster with one instance of CNC Console and two instances of SCP. The single instance of the CNC Console is configuring two instances of SCP with different namespaces.

Figure 4-1 Support for Multiclustert Deployment



4.2 CNC Console Interface

This section provides an overview of the CNC Console to configure SCP features.

You can use SCP integrated with the CNC Console after logging in to CNC Console. To log in to the CNC Console, you must make the following updates to the hosts file at the C:\Windows\System32\drivers\etc location.

1. In the Windows system, open the hosts file in the notepad as an Administrator and append the following set of lines at the end:

```
<CNCC Node IP> cncc-iam-ingress-gateway.cncc.svc.cluster.local
<CNCC Node IP> cncc-core-ingress-gateway.cncc.svc.cluster.local
```

Example:

```
10.75.212.88 cncc-iam-ingress-gateway.cncc.svc.cluster.local
10.75.212.88 cncc-core-ingress-gateway.cncc.svc.cluster.local
```

Note

The IP Address in the above lines may change when deployment cluster changes.

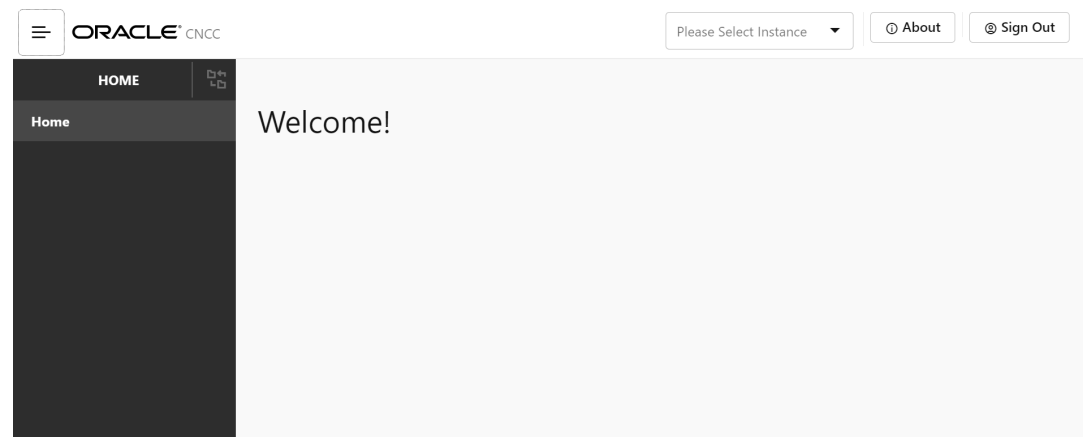
2. Save and close the hosts file.
Before logging in to the CNC Console, create a CNC user and password. Log in to the CNC Console using the same credentials. For information about creating a CNC Console user and password, see *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide*.

Logging in to the CNC Console and Selecting an SCP Instance

Perform the following procedure to log in to the CNC Console and select the required SCP instance to configure SCP features.

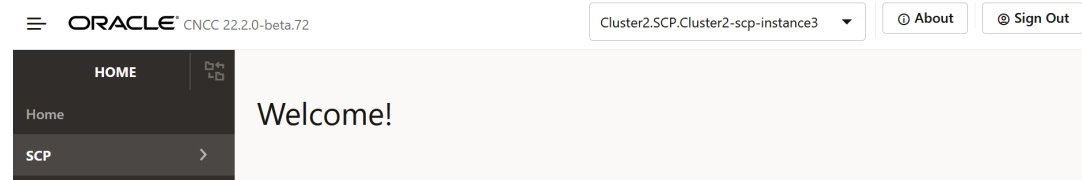
1. Open any web browser.
2. Enter the URL: http://<host name>:<port number>.
Where, <host name> is cncc-iam-ingress-ip and <port number> is cncc-iam-ingressport.
3. Enter the login credentials.
4. Click **Log in**.
The CNC Console Home page appears.

Figure 4-2 CNC Console Welcome Screen



5. In the upper pane, from the **Please Select Instance** drop-down list, select the required SCP instance.
The SCP tab appears in the left navigation pane.

Figure 4-3 Select NF Instance



The **Please Select Instance** drop-down list provides NF instances to configure corresponding NF features. You must select an appropriate SCP instance to configure SCP features. Alternatively, you can click one of the following interface elements on the Welcome screen:

- **About:** This element provides the CNC Console product name and version.
- **Sign Out:** This element exits the CNC Console.

4.2.1 Configuring SCP Features

This section provides information about enabling the following features of SCP:

Note

You must log in to the CNC Console while performing the procedures described in the subsequent subsections.

- Configuring Enhanced NF Status Processing
- Configuring Global Egress Rate Limiting
- Configuring Support for 5G SBI Roaming
- Configuring Mediation
- Configuring Model D Indirect 5G SBI Communication
- Configuring SCP User Agent Info
- Configuring Message Feed
- Configuring Load Control Information (LCI)
- Configuring Host Preference for Egress Message Requests
- Configuring CCA Header Validation
- Configuring Location Header Update for Host Mismatch
- Configuring Support for OAuth2.0
- Configuring SCP Health Check API
- NRF Configuration using DNS SRV Resolution
- Configuring Overload Control Based on the Overload Control Information Header

- Configuring Enhanced NFProfile Processing
- Configuring Error Response Enhancement

4.2.1.1 Configuring Error Response Enhancement

Perform the following procedure to enable or disable the Error Response Enhancement feature.

1. From the left navigation pane, click the SCP tab, and then click the SCP Features tab.
2. From the SCP Features list, click the Enhanced Error Response tab.
3. On the Enhanced Error Response page, click the **Edit** icon.

By default, the **Enabled** option is set to false for enhanced_error_rsp.

Figure 4-4 Error Response Enhancement

Enhanced Error Response	
Feature Name:	enhanced_error_rsp
Enabled:	true
Feature Specific Config	
Problem Details Enhancement	
Error Detail Max Size:	2100

4. In the **Error Detail Max Size**, set the limit of the error string.

This field manages the maximum length limit of the compiled error string in the detail parameter of problemDetails as described in "Configuring SCP Features" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. SCP truncates the error string from end if the compiled string is greater than the maximum limit. The default value is 2100.

5. Click **Save**.

4.2.1.2 Configuring Enhanced NF Status Processing

Perform the following procedure to enable the support for creating or retaining SCP routing rules for NF Profiles with NF Status as SUSPENDED.

① Note

The Enhanced NF Status Processing feature is applicable only when nnrf-nfm (NRF Management) service is used as an audit service for SCP.

1. In the left navigation pane, click the **SCP** tab, and then click the **SCP Features** tab.
2. From the SCP Features list, click **Enhanced Nf Status Processing**.
3. In the Enhanced NF Status section, click **Edit**.

Figure 4-5 Editing Enhanced Nf Status Processing

Enhanced NF Status

Feature Name:	Feature Name enhanced_nf_status_processing
Enabled:	Enabled false

▼ Feature Specific Config

Enhanced Suspended State Routing:	Enhanced Suspended State Routing UDM
Suspended State Routing:	Suspended State Routing PCF

4. Set **Enabled** to true.
By default, this option is set to false.
5. In the **Feature Specific Config** section, configure the following fields as required:
 - **Enhanced Suspended State Routing:** Enter the list of valid NF types for Mode 3 routing.
 - **Suspended State Routing:** Enter the list of valid NF types for Mode 2 routing.
6. Click **Save**.

4.2.1.3 Configuring Global Egress Rate Limiting

Perform the following procedure to configure global egress rate limiting.

1. In the left navigation pane, click the **Global Egress Rate Limiting** tab.
2. In the Global Egress Rate Limit section, click **Edit**.
The Edit Global Egress Rate Limit screen appears.

Figure 4-6 Enabling the Global Egress Rate Limiting

Edit Global Egress Rate Limit

Feature Name:	Feature Name global_egress_ratelimit
Enabled:	☐

Feature Specific Config

Remote ScpOne Enabled:	Remote ScpOne Enabled false
Remote ScpTwo Enabled:	Remote ScpTwo Enabled false

- Set **Enabled** to true.
By default, this option is set to false.
- In the Feature Specific Config section, configure the following fields:
 - Remote ScpOne Enabled:** Set this field to true if you want to enable egress rate aggregation with the SCP instance as defined in the `coherence.federation.remoteScpOne` parameter of the `ocscp-custom-values.yaml` file. By default, this value is set to false.
 - Remote ScpTwo Enabled:** Set this field to true if you want to enable egress rate aggregation with the SCP instance as defined in the `coherence.federation.remoteScpTwo` parameter of the `ocscp-custom-values.yaml` file. By default, this value is set to false.
- Click **Save**.

4.2.1.4 Configuring Support for 5G SBI Roaming

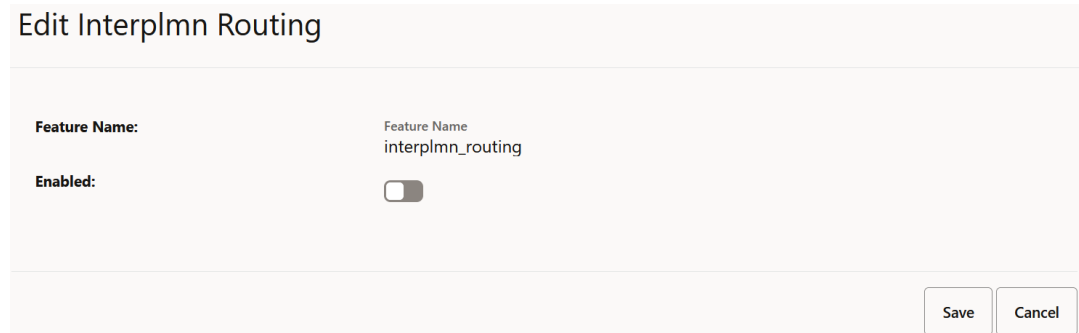
Perform the following procedure to configure support for 5G SBI roaming.

Note

Ensure that one local and one remote PLMNs are configured before enabling this feature. To configure a remote PLMN, see [Configuring SEPP Inter PLMN Routing](#). A local PLMN is configured at the time of SCP deployment or by editing the **CUSTOM_ORACLE_SCP** NF type as described in [Configuring NF Rule Profile](#).

1. In the left navigation pane, click the **Interplmn Routing** tab.
2. In the Interplmn Routing section, click **Edit**.
The Edit Interplmn Routing screen appears.

Figure 4-7 Inter PLMN Routing



Feature Name: interplmn_routing

Enabled: ☐

Save Cancel

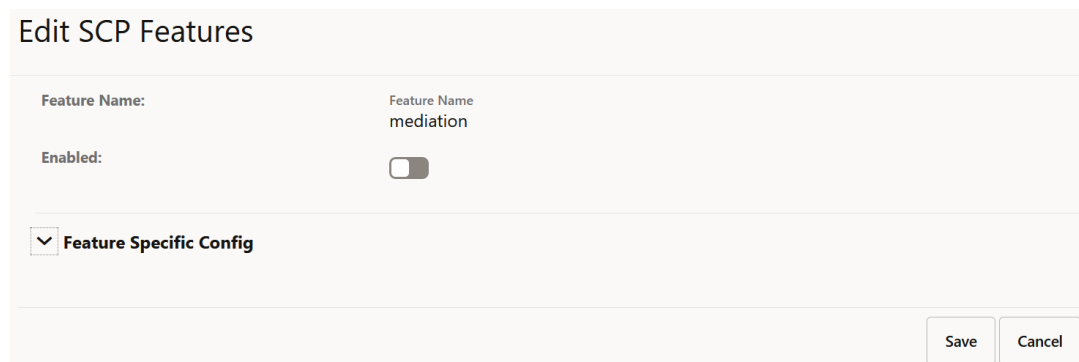
3. Set **Enabled** to true.
By default, this option is set to false.
4. Click **Save**.

4.2.1.5 Configuring Mediation

Perform the following to configure mediation.

1. In the left navigation pane, click the **Mediation** tab.
2. In the Mediation section, click **Edit**.
The Edit Mediation screen appears.

Figure 4-8 Enable Mediation



Feature Name: mediation

Enabled: ☐

Feature Specific Config

Save Cancel

3. Set **Enabled** to true.
By default, this option is set to false.
4. Click **Save**.

4.2.1.6 Configuring Model D Indirect 5G SBI Communication feature

Perform the following to configure Model D indirect 5G SBI communication feature.

1. In the left navigation pane, click the **Modeld Routing** tab.

2. In the Modeld Routing section, click **Edit**.

Figure 4-9 Enabling ModelID Routing

Modeld Routing	
Feature Name:	Feature Name modeld_routing
Enabled:	Enabled false
Feature Specific Config	
Caching:	Caching true
enforceReqSpecificSvcDiscovery:	enforceReqSpecificSvcDiscovery true

The Edit Modeld Routing screen appears.

3. Set **Enabled** to true.
By default, this option is set to false.
4. Set **Caching** to true or false to enable or disable caching to store discovery responses. By default, this option is set to true.
5. Set **enforceReqSpecificSvcDiscovery** to true or false to enable or disable NF Service specific Discovery Requests when possible. By default, this option is set to true.
6. Click **Save**.

4.2.1.7 Configuring SCP User Agent Info

Perform the following to configure SCP User Agent Info.

1. In the left navigation pane, click the **SCP User Agent Info** tab.
2. In the SCP User Agent Info section, click **Edit**.
The SCP User Agent Info screen appears.

Figure 4-10 SCP User Agent Info

SCP User Agent Info	
Feature Name:	Feature Name scp_user_agent_info
Enabled:	Enabled false
Feature Specific Config	
User Agent Header Format:	User Agent Header Format NFTYPE-NFINSTANCEID FQDN

3. Set **Enabled** to true.
By default, this option is set to false.

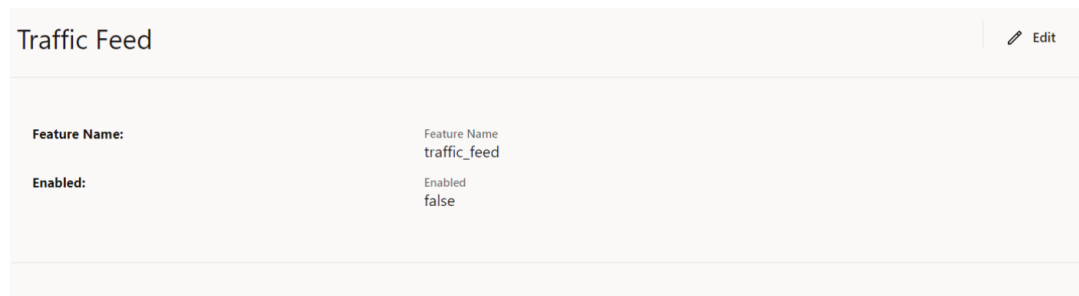
4. Expand the Feature Specific Config option, in the **User Agent Header Format** field, select the required format to set the "User-Agent" header format.
For more information about the "User-Agent" header formats, see "Configuring SCP Features" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
5. Click **Save**.

4.2.1.8 Configuring Message Feed

Perform the following to configure message feed.

1. In the left navigation pane, click the **Traffic Feed** tab.
2. In the Traffic Feed section, click **Edit**.
The Traffic Feed screen appears.

Figure 4-11 Message Feed



Traffic Feed		Edit
Feature Name:	Feature Name traffic_feed	
Enabled:	Enabled false	

3. Set **Enabled** to true.
By default, this option is set to false.
4. Click **Save**.

4.2.1.9 Configuring Load Control Information (LCI)

Perform the following procedure to enable configuring load control information (LCI).

1. In the left navigation pane, click the **LCI** tab.
2. In the LCI section, click **Edit**.

Figure 4-12 Editing LCI

The screenshot shows the 'Edit LCI' configuration page. At the top, there's a 'Feature Name' field containing 'lci'. Below it is an 'Enabled' toggle switch. The main section is titled 'Feature Specific Config' and contains several configuration items, each with a label and a value field or toggle:

- SCP LCI Conveyance Enable:** A toggle switch.
- Relay Peer Lci:** A toggle switch.
- SCP LCI Conveyance Interval:** A text input field containing '3600'.
- SCP LCI Minimum Load Change:** A text input field containing '25'.
- SCP LCI Conveyance Minimum Load Threshold:** A text input field containing '25'.
- SCP LCI Conveyance to Unknown peer:** A toggle switch.
- Peer LCI Processing Minimum Load Change:** A text input field containing '25'.
- Unknown peer LCI Expiry time:** A text input field containing '900'.

At the bottom right, there are 'Save' and 'Cancel' buttons.

3. Set Enabled to true.
By default, this option is set to false.
4. In the **Feature Specific Config** section, complete the following configurations:

Note

All the parameters are mandatory to configure the LCI feature.

- **SCP LCI Conveyance Enable:** Set to true to enable SCP to add its LCI in request and response.
- **Relay Peer Lci:** Set to true to allow SCP to forward the received LCI header from producer NF.
- **SCP LCI Conveyance Interval:** Enter a value from 100 milliseconds to 3600000 milliseconds to report the SCP LCI update interval.
- **SCP LCI Minimum Load Change:** Enter a value from 0 to 25 to report the minimum delta value when reporting LCI based on load change.
- **SCP LCI Conveyance Minimum Load Threshold:** Enter a value from 0 to 60 to set a minimum load change threshold to trigger the generation of an LCI header and its reporting to peers if allowed.

Note

If the SCP load value is less than the configured SCP LCI Conveyance Minimum Load Threshold value, SCP sends its load value as 0%.

- **SCP LCI Conveyance to Unknown peer:** Set to true to allow SCP to send its LCI to an unknown peer NF, a peer NF that is not registered with SCP.

Note

- SCP sends its LCI in all the messages that go to peer NFs that are not registered with SCP.
- If the SCP load value is less than the configured SCP LCI Conveyance Minimum Load Threshold value, SCP sends its load value as 0%.

- **Peer LCI Processing Minimum Load Change:** Enter a value from 0 to 25 to define a minimum load change threshold in peer NF's load as indicated in LCI, which can trigger a re-evaluation of routing rules.
- **Unknown peer LCI Expiry time:** Enter a value between 30 seconds to 900 second to set the minimum number of seconds required to remove an unknown peer's LCI from cache.

5. Click **Save**.

4.2.1.10 Configuring Host Preference for Egress Message Requests

Perform the following procedure to configure host preference for egress message requests.

1. In the left navigation pane, click the **Egress Host Preference** tab.
2. In the Egress Host Preference section, click **Edit**.

Figure 4-13 Editing Egress Host Preference

The screenshot displays the 'Editing Egress Host Preference' configuration page. It is divided into two main sections: 'Request' and 'Response'. The 'Request' section contains three dropdown menus: 'Api Root Header Present' (set to 'PASSTHROUGH'), 'Api Root Header Not Present' (set to 'IP'), and 'SCP Generated NRF Message' (set to 'IP'). The 'Response' section contains one dropdown menu: 'Headers' (set to 'FOLLOWREQUEST'). Below these sections is the 'Fqdn Resolution' section, which contains two dropdown menus: 'Fqdn' (set to 'NFPROFILE') and 'InterPlmn Fqdn' (set to 'DNS').

3. Set **Enabled** to true.
By default, this option is set to false.
4. In the **Feature Specific Config** section, to configure the host preference, set the following fields as described in the following table:

Table 4-1 Host Preference

Field Name	Description
Api Root Header Present	This parameter indicates the host preference (":authority" header) for egress message requests if the 3gpp-Sbi-Target-apiRoot header is present in the ingress message request. You can set this field to PASSTHROUGH to use the same type of host in egress message requests as received through ingress message requests. Default value: PASSTHROUGH Range: IP, FQDN, or PASSTHROUGH
Api Root Header Not Present	This parameter indicates the host preference (":authority" header) for egress message requests if the 3gpp-Sbi-Target-apiRoot header is absent in ingress message requests. Default value: IP Range: IP or FQDN
SCP Generated NRF Message	This parameter indicates the host preference (":authority" header) for SCP generated NRF messages. Default value: IP Range: IP or FQDN
Fqdn	This parameter indicates the resolution preference of egress message requests host FQDN if present. Default value: NFPROFILE Range: DNS or NFPROFILE
InterPlmn Fqdn	This parameter indicates the resolution preference of egress message requests inter-plmn FQDN if present. Default value: DNS Range: DNS or NFPROFILE

5. Click **Save**.

4.2.1.11 Configuring CCA Header Validation

Perform the following option to configure CCA Header validation.

1. In the left navigation pane, click the **CCA Header Validation** tab.
2. In the CCA Header Validation section, click **Edit**.

Figure 4-14 Edit CCA Header Validation

The screenshot shows the 'CCA Header Validation' configuration page in the Oracle CNC Console. The sidebar on the left lists various features under 'SCP Features'. The main content area has a header 'CCA Header Validation' with 'Edit' and 'Refresh' buttons. Below this, there are fields for 'Feature Name' (cca_header_validation) and 'Enabled' (false). A section titled 'Feature Specific Config' contains a table for 'Validations' with columns for Type, Error Profile Name, and Product NF Type. The table has one row with Type 'subject' and Error Profile Name 'ccaVerificationError'. Below the table, there is a section for 'tls.crt.san' with fields for 'preferred_validation_order' and 'max_entries_to_process'.

3. Set Enabled to true.
By default, this option is set to false.
4. In the **Feature Specific Config** section, to configure the validations, set the following fields as described in the following table:

Table 4-2 Validations

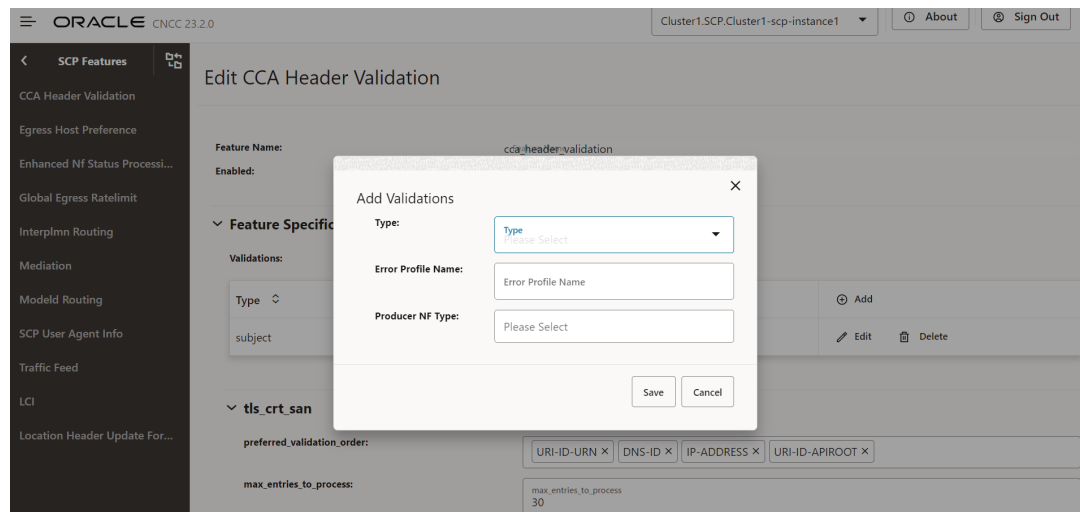
Field Name	Description
Type	Type of validation: - headerPresence: SCP checks the presence of the 3gpp-Sbi-Client-Credentials header in the ingress request. - Validation will pass if the header is present. - Validation will fail if the header is not present. - subject: SCP checks the NF instance ID from the "sub" parameter in the 3gpp-Sbi-Client-Credentials header with the NF instance ID from the list of SANs in the client's TLS certificate. SANs directly have the NF Instance Id or the client's other identity, like the FQDN or IP address, which will be used to get the NF Instance Id. If the 3gpp-Sbi-Client-Credentials header is not present, then this validation will not be performed. - Validation will pass if the NF instance ID from the "sub" parameter matches the NF instance ID from the SAN. - Validation will fail if the NF instance ID from the "sub" parameter doesn't match the NF instance ID from the SAN.
Error Profile Name	The error profile name is used to generate an error response if the corresponding validation fails. Error profiles can be configured using the REST API: /ocscp/scpc-configuration/{version}/errorProfileConfig.
Product NF Type	Default Value: Null
preferred_validation_order	An order format-wise from which the SCP picks SAN from the client's TLS certificate for verification of the "subject" type of validation. Note: Applicable only when validation type "subject" is selected.

Table 4-2 (Cont.) Validations

Field Name	Description
<code>max_entries_to_process</code>	Maximum number of SANs from the client's TLS certificate that SCP picks for validation. Note: Applicable only when validation type "subject" is selected.

- Click **Add** to add validations:

Figure 4-15 Add Validations

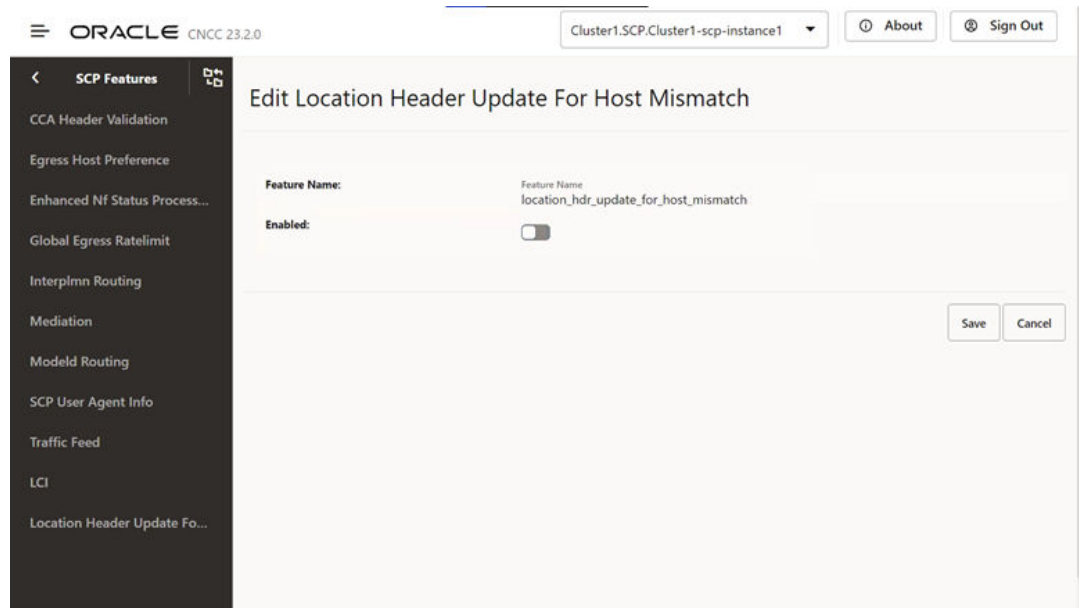


- Click **Save**.

4.2.1.12 Configuring Location Header Update for Host Mismatch

Perform the following the procedure to configure location header update for host mismatch.

- In the left navigation pane, click the **Location Header Update For Host Mismatch** tab.
- In the Location Header Update For Host Mismatch section, click **Edit**.

Figure 4-16 Edit Location Header Update For Host Mismatch

3. Set Enabled to true.
By default, this option is set to false.
4. Click **Save**.

4.2.1.13 Configuring Support for OAuth2.0

Perform the following the procedure to configure support for OAuth2.0.

1. In the left navigation pane, click the **OAuth2 Support** tab.
2. In the OAuth2 Support section, click **Edit**.

Figure 4-17 OAuth2 Support

Feature Name:	Feature Name
Enabled:	☐
Feature Specific Config	
SCP Access Token Capability:	SCP Access Token Capability
Access Token Conveyance:	☐
OAuth2 Access Token Validation:	☐
Access Token Validation Types:	Access Token Validation Types
Access Token Refresh Guard Time:	Access Token Refresh Guard Time
Access Token Validity Guard Time:	Access Token Validity Guard Time
Access Token Clean Up Post Expiry:	Access Token Clean Up Post Expiry
Access Token History Size:	Access Token History Size
Access Token Cache Size:	Access Token Cache Size
Requester Info:	Requester Info

- Set Enabled to true.
By default, this option is set to false.
- In the **Feature Specific Config** section, complete the following configurations:

Table 4-3 Configure Access Token

Field Name	Description
SCP Access Token Capability	This field indicates Access token support for the listed indirect communication modes at SCP. (ENUM). Possible values: [INDIRECT_COM_WITH_DELEG_DISC, INDIRECT_COM_WITHOUT_DELEG_DISC] - INDIRECT_COM_WITH_DELEG_DISC: SCP initiates access token request toward NRF in delegated discovery service request. - INDIRECT_COM_WITHOUT_DELEG_DISC: SCP is expected to forward the service request with or without access token as per configuration at SCP. SCP is not expected to initiate access token request toward NRF.

Table 4-3 (Cont.) Configure Access Token

Field Name	Description
Access Token Conveyance	This field conveys acquired access token in the "3gpp-Sbi-Access-Token" header in service response to consumer NFs. - Default value: true - Range: true or false
OAuth2 Access Token Validation	This field enables or disables validation of OAuth2 access token from consumer NFs. - Default value: Disabled - Range: Disabled or Enabled
Access Token Validation Types	This field configures the list of required validation types in the network. - Default value: empty as default - Range: Disabled or Enabled
Access Token Refresh Guard Time	This field initiates proactive refresh of cached access token when the configured time expires. The proactive refresh occurs if the relevant SBI messages are in exchange. - Default value: 60000ms - Range: 100ms - 300000ms
Access Token Validity Guard Time	This field indicates the time before the access token expiry when SCP considers not to use the existing access token and obtains new access token in the service request forwarded to producer NFs. - Default value: 30000ms - Range: 100ms - 300000ms
Access Token Clean Up Post Expiry	This field indicates the duration to purge the token from cache. - Default value: 900000ms - Range: 0ms - 3600000ms
Access Token History Size	This field indicates the number of access tokens signature history to identify whether the access token initiated by SCP or not. - Default value: 10 records - Range: 0 - 20 records
Access Token Cache Size	This field indicates the number of access tokens that can be cached in SCP. - Default value: 50000 records - Range: 5000 - 100000 records
Requester Info	This field indicates access token requester (consumer NF) Info to generate the access token request. - Prioritized list of default values: - DISCOVERY-HEADERS - CCA-HEADER - USER-AGENT-HEADER
Cache Enabled	This option enables or disables caching of access tokens. - Default value: true - Range: true or false

5. Click **Save**.

4.2.1.14 Configuring SCP Health Check API

Perform the following the procedure to configure SCP health check API.

1. In the left navigation pane, click the **Health Check** tab.
2. In the Health Check section, click **Edit**.

Figure 4-18 Health Check

Feature Name:	health_check
Enabled:	☒
Feature Specific Config	
SCP Health API	
Success Response Type:	Success Response Type 200_WITH_EMPTY_PAYLOAD
Overload	
Average SCP Load Threshold Value:	Average SCP Load Threshold Value 75
Overload Response Profile:	Overload Response Profile healthCheckErrorProfile
Next Hop SCP	
SCP Health Check Service Enabled:	☐
Request Timeout:	Request Timeout 1000
Polling Interval:	Polling Interval 1000

3. Ensure that Enabled is set to true.
This is the default configuration.
4. In the **Feature Specific Config** section, you can modify the following configurations:

Table 4-4 Configure Health Check API

Field Name	Description
Success Response Type	This field indicates successful responses for a healthy SCP. • Default value: 200StatusCodeAndEmptyPayload • Range: 200WithPayload, 200WithEmptyPayload, and 204WithNoContent
Average SCP Load Threshold Value	This field provides the overall average SCP load threshold value. • Default value: 75% • Range: 75% - 90%

Table 4-4 (Cont.) Configure Health Check API

Field Name	Description
Overload Response Profile	This field indicates the error profile configuration for the health query response in the exception conditions. Default value: healthCheckErrorProfile. For more information about this option, see "Configuring Error Profiles" in <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>.
SCP Health Check Service Enabled	This option enables or disables the SCP health check API feature in inter-scp scenarios. - Default value: false - Range: true or false
Request Timeout	This field indicates the timer to monitor the waiting time for health check response in inter-scp scenarios. - Default value: 1000ms - Range: 300ms -60000ms
Polling Interval	This field indicates the duration to control the periodicity of health check requests in inter-scp scenarios. - Default value: 1000ms - Range: 300ms -60000ms
Consecutive Error Response	This field indicates the total number of consecutive failure responses that leads to failover in inter-scp scenarios. - Default value: 3 - Range: 1 - 20
Consecutive Success Response	This field indicates the total number of consecutive successful responses that leads to fallback in inter-scp scenarios. - Default value: 3 - Range: 1 - 20

- Click **Save**.

4.2.1.15 NRF Configuration using DNS SRV Resolution

Perform the following procedure to NRF configuration using DNS SRV resolution.

- In the left navigation pane, click the **NRF Bootstrap Info** tab.
- In the NRF Bootstrap Info section, click **Edit**.

Figure 4-19 Edit NRF Bootstrap Info

The screenshot shows the 'Edit NRF Bootstrap Info' configuration page. It includes the following fields and controls:

- Feature Name:** nrf_bootstrap_info
- Enabled:** A toggle switch that is currently turned off.
- Feature Specific Config:** A section with a dropdown menu for 'Source' set to 'DNS_SRV' and a toggle switch for 'DeRegister Scp During Migration' which is turned on.
- Buttons:** 'Save' and 'Cancel' buttons at the bottom right.

- Set Enabled to true.
By default, this option is set to false.

4. In the **Feature Specific Config** section, enter the following input for the following configurations:

Table 4-5 Configure NRF Source Configuration

Field Name	Description
Source	This field is used to select whether the NRF Configuration Using DNS SRV Resolution feature should be enabled or disabled. SCP will enable the feature if the source is DNS_SRV. The default value is DNS_SRV.
deRegisterScpDuringMigration	In the migration from static to DNS SRV task, if static and DNS SRV NRF configurations are the same, then this parameter will be used to deregister SCP with the old or static NRFset. • Default value: false • Range: true or false

5. Click **Save**.

4.2.1.16 Configuring Overload Control Based on the Overload Control Information Header

Perform the following procedure to configure overload control based on the overload control information header.

1. From the SCP Features list, click the OCI tab.
2. On the OCI page, click the **Edit** icon.

Figure 4-20 Edit OCI

Edit OCI

Feature Name:	Feature Name oci
Enabled:	☐
▼ Feature Specific Config	
olcHSupportInd:	☒
SCP OCI Conveyance Interval:	SCP OCI Conveyance Interval 2,000
SCP OCI Recovery Validity Period:	SCP OCI Recovery Validity Period 3,600
Next Hop SCP OCI Enabled:	☒

3. In the Edit OCI section, set **Enabled** to true.
By default, this option is set to false.

Figure 4-21 Enabling OCI

Next Hop SCP OCI Rule Name:

Next Hop SEPP OCI Enabled: ☒

Next Hop SEPP OCI Rule Name:

▼ **scpOciConveyance**

enable: ☒

▼ **unknownPeer**

request: ☒

response: ☒

Save Cancel

4. To configure OCI, in the Feature Specific Config section, set the following fields:

Table 4-6 Configure OCI

Field Name	Description
olcHSupportInd	This option indicates that SCP supports the Overload Control Information (OCI) feature based on the 3gpp-Sbi-Oci header. By default, this option is enabled.
SCP OCI Conveyance Interval	This field indicates the interval for reporting SCP OCI to peer NFs. For every interval until the validity period, the last sent OCI is reported to peer NF irrespective of any change to SCP OCI. Sending SCP OCI periodically ensures that the peer NF has not missed an earlier reported SCP OCI. The same OCI header as sent on the last OCI threshold change or validity period expiry is sent. - Default value: 2000 milliseconds - Range: 2000 to 3600000 milliseconds Note: OCI is sent to peer NFs only if there is a message for that NF.
SCP OCI Recovery Validity Period	This field indicates the validity period to send OCI headers to peers when SCP recovers from an overloaded state. SCP sends OCI to peers with reduction metric as 0. - Default value: 3600 seconds - Range: 5 to 3600 seconds
Next Hop SCP OCI Enabled	This option enables or disables the OCI feature between SCP instances and SCP scope OCI. By default, this option is enabled.
Next Hop SCP OCI Rule Name	If the Next Hop SCP OCI Enabled option is enabled, the OCI feature enforcement is done based on the ociConfigRule REST API configuration with the name provided as the value for this option. By default, this value is NextHopScpOciRuleName.
Next Hop SEPP OCI Enabled	This option enables or disables OCI enforcement between SCP and SEPP interfaces and SEPP scope OCI. By default, this option is enabled.

Table 4-6 (Cont.) Configure OCI

Field Name	Description
Next Hop SEPP OCI Rule Name	If the Next Hop SEPP OCI Rule Name option is enabled, the OCI feature enforcement is done based on the ociConfigRule REST API configuration with the name provided as the value for this option. By default, this value is NextHopSeppOciRuleName.
scpOciConveyance	If this option is set to true, SCP starts conveying the 3gpp-Sbi-Oci header based on self-overload information. By default, this option is enabled.
unknownPeer • request	If this field is set to true, SCP starts conveying the 3gpp-Sbi-Oci header based on self-overload information to requests to unknown peers. For unknown peers, identification of peer NFs is done based on the message request's FQDN. By default, this option is enabled.
unknownPeer • response	If this option is set to true, SCP starts conveying the 3gpp-Sbi-Oci header based on self-overload information to responses to unknown peers. If this parameter is set to true, SCP adds self-OCI header in every response message. By default, this option is disabled.

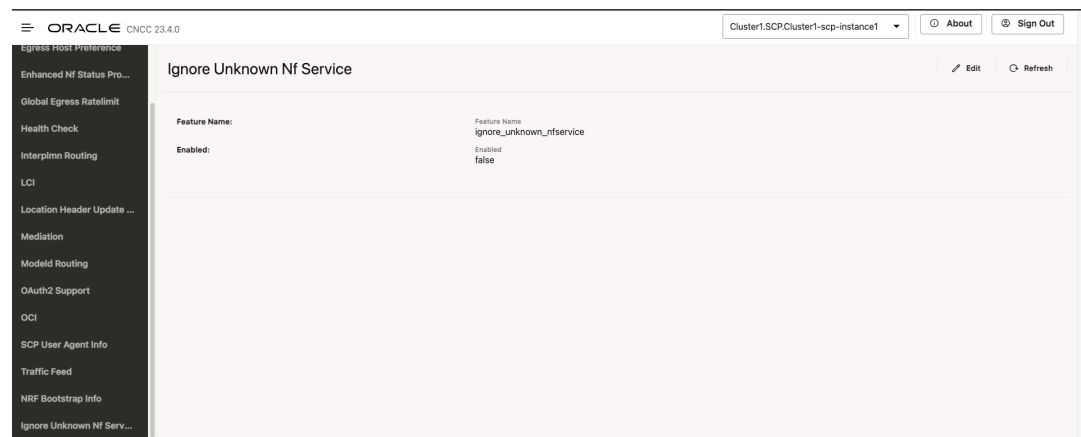
- Click **Save**.

4.2.1.17 Configuring Enhanced NFProfile Processing

Perform the following procedure to enable the NFProfile Processing enhancements supported by SCP using Cloud Native Configuration Console (CNC Console).

- In the left navigation pane, click the **Ignore Unknown Nf Service** tab.
The Ignore Unknown Nf Service page appears.

Figure 4-22 Ignore Unknown Nf Service



- On the Ignore Unknown Nf Service page, click the **Edit** icon.
- In the **Edit Ignore Unknown Nf Service** section, set **Enabled** to true.
By default, this option is set to false.

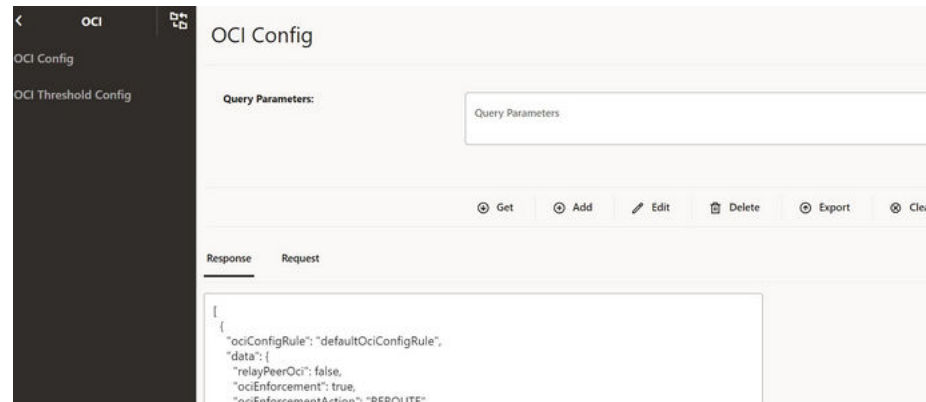
4.2.2 Configuring OCI

Perform the following procedure to retrieve, modify, and remove OCI configuration data.

- From the left navigation pane, click the **SCP** tab, and then click the **OCI** tab.

2. From the left navigation pane, click the **OCI Config** tab.
3. In the OCI Config section, click **Get** to retrieve OCI configuration data based on the query parameters.

Figure 4-23 OCI Config



4. To add a new request, click **Add** and enter the request body information in the **Request** field.
5. Click **Submit**.

The request body message appears in the **Response** field.

To edit, export, delete, or clear the data, click the required buttons as described in the following table:

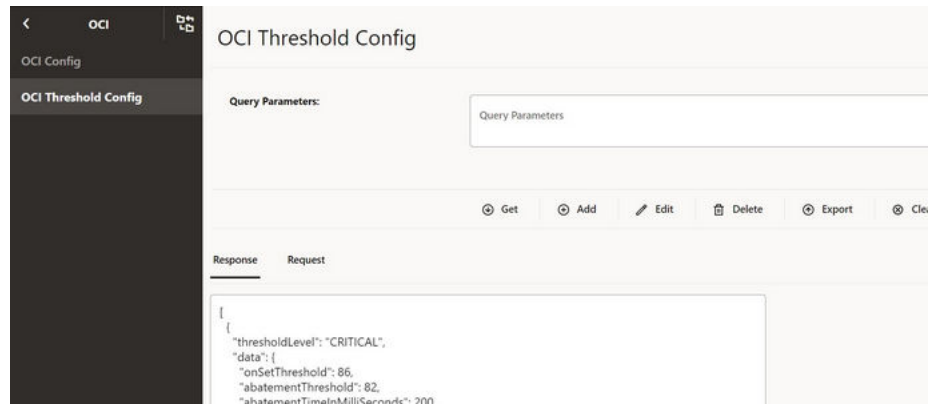
Table 4-7 Button and Field Descriptions of OCI

Button and Field Names	Description
Query Parameters	Allows to enter parameters to retrieve, add, and update OCI configuration data based on the query parameters.
Get	Retrieves OCI configuration data. You can modify this data using the Edit button.
Add	Adds new message requests in the Request field.
Edit	Modifies any existing data by updating mandatory query parameters in the Query Parameters field.
Delete	Removes any message request.
Export	Exports existing records in the JSON format.
Clear	Refreshes the existing screen by removing the data.

4.2.3 Configuring OCI Threshold Levels

Perform the following procedure to retrieve, modify, and remove OCI threshold configuration data.

1. From the left navigation pane, click the **SCP** tab, and then click the **OCI** tab.
2. From the left navigation pane, click the **OCI Threshold Config** tab.
3. In the OCI Threshold Config section, click **Get** to retrieve OCI threshold configuration data based on the query parameters.

Figure 4-24 OCI Threshold Config

4. To add a new request, click **Add** and enter the request body information in the **Request** field.
5. Click **Submit**.

The request body message appears in the **Response** field.

To edit, export, delete, or clear the data, click the required buttons as described in the following table:

Table 4-8 Button and Field Descriptions of OCI Threshold Configuration

Button and Field Names	Description
Query Parameters	Allows to enter parameters to retrieve, add, and update OCI threshold configuration data based on the query parameters.
Get	Retrieves OCI threshold configuration data. You can modify this data using the Edit button.
Add	Adds new message requests in the Request field.
Edit	Modifies any existing data by updating mandatory query parameters in the Query Parameters field.
Delete	Removes any message request.
Export	Exports existing records in the JSON format.
Clear	Refreshes the existing screen by removing the data.

4.2.4 Viewing cnDBTier APIs in CNC Console

Perform the following procedure to view the cnDBTier version, the status of cnDBTier clusters, georeplication status, and georeplication recovery status on the CNC Console.

Note

The following cnDBTier APIs are read only.

4.2.4.1 Backup List

Perform the following procedure to view the list of completed backups:

- From the left navigation pane, click the **SCP** tab, and then click the **cnDBTier** tab.

- Click the **Backup List** to view the list of completed backups along with Backup ID, Backup size, and Creation Timestamp.
The **Backup List** screen is displayed.

Table 4-9 Backup List

Fields	Description
Backup Details	This field displays information such as backup Id, backup size, and backup creation timestamp.
Site Name	This field displays the name of the current site to which SCP is connected.
Backup Id	This field displays the ID of the stored backup.
Backup Size (bytes)	This field displays the size of the stored backup.
Creation TimeStamp	This field displays the time recorded when the backup was stored.

4.2.4.2 cnDBTier Version

Perform the following procedure to view the version:

- From the left navigation pane, click the **SCP** tab, and then click the **cnDBTier** tab.
- Click the **cnDBTier Version** to view the version.

Table 4-10 cnDBTier Version Attributes

Fields	Description
cnDBTier Version	This field displays the cnDBTier version.
NDB Version	This field displays the network database (NDB) version.

4.2.4.3 Database Statistics Report

Perform the following procedure to view the available database.

- From the left navigation pane, click the **SCP** tab, and then click the **cnDBTier** tab.
- Click the **Database Statistics Report** to view the available database.

Table 4-11 Database Statistics Report

Fields	Description
Database Count	This field displays the number of available database.
Database Tables Count	This field displays the available database names and their table count.
Database Name	This field displays the database name.
Table Count	This field displays the table count for each database.
Database Table Rows Count	This field displays the table rows present in each table.

Click on **View** icon available next to the database name to view the **View Database Table Rows Count** screen.

Table 4-12 View Database Table Rows Count

Fields	Description
Database Name	This field displays the database name.
Tables	This field displays the table names and the corresponding rows in each table.
Table Name	This field displays the table name.
Row Count	This field displays the table rows present in each table.

4.2.4.4 Georeplication Recovery

Perform the following procedure to mark cnDBTier cluster as failed, execute georeplication recovery, and monitor their status:

1. From the left navigation menu, navigate to **SCP** and then click **cnDBTier** tab.
2. Click **Georeplication Recovery** to access the Georeplication Recovery Status of the cnDBTier cluster. This includes options such as Update Cluster As Failed, Start Georeplication Recovery, and Georeplication Recovery Status.
 - Click the **Update Cluster As Failed** to mark the cluster as FAILED. The **Update Cluster As Failed** page is displayed.

Table 4-13 Update Cluster As Failed

Fields	Description
Cluster Names	This field displays a list of cnDBTier clusters that can be marked as failed.
Failed Cluster Names	This field displays a cnDBTier cluster that is marked as failed.

Click **Update Cluster**. The selected cluster name is updated in the **Failed Cluster Names** field.

- Click the **Start Georeplication Recovery** to start the georeplication recovery process for a failed site. The **Start Georeplication Recovery** page is displayed.

Table 4-14 Start Georeplication Recovery

Attribute	Description
Failed Cluster Name	This field displays a list of all the clusters that have been marked as failed.
Backup Cluster Name (Optional)	This field displays a list of all the healthy clusters. If no cluster is selected, the system uses the first available healthy cluster for the backup.

Click **Start Georeplication Recovery** to initiate georeplication recovery.

- Click the **Georeplication Recovery Status** to view the status of georeplication recovery for cnDBTier clusters. The **Georeplication Recovery Status** page is displayed.

Table 4-15 Georeplication Recovery Status

Attribute	Description
Local Cluster Name	This field displays the name of the local cluster.
Georeplication Recover Status Details	This field displays the details of the georeplication recovery status of cnDBTier clusters.
Cluster Name	This field displays the clusters by name.
Georeplication Recovery Status	This field displays the current georeplication recovery status of the corresponding cluster.

The following are the statuses of Georeplication Recovery:

Table 4-16 Georeplication Recovery Status

Georeplication Recovery Status	Description
ACTIVE	The cluster is in a healthy state, and replication is up and running with its respective mate cluster.
REINSTALLED	The cluster enters this state during fatal error recovery when the end user reinstalls the cluster.
STARTDRRESTORE	When Georeplication recovery is started, the cluster will transition into this state.
INITIATEBACKUP	When Georeplication recovery is started, the cluster will identify a healthy cluster for backup initiation and transition into this state.
CHECKBACKUP	When the backup is initiated, the georeplication recovery cluster will monitor the progress of the backup until its completion. If the backup fails, the cluster will restart the backup.
COPY_BACKUP	Upon completion of the backup, the georeplication recovery cluster will request the transfer of the backup from the healthy cluster to the georeplication recovery cluster.
CHECK_BACKUP_COPY	When backup copy is started georeplication recovery cluster will monitor for the backup transfer progress till it's completion and if it's fails the cluster will re-initiates the backup transfer.
BACKUPCOPIED	When the backup copy is started, the georeplication recovery cluster will monitor the progress of the backup transfer until its completion. If the transfer fails, the cluster will restart the backup transfer.
BACKUPEXTRACTED	This state indicates that the backup has been successfully extracted at the georeplication recovery cluster, allowing the restoration of the backup to start.
FAILED	This state is used by end user to mark specific cluster as failed and hence georeplication recovery is essential to recover the cluster. This state can also indicates that georeplication recovery started and the database is restored using the healthy cluster backup.
UNKNOWN	This state is used by the end user to mark a specific cluster as failed, necessitating georeplication recovery for cluster recovery. Additionally, this state can indicate that georeplication recovery has started and the database has been restored using the backup from the healthy cluster.
RECONNECTSQLNODES	This state is used to instruct SQL nodes to be offline during backup restoration to prevent any records from entering the binlog of the georeplication recovery cluster.

Table 4-16 (Cont.) Georeplication Recovery Status

Georeplication Recovery Status	Description
BACKUPRESTORE	This state indicates that the backup, successfully copied from the healthy cluster, is currently being used to restore the georeplication recovery cluster.
RESTORED	When the backup is successfully restored in the georeplication recovery cluster, the cluster will enter this state to start the reestablishment of replication channels.
BINLOGINITIALIZED	This state indicates the start of binlogs for the restoration of replication channels, necessary to start the restore process
RECONFIGURE	When the binlog is restarted, the georeplication recovery cluster will reestablish the replication channels with respect to all its mate clusters.

4.2.4.5 Georeplication Status

Perform the following procedure to view the local site and remote site name to which SCP is connected.

1. From the left navigation pane, click the **SCP** tab, and then click the **cnDBTier** tab.
2. Click the **Georeplication Status** to view the local site and remote site name to which SCP is connected.

Table 4-17 Georeplication Status

Fields	Description
Local Site Name	This field displays the local site name to which SCP is connected.
Remote Site Name	This field displays the remote site name.
Replication Status	This field displays the replication status with corresponding sites.
Seconds Behind Remote Site	This field displays the number of seconds that the last record read by the local site is behind the latest record written by the remote site for all the replication groups.

- a. Click the **View** icon in the **Actions** menu to view the **View Georeplication Status** screen.

Table 4-18 Georeplication Status

Fields	Description
Replication Group Delay	This field displays the number of seconds that the last record read by the local site is behind the latest record written by the remote site for individual replication groups.
Replication Channel Group Id	This field displays the ID of the replication channel group.

- b. Click the **View** icon to view the **Replication Group Delay** attributes.

Table 4-19 View Replication Group Delay

Fields	Description
Channel Details	This field displays the channel details such as Remote Replication IP and Role.
Remote Replication IP	This field displays the IP of the remote replication channel.
Role	This field displays the role of the replication channel IP.

4.2.4.6 HeartBeat Status

Perform the following procedure to view the connectivity between local site and remote site name to which SCP is connected.

1. From the left navigation pane, click the **SCP** tab, and then click the **cnDBTier** tab.
2. Click the **HeartBeat Status** to view the connectivity between local site and remote site name to which SCP is connected.

Table 4-20 HeartBeat Status Details

Fields	Description
Site Name	This field displays the name of the current site to which SCP is connected.
HeartBeat Details	This field displays information such as the remote site name, heartbeat status, heartbeat lag, and replication channel group id.
Remote Site Name	This field displays the remote site name.
Heartbeat Status	This field displays the connectivity status with corresponding sites.
Heartbeat Lag	This field displays the lag or latency in seconds it took to synchronize between sites.
Replication Channel Group Id	This field displays the ID of the replication channel group.

4.2.4.7 Local Cluster Status

Perform the following procedure to view the local cluster status for the current site.

1. From the left navigation pane, click the **SCP** tab, and then click the **cnDBTier** tab.
2. Click the **Local Cluster Status** to view the local cluster status for the current site:

Table 4-21 Local Cluster Status

Fields	Description
Site Name	This field displays the name of the current site to which SCP is connected.
Cluster Status	This field displays the local cluster status for the current site.

4.2.4.8 On Demand Backup

Perform the following procedure to view the available database.

1. From the left navigation pane, click the **SCP** tab, and then click the **cnDBTier** tab.

2. Click the **On Demand Backup** to create a new backup and view the status of initiated on-demand backups.

Table 4-22 On Demand Backup Details

Fields	Description
Site Name	This field displays the name of the current site to which SCP is connected.
DR Status	This field displays the status of DR.
Backup Id	This field displays the ID of the stored backup.
Backup Status	This field displays the status of backup.
Remote Transfer Status	The field displays the status of remote transfer.
Initiate Backup	The field displays whether the backup is initiated or not.

- a. Click the **Edit** icon.
The **Edit** On Demand Backup screen appears.

Note

The **Edit** mode is available only for Initiate Backup.

- b. Enable the Initiate Backup option click **Save**.
A confirmation message "Save successfully" appears.
- c. Click the **Cancel** to navigate back to the On Demand Backup screen.
- d. Click the **Refresh** to reload the On Demand Backup screen.

4.2.4.9 cnDBTier Health

Perform the following procedure to view the health status of the microservices:

1. From the left navigation menu, navigate to **SCP** and then click **cnDBTier** tab.
The **cnDBTier** page is displayed.
2. Click **cnDBTier Health** to view the health status of the microservices like replication, backup manager, monitor services, and NDB services.
The **cnDBTier Health** page is displayed.
 - Click the **Backup Manager Health Status** to view the health status of the backup manager.
The **Backup Manager Health Status** page is displayed.

Note

The following APIs are read-only.

Table 4-23 Backup Manager Health Status

Fields	Description
Service Name	This attribute displays the service name of the backup manager microservice.

Table 4-23 (Cont.) Backup Manager Health Status

Fields	Description
Service Status	This attribute displays the service status of the backup manager microservice. Possible values are UP and DOWN.
DB Connection Status	This attribute displays the database connection status of the backup manager microservice. Possible values are UP and DOWN.
Overall Backup Manager Service Health	This attribute displays the overall health status of the backup manager microservice. Possible values are UP and DOWN.
Backup Executor Health Status	This attribute displays the following information like node id and DB connection status of the backup executor.
Node Id	This attribute displays the id of the node.
DB Connection Status	This attribute displays the backup executor database connection status with the nodes. Possible values are UP and DOWN.

- Click the **Monitor Health Status** to view the health status of the services. The **Monitor Health Status** page is displayed.

Note

The following APIs are read-only.

Table 4-24 Monitor Health Status details

Attribute	Description
Service Name	This attribute displays the service name of the monitor microservice.
DB Connection Status	This attribute displays the database connection status of the monitor microservice. Possible values are UP and DOWN.
Metric Scrape Status	This attribute displays the status of the metric scrape, that is if the metrics are fetched or not. If the metrics are fetched then the service is up and vice versa. Possible values are UP and DOWN.
Overall Monitor Service Health	This attribute displays the overall health status of the monitor microservice. Possible values are UP and DOWN.

- Click the **NDB Health Status** to view the health status of the network database. The **NDB Health Status** page is displayed.

Note

The following APIs are read-only.

Table 4-25 NDB Health Status details

Attribute	Description
Local Site Name	This attribute displays the name of the current site. For example, site 1, site 2.
NDB Health Status Details	This attribute displays the health status of the network database like name of the NDB service, status of the service, health status of PVC.
Service Name	This attribute displays the service name. For example, ndbmgmd-0, ndbmtid-0, ndbmyappsqld-1, ndbmysqld-2.
Service Status	This attribute displays the status of the service. Possible values are UP and DOWN.
PVC Health Status	This attribute displays the health status of the PVC. Possible values are UP, DOWN, and NA. Note: This attribute is set to NA when some of the database pods are not connected to the PVC.

- Click the **Replication Health Status** to view the health status of the replication sites. The **Replication Health Status** page is displayed.

Note

The following APIs are read-only.

Table 4-26 Replication Health Status details

Attribute	Description
Local Site Name	This attribute displays the name of the current site (site 1, site 2).
Health Status Details	This attribute displays the health status details of the local site like replication service name, replication service status, database connection status of the replication service, and the overall health status of the replication microservices. The number of rows in this table varies depending on the type of deployment (for example, two-site, three-site deployments).
Service Name	This attribute displays the name of the available replication service.
Service Status	This attribute displays the status of the available replication service. Possible values are UP and DOWN.
DB Connection Status	This attribute displays the database connection status of the replication microservice. Possible values are UP and DOWN.

Table 4-26 (Cont.) Replication Health Status details

Attribute	Description
Overall Replication Service Health	This attribute displays the overall health status of the replication microservice. Possible values are UP and DOWN.

4.2.5 Configuring NRF SRV

Perform the following procedure to configure NRF using DNS SRV Resolution.

1. From the left navigation pane, navigate to **SCP** and click the **NRF SRV Configuration** tab.
2. In the NRF SRV FQDN section, click the **Add** button to create the new NRF SRV configuration for the given NRF SRV FQDN.
You can configure the NRF configurations at run time, and this NRF SRV FQDN shall be mapped to target NRF FQDNs. SCP uses this NRF SRV FQDN to query the DNS SRV server to get the actual target NRF FQDNs. NRF SRV FQDN

Figure 4-25 Add NRF SRV Configuration

3. To configure the NRF SRV, enter the input for the following fields:

Table 4-27 NRF SRV Configuration

Field Name	Description
NF SetId List	This is the SetId list for this NRF SRV configuration. You can configure multiple nfSetIds, but the NRF profile and rule creation will only take into account the nfSetId from the 0th index. This setId must be unique for each NRF SRV configuration; that is, this setId must not be present in any other NRF SRV configuration.
Perform Subscription	This field allows to decide whether NRF from this NRF SRV should be used for a subscription or not. The possible values are true or false.

Table 4-27 (Cont.) NRF SRV Configuration

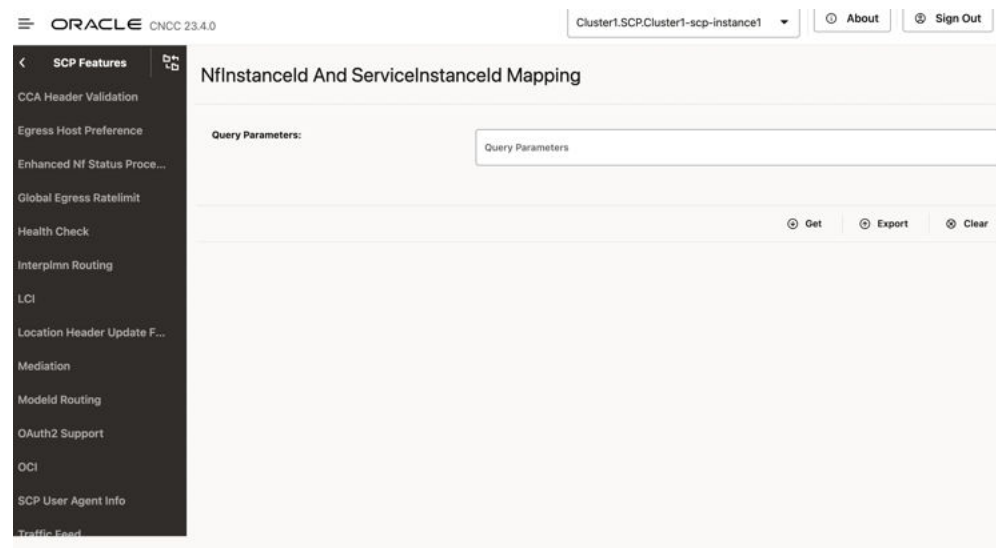
Field Name	Description
Perform Audit	This field allows to decide whether NRF from this NRF SRV should be used for a audit or not. The possible values are true or false.
Register SCP	This field allows to decide whether to register SCP with the NRF from the NRF Set. The possible values are true or false.
Scheme	This field is used for the URI Scheme. The supported value is http/https.
PLMN List	This field indicates the list of NRFs serving Public Land Mobile Network (PLMN). A PLMN has a unique PLMN code, which consists of a MCC (Mobile Country Code) and an MNC (Mobile Network Code). Click the Add button to add MCC and MNC, respectively.
Versions	This field lists the NFServiceVersion. Configuring multiple API versions is permissible, but at least one entry in the version list must have its apiVersionInUri set to "v1." This is because SCP currently utilizes "v1" for its self-generated requests towards NRF. - Click the Add button to add NFServiceVersion. - The supported value of apiVersionInUri is v1/v2. - The apiFullVersion should be in format x.y.z., for example, 1.0.0.
apiPrefix	This field is used in the URI while communicating with NRF.
serviceNames	Indicates the service name of the NRF SRV configuration. The nnrf-nfm and nnrf-disc are mandatory for NRF SRV configurations. The supported value is nnrf-nfm/nnrf-disc/nnrf-oauth2.
isInterPlmnFqdn	When isInterPLMNFqdn is true, it enables InterPLMN request routing. This means requests coming from foreign PLMNs to the SCP can be routed. SCP will perform intra-PLMN routing to local NRFs based on the interplmnFQDN.

- In the NRF SRV FQDN section, click the **Edit** icon to update the existing NRF SRV configuration.
- Click **Save**.

4.2.5.1 Mapping NFInstanceId and ServiceInstanceId

Perform the following procedure to map the NRF FQDN InstanceId.

- In the left navigation pane, click the **SCP** tab, and then click the **SCP Features** tab.
- From the SCP Features list, click **NFInstanceId And ServiceInstanceId Mapping**.
- In the NFInstanceId And ServiceInstanceId Mapping section, click **Get** to retrieve the NRF FQDN and corresponding instanceId mapping details.

Figure 4-26 NFInstanceId And ServiceInstanceId Mapping

4. To retrieve, export, or clear the data, click the required buttons as described in the following table:

Table 4-28 Buttons and Descriptions of Mapping NRF FQDN InstanceId

Button and Field Names	Description
Query Parameters	Allows to enter parameters to retrieve and update data required for mapping NRF FQDN InstanceId.
Get	Retrieves the NRF FQDN and corresponding instanceId mapping details. You can modify this data using the Edit button.
Export	Exports existing records in the JSON format.
Clear	Refreshes the existing screen by removing the data.

4.2.6 Configuring NRF

Perform the following procedure to configure NRF preferred by SCP for access token requests.

1. From the left navigation pane, click the SCP tab, and then click the NRF Configuration tab.
2. In the NRF CONFIGURATION section, click **Get** to retrieve NRF configuration information.

Figure 4-27 NRF Configuration

The screenshot shows the 'NRF CONFIGURATION' page. At the top, there's a header 'NRF CONFIGURATION'. Below it, on the left, is a label 'Query Parameters:' followed by a text input field labeled 'Query Parameters'. To the right of the input field is a toolbar with four buttons: 'Get' (with a magnifying glass icon), 'Edit' (with a pencil icon), 'Export' (with a download icon), and 'Clear' (with an 'X' icon). Below the toolbar, there are two tabs: 'Response' and 'Request', with 'Request' being the active tab. The main content area below the tabs is empty.

3. To edit, export, or clear the data, click the required buttons as described in the following table:

Table 4-29 Button and Field Descriptions of NRF Configuration

Button and Field Names	Description
Query Parameters	Allows to enter parameters to retrieve, add, and update NRF configuration data.
Get	Retrieves NRF configurations. You can modify this data using the Edit button.
Edit	Modifies any existing data by updating mandatory query parameters in the Query Parameters field.
Export	Exports existing records in the JSON format.
Clear	Refreshes the existing screen by removing the data.

Note

This configuration is only applicable to the OAuth service.

4.2.7 Configuring Upgrade and Rollback Event

Perform the following procedure to fetch upgrade and rollback event information.

1. From the left navigation pane, click the **SCP** tab, and then click the **Upgrade Status** tab.
2. On the Upgrade Status page, the following columns display upgrade and rollback events:

Figure 4-28 Upgrade and Rollback Events

Upgrade Status

✕
↻ Refresh

Service Name	Event	Source Release	Target Release	Creation Timestamp
No data to display.				

Table 4-30 Upgrade and Rollback Events

Column Name	Description
Service Name	Displays the names of the services, such as SCPC-Audit, SCPC-Configuration, SCPC-Notification, and so on.
Event	Displays the names of the upgrade and rollback events, such as Pre_Upgrade_Started, Pre_Upgrade_Completed, Pre_Upgrade_Failed, and so on.
Source Release	Displays the source release version for performing upgrade and rollback. The convention to identify a release is as follows: 1.12.0 is identified as 101200 1.14.0 is identified as 101400
Target Release	Displays the target release version for performing upgrade and rollback. The convention to identify a release is as follows: 1.15.0 is identified as 101500 22.1.0 is identified as 221000
Creation Timestamp	Displays the time of creation of the event, for example, creationTimestamp": "2021-05-26T01:17:15.000+00:00"

4.2.8 OAuth2.0 Configurations

This section provides information about configuring access token granularity and requests for NF types or NF service instances.

You must log in to the CNC Console while performing the procedures described in the subsequent subsections. Using these procedures, you can perform the followings tasks:

- Configure access token granularity
- Configure OAuth2.0 Local PLMN

4.2.8.1 Configuring Access Token Granularity

Perform the following procedure to configure access token granularity and access token requests for NFType, specific NF, or NF instance ID.

1. From the left navigation pane, click the SCP tab, and then click the OAuth Services tab.
2. From the left navigation pane, click the **Access Token Granularity Configuration** tab.
3. In the Access Token Granularity Configuration section, click **Get** to retrieve access token granularity configurations data based on the query parameters.

Figure 4-29 Access Token Granularity

Access Token Granularity Configuration

Query Parameters:

Get Add Edit Delete Export Clear

Response Request

```
[
  {
    "ruleName": "Config1",
    "data": {
      "requesterPlmnIds": [
        {
          "mcc": "325",
          "mnc": "13"
        },
        {
          "mcc": "326",
          "mnc": "14"
        }
      ]
    }
  }
]
```

- To add a new request, click **Add** and enter the request body information in the **Request** field.
- Click **Submit**.
The request body message appears in the **Response** field.
- To edit, export, delete, or clear the data, click the required buttons as described in the following table:

Table 4-31 Button and Field Descriptions of Access Token Granularity

Button and Field Names	Description
Query Parameters	Allows to enter parameters to retrieve, add, and update access token granularity configuration data based on the query parameters.
Get	Retrieves access token granularity configuration data. You can modify this data using the Edit button.
Add	Adds new message requests in the Request field.
Edit	Modifies any existing data by updating mandatory query parameters in the Query Parameters field.
Delete	Removes any message request.
Export	Exports existing records in the JSON format.
Clear	Refreshes the existing screen by removing the data.

4.2.8.2 Configuring OAuth2.0 Local PLMN Required

Perform the following procedure to configure local PLMN OAuth2.0.

- From the left navigation pane, click the SCP tab, and then click the OAuth Services tab.
- From the left navigation pane, click the OAuth2Required Configuration tab.

3. In the OAuth2Required Configuration section, click **Get** to retrieve OAuth2Required configurations data based on the query parameters.

Figure 4-30 OAuth2Required Configuration

OAuth2Required Configuration

Query Parameters:

Get Add Edit Delete Export Clear

Response Request

```
{
  "ruleName": "default",
  "data": {
    "nfType": "**",
    "oauth2Required": false,
    "defaultScope": {
      "scopeSource": "NF-TYPE"
    }
  }
}
```

4. To add a new request, click **Add** and enter the request body information in the **Request** field.
5. Click **Submit**.
The request body message appears in the **Response** field.
6. To edit, export, delete, or clear the data, click the required buttons as described in the following table:

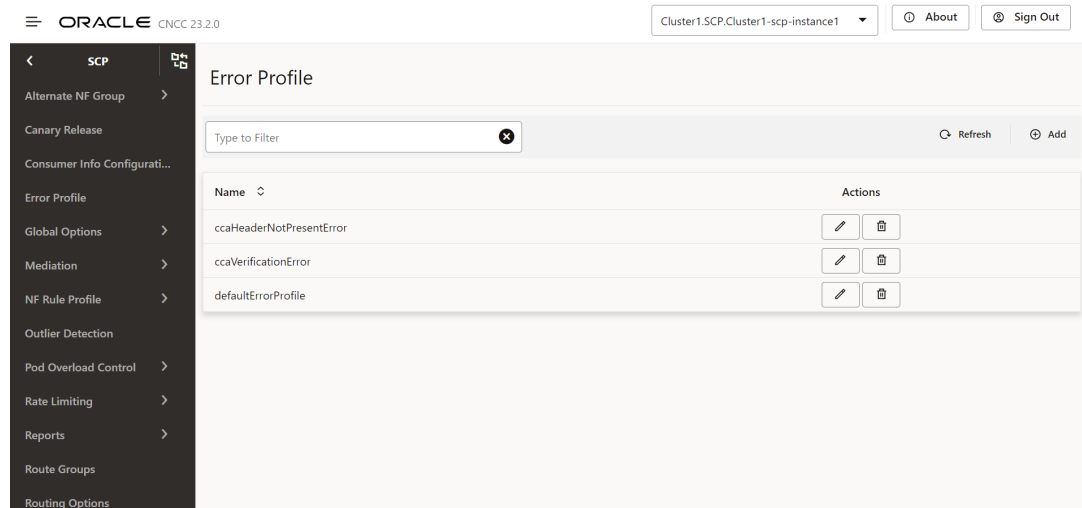
Table 4-32 Button and Field Descriptions of OAuth2Required Configuration

Button and Field Names	Description
Query Parameters	Allows to enter parameters to retrieve, add, and update OAuth2Required configuration data based on the query parameters.
Get	Retrieves OAuth2Required configuration data. You can modify this data using the Edit button.
Add	Adds new message requests in the Request field.
Edit	Modifies any existing data by updating mandatory query parameters in the Query Parameters field.
Delete	Removes any message request.
Export	Exports existing records in the JSON format.
Clear	Refreshes the existing screen by removing the data.

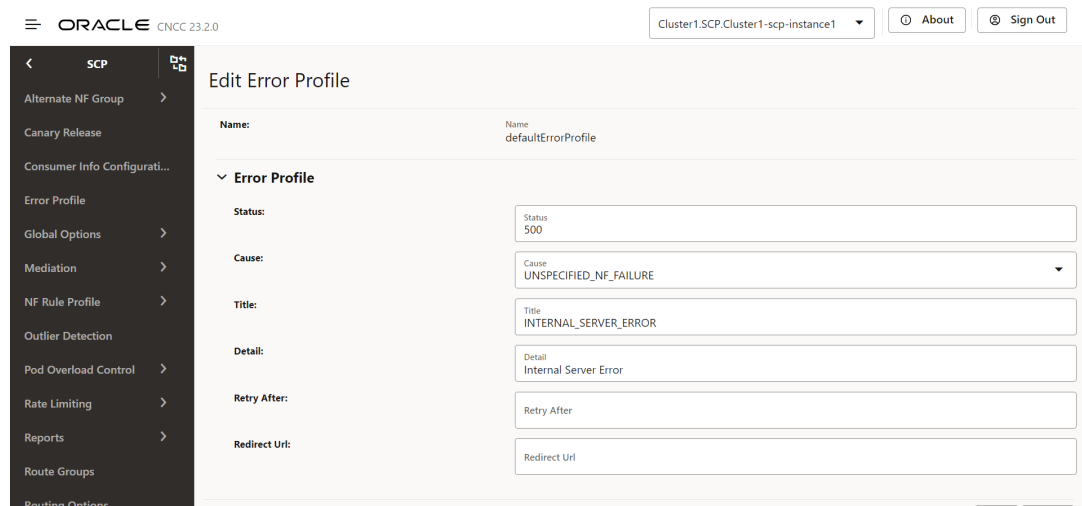
4.2.9 Configuring Error Profile

Perform the following procedure to configure different error profiles, which can be used to build "ProblemDetails" sent in the response body to the consumer.

1. From the left navigation pane, click the **SCP** tab, and then click the **Error Profile** tab.

Figure 4-31 Error Profile

- Click **Edit** icon from the **Actions** column for the error profiles that must be modified. The **Edit Error Profile** screen appears.

Figure 4-32 Edit Error Profile

- If the required cause is not available in the **Cause** drop-down list, select **CUSTOM** to customize the cause:

Figure 4-33 Customize the Cause

The screenshot shows the 'Edit Error Profile' interface in the CNC Console. The left sidebar contains a navigation menu with the following items: SCP, Alternate NF Group, Canary Release, Consumer Info Configurati..., Error Profile (selected), Global Options, Mediation, NF Rule Profile, Outlier Detection, Pod Overload Control, Rate Limiting, Reports, Route Groups, and Routing Options. The main content area is titled 'Edit Error Profile' and shows the configuration for the 'defaultErrorProfile'. The 'Error Profile' section is expanded, revealing the following fields: Name (defaultErrorProfile), Status (500), Cause (CUSTOM), Custom Cause (Custom Cause), Title (INTERNAL_SERVER_ERROR), Detail (Internal Server Error), Retry After, and Redirect Url.

4. Make the required modifications for the following parameters:

Table 4-33 Edit Error Profile

Field	Description
Status	HTTP Status Code
Cause	The list of error causes that are specific to the occurrence of the problem.
Custom Cause	User defined custom cause. This field will be used only if cause field value is set to "CUSTOM".
Title	If this field is null, a standard HTTP status code description is added.
Detail	If present, the same data is used; otherwise, the application can add it optionally.
Retry After	Indicates the number of seconds after client should retry.
Redirect URL	Indicates the AbsoluteURL of the resource to which the message is redirected to.

For information about configuring error profiles, see "Configuring Error Profile" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

5. Click **Save**.

4.2.10 Configuring Canary Release

Following is the procedure to configure Canary Release parameters:

1. From the left navigation menu, navigate to **SCP** and then select **Canary Release**.
2. Click **Edit** from the **Actions** column for the required service that must be modified. The **Edit Canary Release** Screen appears.

Figure 4-34 Canary Release

SCP

Canary Release

Global Options

Mediation

Message Priority

NF Rule Profile

Rate Limiting

Reports

Routing Options

SCP Profile

Service Groups

Topology Source Info

Canary Release

Type to Filter

Refresh

Service Name	Canary Release Flag	API Full Version	Canary Traffic (%)	Actions
n5g-eir-eic	false	2.0.0	5	
namf-comm	false	2.0.0	5	
namf-evts	false	2.0.0	5	
namf-loc	false	2.0.0	5	
namf-mt	false	2.0.0	5	
nausf-auth	false	2.0.0	5	
nausf-someintecion	false	2.0.0	5	

3. Make the required modifications for the following parameters:
- a. **Canary Release Flag:** Enable/Disable the flag using toggle.

b. **API Release Version:** Provide the release version of the API.

c. **Canary Traffic:** Set the value for the traffic that should be distributed to Canary Release.

Figure 4-35 Edit Canary Release

2.0

AboutSign Out

Edit Canary Release

Help

Service Name:

Service Name

n5g-eir-eic

Canary Release Flag:

☐

API Full Version:

API Full Version

2.0.0

Canary Traffic (%):

Canary Traffic (%)

5

Save

Cancel

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

4. Click **Save**.
5. Click **Refresh** to view the updated values on the screen.

4.2.11 Configuring Global Options

Perform the following procedure to configure global options.

Logging Config

1. From the left navigation menu, navigate to **SCP** and select **Global Options**.
2. Select **Logging Config**.
3. Click **Edit** from the top right side to edit or update Logging Config parameters.

Figure 4-36 Logging Config

Service	Application Log Level	
scp-cache	WARN	View
scp-nrfproxy	WARN	View
scpc-alternate-resolution	INFO	View
scp-worker	WARN	View
scpc-notification	INFO	View
scp-nrfproxy-oauth	WARN	View

4. Select **Service Type** from the drop-down list to select any one of the following service type:
 - **scpc-audit**
 - **scpc-alternate-resolution**
 - **scpc-subscription**
 - **scpc-configuration**
 - **scpc-notification**
 - **scp-worker**
 - **scp-cache**
 - **scp-nrfproxy**
5. Select the **Application Log Level** from the drop-down list to set the constraints of the log. The description of each logging level is as follows:

Table 4-34 Logging Level

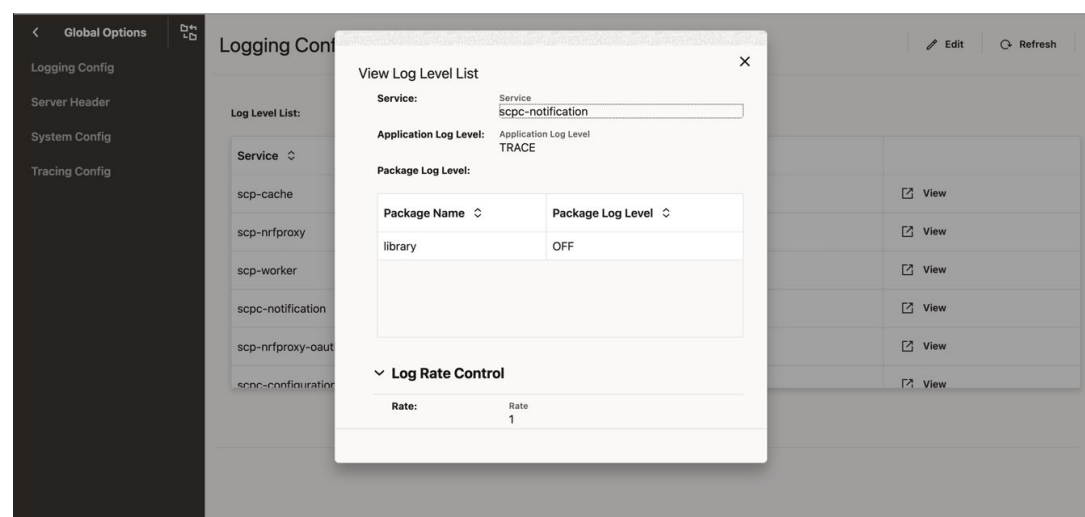
Logging Level	Description
DEBUG	A log level used for events considered to be useful during software debugging when more granular information is required.
INFO	The standard log level indicating that something happened, the application entered a certain state, and so on.

Table 4-34 (Cont.) Logging Level

Logging Level	Description
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 has 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 reaches an issue preventing one or more functionalities from properly functioning.
TRACE	The TRACE log level captures all the details about the behavior of the application. It is mostly diagnostic and is more granular and finer than DEBUG log level. Note: The scpc-notification microservice supports the TRACE log level.

For instance, when you select "scpc-notification" as the service type from the **Service** drop-down list, the **Application Log Level** drop-down list will display the "TRACE" log level.

Figure 4-37 TRACE Log Level



- Click **Add** provided under **Package Log Level** section to specify the name of the package and its log level.
- Click **Save**.

Note

You can also **Edit** and **Delete** the packages under **Package Log Level**.

System Config

Perform the following procedure to edit the system config parameters.

- From the left navigation menu, navigate to **SCP** and select **Global Options**.

2. Select **System Config**.
3. Click **Edit** from the top right side to edit or update **System Config** parameters.

Figure 4-38 System Config

Parameter	Value
OD Enabled:	OD Enabled false
Scp Rule Name:	Scp Rule Name nextHopScpODRule
OD Sepp Enabled:	OD Sepp Enabled false
Sepp Rule Name:	Sepp Rule Name defaultRule
CB Scp Enabled:	CB Scp Enabled false
CB Scp Rule Name:	CB Scp Rule Name defaultRule
CB Sepp Enabled:	CB Sepp Enabled false
CB Sepp Rule Name:	CB Sepp Rule Name defaultRule
Forward Nf Discovery Headers:	Forward Nf Discovery Headers true
TTL:	TTL

4. Update the attributes as described in "Configuring SystemOptions" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Note

Ensure that **Forward Nf Discovery Headers** is set to true to forward discovery headers to Nexthop SCP as part of the Model D Indirect 5G SBI Communication feature.
The default value is true.

5. Click **Save** after performing the modifications.

Tracing Config

Perform the following procedure to edit tracing config options.

1. From the left navigation menu, navigate to **SCP** and select **Global Options**.
2. Select **Tracing Config**.
3. Click **Edit** from the top right side to edit or update **Tracing Config** options.

Figure 4-39 Tracing Config

Service Name	Tracing Enabled	Message JSON Body Enabled	Actions
scp-worker	false	false	

4. Set the following switches as required:

- **Tracing Enabled**
- **Message JSON Body Enabled**

For information about tracing parameters, see "Updating HELM Configurable Parameters with REST APIs" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

5. Click **Save** after performing the modifications.

4.2.12 Configuring Mediation

Perform the following procedure to configure trigger points for defining the filter criteria for a message to decide whether to invoke mediation or not.

1. From the left navigation pane, click the **SCP** tab, and then click the **Mediation** tab. Trigger Points and Routing Options tabs appear.
2. To configure trigger points for defining the filter criteria for a message, in the left navigation pane, click the **Trigger Points** tab.
3. In the Trigger Points section, click **Add** and configure the fields as described in the following table:

Table 4-35 Trigger Points Field Description

Field	Description
Group Id	Indicates the group ID for which mediation configuration is required. The Mediation configuration for a specific group is applicable to the mediation requests or responses received only from the same group. The HTTP Mediation service consumer NFs, which requires the same mediation rules, can be grouped together using this group ID.
Rule Name	Indicates the unique rule name for each mediation configuration that is considered as a primary key.
Service Name	Indicates the service of NFType for which mediation configuration is required.
NF Type	Indicates the NFType for which mediation configuration is required.
Http Methods	Indicates the HTTP methods such as GET, POST, PUT, PATCH, DELETE, and OPTION.
Mediation Trigger Points	Indicates the list of trigger points to be enabled if matches one or more of the following: requestIngress, requestEgress, responseIngress, and responseEgress.
Match	Indicates the list of match blocks to be satisfied for the rule to be activated. The list of match blocks have OR semantics. Minimum number of blocks = 1 Maximum number of blocks = 20
Add	Adds HTTP headers and JSON body criteria

4. In the Match section, click **Add**. The Add Match dialog box appears.

Figure 4-40 Add Match

Add Match

Header:

Name ↑↓	Value ↑↓	⊕ Add
No data to display.		

Body:

Name ↑↓	Value ↑↓	⊕ Add
No data to display.		

Save

Cancel

5. In the Header section, click **Add** to set the header details as follows:
 - **Name:** Enter a name for the header.
 - **Value:** Enter a value of the header.
 - **Match Type:** Select a range type from the **Match Type** drop-down list.
6. In the Range section, if the header type has a numeric value, enter the range of that value in **Start** and **End** fields.
7. Click one of the following buttons:
 - **Remove:** To discard the changes made to the Range section.
 - **Save:** To save the changes.
8. In the Body section, click **Add Body** to set the JSON body details as follows:
 - **Name:** Enter a name for the body.
 - **Value:** Enter a value for the header.
 - **Match Type:** Select a range type from the **Match Type** drop-down list.
9. In the Range section, if the header type has a numeric value, enter the range of that value in **Start** and **End** fields.
10. Click one of the following buttons:
 - **Remove:** To discard the changes made to the Range section.

- **Save:** To save the changes.
- **Cancel:** To reset all the fields.

4.2.12.1 Configuring Routing Options for Mediation

Perform this procedure to configure routing options with Mediation.

1. From the left navigation pane, click the **SCP** tab, and then click the **Routing Options** tab.
2. In the Routing Options section, click **Edit**.

The Edit Routing Options screen appears.

Figure 4-41 Edit Routing Options

Edit Routing Options

App Name: mediation

Routing Options

Retry: ☒

Max Retry Attempts: 2

Response Timeout:

Exception Error Responses:

Status Code ↑↓	Action ↑↓	
DEFAULT	continue_processing	⊕ Add ✎ Edit 🗑 Delete

3. In the Routing Options section, set **Retry** to true to enable retry attempt.
4. In the **Max Retry Attempts** field, enter a value to set the number of retry attempts.
5. In the **Response Timeout** field, enter a time interval to respond to a message request.
6. In the Exception Error Responses section, click **Add** to add exception error responses.

This option enables you to send an error message to the consumer NF or process the message request to the producer NF when the Mediation service is unavailable. By default, this status code is DEFAULT.

7. In the Add Exception Error Responses dialog box, configure the fields as described in the following table:

Table 4-36 Add Exception Error Responses

Field	Description
Status Code	Indicates the list of status codes that matches with the status code received from the application, the required action specified by action field is taken. This value can be any HTTP Status Codes, HTTP Status with other defined custom status (RESPONSE_TIMEOUT and CONNECTION_FAILURE)
Action	Indicates the action to be taken when the above mentioned status code is received. It can have the following values: - send_error_response: Sends the error code to the consumer NF. - continue_processing: Continues processing the message in case of errors.
Error Code	Indicates configurable error codes to be sent by SCP to consumer NFs. It is a valid HTTP status code or allowed custom code.
Error Cause	Indicates the error cause that is specific to the occurrence of the problem.
Error Title	Indicates the title of the error.
Error Description	Indicates an explanation specific to the occurrence of the problem.
Retry After	Indicates the retry interval.
Redirect Url	Indicates the AbsoluteURL of the resource to which the message is redirected to.

8. Click **Save**.

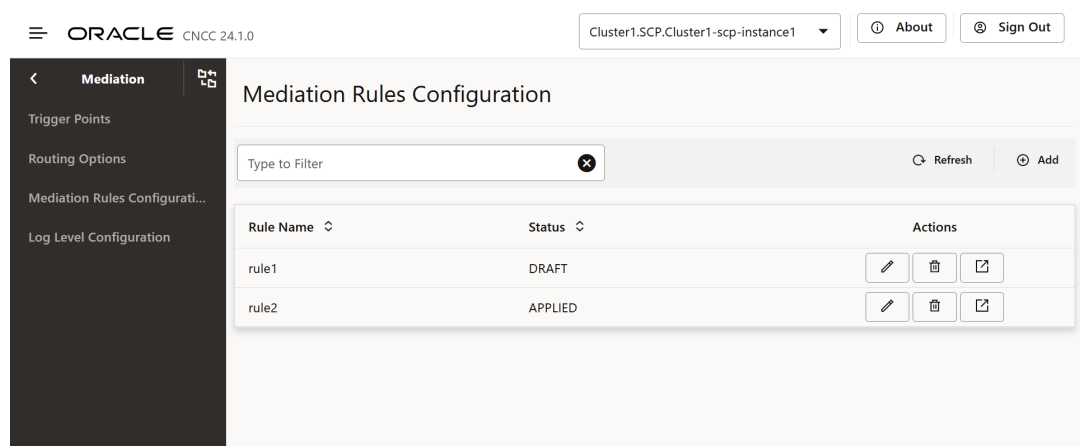
4.2.12.2 Configuring Mediation Rules

Perform this procedure to configure mediation rules.

1. From the left navigation pane, click the **SCP** tab, and then click the **Mediation** tab.
2. On Mediation tab, click the **Mediation Rules Configuration** subtab.

The Mediation Rules Configuration screen with a list of rules saved in database appears.

Figure 4-42 Mediation Rules Configuration



3. Click the **View** icon to view the mediation rules.
4. Click the **Add** to create a mediation rule.

The Add Mediation Rule screen allows you to configure the fields to create new rules as described in the following table:

Table 4-37 Add Mediation Rules

Field	Description
Rule Name	Indicates the name of the rule. This value should be unique.
Format	Indicates the Format in which the rules are defined, such as DRL.
Status	Indicates the status of the rule. It can have the following values: • DRAFT • APPLIED The status must always be in DRAFT to create a new rule.
Mediation Mode	Indicated whether the rule applies to mediation active or mediation test. • Mediation Active allows actual messages to be manipulated according to configured mediation rules. • Mediation Test executes the mediation rule on message copy instead of the actual message and verifies the behavior of the mediation rule.
Code	Indicates the configurable drool expression that matches the request/response headers and body sent by SCP. For example, <pre> package com.oracle.cgbu.ocmediation.nfmediation; import com.oracle.cgbu.ocmediation.factdetails.Request; dialect "mvel" rule "newRuleTest" when req : Request(headers.has("header") == true) then req.headers.add("NewHeader","132465") end </pre>
State	Indicates the state. The value of State must be SAVE to create a new rule. For more information, see Table 3-20

Note

Special characters are not supported and can lead to a compilation error.

5. Click the **Edit** icon to edit the existing rule.

The Edit Mediation Rule screen allows you to configure the fields as described in the following table:

Table 4-38 Edit Mediation Rules

Field	Description
Rule Name	Indicates the name of the rule. This value should be unique.
Format	Indicates the Format in which the rules are defined, such as DRL.

Table 4-38 (Cont.) Edit Mediation Rules

Field	Description
Status	Indicates the status of the rule. It can have the following values: • DRAFT • APPLIED The following fields are allowed to be modified in DRAFT status: Mediation Mode, Code, and State. The valid values for state are SAVE, CLONE, COMPILE, and APPLY. The Mediation Mode can only be modified in APPLIED status. The valid values for state are SAVE, CLONE, or DRAFT.
Mediation Mode	Indicated whether the rule applies to mediation active or mediation test. • Mediation Active allows actual messages to be manipulated according to configured mediation rules. • Mediation Test executes the mediation rule on message copy instead of the actual message and verifies the behavior of the mediation rule.
Code	Indicates the configurable drool expression that matches the request/response headers and body sent by SCP.
State	Indicates the action to be applied to the rule. The default value is SAVE Note: The DRAFT state is an invalid transition that could lead to an error.
New Rule Name	Indicates the new name of the rule. This field appears when a CLONE state is selected.

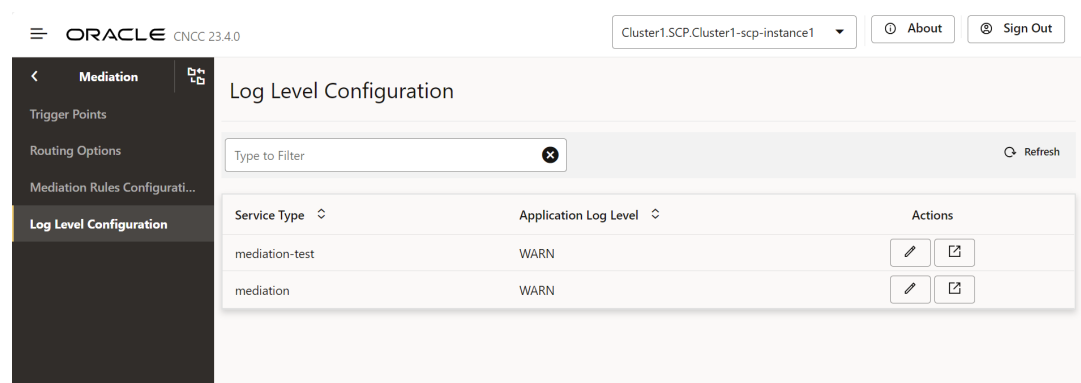
- Click the **Delete** icon to delete the existing rule.
The status of the rule must always be in DRAFT to delete an existing rule.
- Click **Save**.

4.2.12.3 Configuring Mediation Log Levels

Perform this procedure to configure the log levels of mediation.

- From the left navigation pane, click the **SCP** tab, and then click the **Mediation** tab.
- On Mediation tab, click the **Log Level Configuration** subtab.
The Log Level Configuration screen with a list of default log level type of mediation and mediation-test service assigned during deployment appears.

Figure 4-43 Log Level Configuration



3. Click the **View** icon to view the log level of mediation or mediation test service.
4. Click the **Edit** icon to edit or update the log level of mediation or mediation test service. The Edit Log Level Configuration screen allows you to configure the fields as described in the following table:

Table 4-39 Edit Log Level Configuration

Field	Description
Service Type	Indicates the name of the type of service. For examples, mediation and mediation-test
Application Log Level	Indicates the type of log level. The following are the supported log levels: • WARN: Designates potentially harmful situations. • INFO: Designates informational messages that highlight the progress of the application at coarse-grained level. • ERROR: Designates error events that might still allow the application to continue running. • DEBUG: Designates fine-grained informational events that are most useful to debug an application. The default value is WARN .
Package Log Level	Indicates the package level log information.
Log Rate Control	Indicates the log rate control information.

5. Select the application log level from the **Application Log Level** drop-down list to collect the logs as per the requirement.
6. Click **Save** to update the log level of service.

Package Log level Configuration

To update the package level information, perform the following:

1. On the Package Log level section, click the **EDIT** icon next to the package name. The Edit Package Log Level screen allows you to configure the fields as described in the following table:

Table 4-40 Edit Package Log Level

Field	Description
Package Name	Indicates the name of the package. It should be a valid library name. • The valid package names are org, io, springfox, com.zaxxer, reactor, validator, and library. • The default package name is library.
Package Log Level	Indicates the type of package log level. The following are the supported log levels: • WARN: Designates potentially harmful situations. • INFO: Designates informational messages that highlight the progress of the application at a coarse-grained level. • ERROR: Designates error events that might still allow the application to continue running. • DEBUG: Designates fine-grained informational events that are most useful to debug an application • OFF: Disables the log rate. The default value is OFF

2. Edit the package name and Package Log Level from the drop-down menu and click **Save**.

Log Rate Control Configuration

To configure the average number of logs allowed per second, perform the following:

1. The Log Rate Control section allows you to configure the fields as described in the following table:

Table 4-41 Log Rate Control

Field	Description
Rate	Indicates the average number of logs allowed per second. • The default rate is 1. • The range is in between 1 and 10000
Log Level	Indicates the type of log level where rate control is applied. The following are the supported log levels: • WARN: Designates potentially harmful situations. • INFO: Designates informational messages that highlight the progress of the application at a coarse-grained level. • ERROR: Designates error events that might still allow the application to continue running. • DEBUG: Designates fine-grained informational events that are most useful to debug an application • OFF: Designates to turn off logging. The default value is OFF

2. Select the log level from the **Log Level** drop-down to change the log level type, and click **Save**.

4.2.13 Configuring NF Rule Profile

Perform the following procedure to configure NF Rule Profile parameters.

Note

When the learning source is set to "Local", users can modify the 5G NF profile regardless of whether it was learned from NRF (source='NRF') or configured statically (source='Local'). By setting the source to Local, the system does not delete already determined profiles.

1. From the left navigation, navigate to **SCP**, and then click **NF Rule Profile**.
2. To add or configure an NF rule profile, click the **NF Rule Profile Configuration** tab in the left navigation pane.

Figure 4-44 NF Rule Profile Configuration

NF Instance Id	NF Type	NF Status	FQDN	Actions
3fa2bbc-6e4a-2828-a507-a14ef8e1bc7b	NRF	REGISTERED		
3fa2bbc-6e4a-2828-a507-b14ef8e1bc7a	NRF	REGISTERED		
3fa3bbc-6e4a-2828-a507-a14ef8e1bc6a	NRF	REGISTERED		
3fa3bbc-6e4a-2828-a507-a14ef8e1bc6b	NRF	REGISTERED		
6faf1bbc-6e4a-2828-a507-a14ef8e1bc5a	NRF	REGISTERED		
6faf1bbc-6e4a-2828-a507-a14ef8e1bc5b	NRF	REGISTERED		
6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e	SCP	REGISTERED	ocsdp-scp-worker.scpssc.svc.cluster.local	

3. Click **Add** to add NF rule profile.

Figure 4-45 Add NF Rule Profile

The screenshot shows the 'Add NF Rule Profile Configuration' form. The sidebar on the left has a dark background with the text 'NF Rule Profile' and 'NF Rule Profile Data'. The main form area has a light background and contains the following fields:

Add NF Rule Profile Configuration	
NF Instance Id:	NF Instance Id
NF Type:	NF Type
NF Status:	NF Status
FQDN:	FQDN
Inter PLMN FQDN:	Inter PLMN FQDN
IPv4 Addresses:	IPv4 Addresses
IPv6 Addresses:	IPv6 Addresses
Priority:	Priority
Capacity:	Capacity
Load:	Load

4. Add the following parameters for all the NF Info:
 - **Group Id:** Identity of the NF info.
 - **Routing Indicators:** List of Routing Indicator information.
 - Click **Add** to add the SUPI Ranges.
 - Click **Save** to save the SUPI Ranges.
 - Click **Remove** if you want to remove the NF Info.
 - Click **Add** to add the NF info details.
5. Click **Save** to save the NF Rule Info.
6. Click **Refresh** to view the updated values on the screen.
7. To configure a local PLMN, in the PLMN List section, click **Add** and enter the required Mobile Country Code (MCC) and Mobile Network Code (MNC) values in the displayed fields.

Modifying IP Addresses

Perform the following procedure to edit the IP addresses which are present under NF service level:

1. In the NF Service section, to modify any NF service, click the corresponding **Edit** icon.
2. In the IP End Points section, click **Add** to add IP end points.

Figure 4-46 Adding IP End Points

Add IP End Points [X]

IPv4Address:

IPv6Address:

Port:

[Save] [Cancel]

3. Enter IP address in one of the following fields:

- **IPv4Address**
- **IPv6Address**

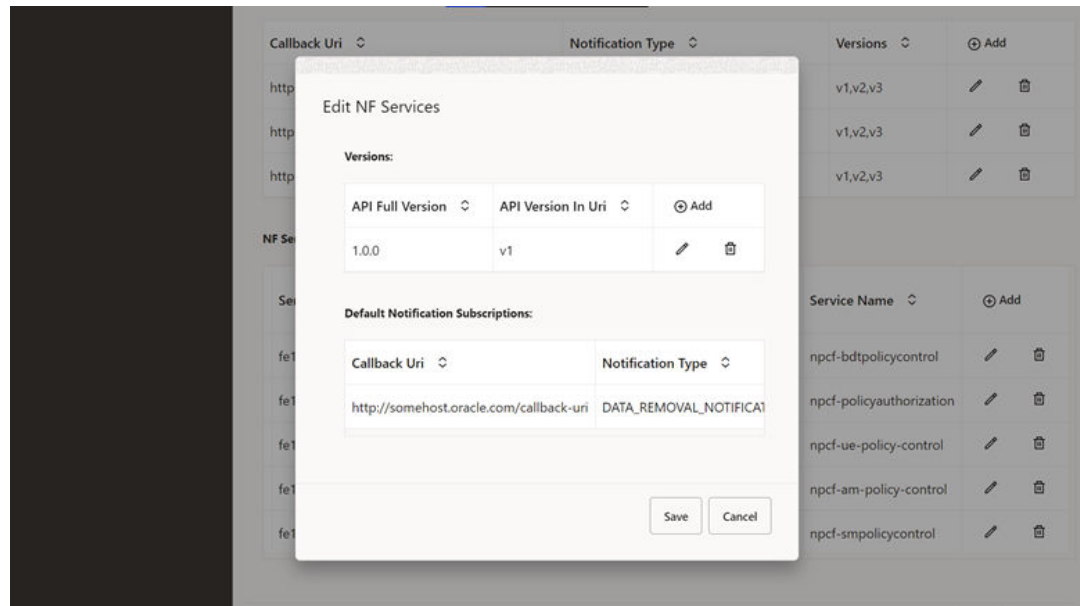
Note

The system displays an error message if you enter IP address in both the above mentioned fields.

For more information about **IPv4Address** and **IPv6Address** configurations, see "Configuring Route Groups" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

4. In the **Port** field, enter the port number.
5. Click the **Edit** icon to edit or add the API full versions.

Figure 4-47 Edit NF Services



6. Click **Save** to save the NF Rule Info.

Viewing or Editing the Default Notification Subscriptions

Perform the following procedure to view or edit the default notification subscription which are present under NF service level or NF profile level:

Note

Users cannot modify values for profiles in Default Notification Subscription if the learning source is set to NRF. They can only view it. Editing is only possible when the learning source is set to "Local".

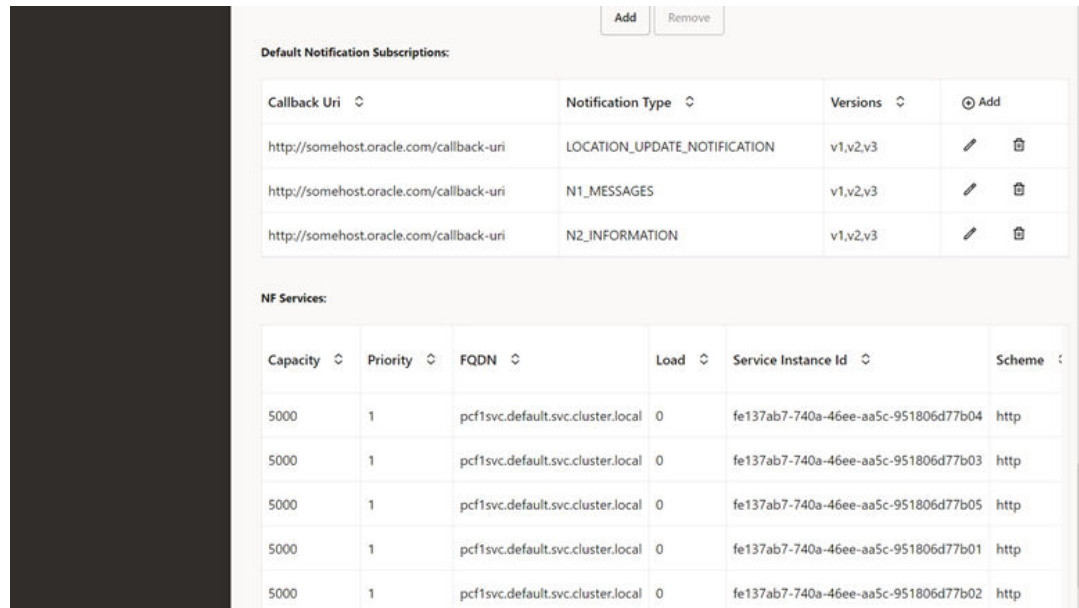
1. In the NF Rule Profile Configuration section, click **Edit** from Actions. The **Edit NF Rule Profile Configuration** screen appears.

Figure 4-48 Edit NF Rule Profile Configuration

Edit NF Rule Profile Configuration

NF Instance Id:	NF Instance Id 7faf2bbc-6e4a-2828-a507-a14ef8e1bc7b
NF Type:	NF Type UDM
NF Status:	NF Status REGISTERED
FQDN:	FQDN
Inter PLMN FQDN:	Inter PLMN FQDN nrfr2l2.5gc.mnc215.mcc410.3gppnetwork.org
IPV4 Addresses:	IPV4 Addresses
IPV6 Addresses:	IPV6 Addresses
Priority:	Priority 10
Capacity:	Capacity 10000
Load:	Load
Locality:	Locality Loc2r2

2. In the Default Notification Subscription section, click the **Edit** icon for the required callback URI that must be modified. The **Edit Default Notification Subscription** screen appears.

Figure 4-49 Default Notification Subscription


The screenshot displays the 'Default Notification Subscriptions' section in the CNC Console. It features a table with columns for 'Callback Uri', 'Notification Type', 'Versions', and 'Add'. Below this is the 'NF Services' section, which includes a table with columns for 'Capacity', 'Priority', 'FQDN', 'Load', 'Service Instance Id', and 'Scheme'.

Callback Uri	Notification Type	Versions	Add
http://somehost.oracle.com/callback-uri	LOCATION_UPDATE_NOTIFICATION	v1,v2,v3	
http://somehost.oracle.com/callback-uri	N1_MESSAGES	v1,v2,v3	
http://somehost.oracle.com/callback-uri	N2_INFORMATION	v1,v2,v3	

Capacity	Priority	FQDN	Load	Service Instance Id	Scheme
5000	1	pcf1svc.default.svc.cluster.local	0	fe137ab7-740a-46ee-aa5c-951806d77b04	http
5000	1	pcf1svc.default.svc.cluster.local	0	fe137ab7-740a-46ee-aa5c-951806d77b03	http
5000	1	pcf1svc.default.svc.cluster.local	0	fe137ab7-740a-46ee-aa5c-951806d77b05	http
5000	1	pcf1svc.default.svc.cluster.local	0	fe137ab7-740a-46ee-aa5c-951806d77b01	http
5000	1	pcf1svc.default.svc.cluster.local	0	fe137ab7-740a-46ee-aa5c-951806d77b02	http

3. Make the required modifications for the following fields:

Table 4-42 Default Notification Subscriptions

Attribute name	Description
Callback Uri	This attribute contains a default notification endpoint to be used by a NF service producer towards an NF service consumer that has not explicitly registered a callback URI with the NF service producer, for example, as a result of an implicit subscription.
Notification Type	Type of notification for which the corresponding callback URI is provided.
Versions	API versions, for example, v1, are supported for the default notification type.
N1 Message Class	If the notification type is N1_MESSAGES, this IE must be present to identify the class of N1 messages to be notified.
N2 Information Class	If the notification type is N2_INFORMATION, this IE must be present to identify the class of N2 information to be notified.
Binding	When present, this IE contains the value of the binding indication for the default subscription notification, which is the value part of the 3gpp-Sbi-Binding header. See the note for details.

Note

When delivering a notification for a default subscription to a specific NF consumer but the latter is not reachable, a NF service producer uses the binding indication for default subscription to reselect an alternative NF service consumer instance. For example, an AMF notifies corresponding uplink LPP or NRPPa messages through a default subscription to the LMF instance that previously sent downlink LPP or NRPPa messages during a location procedure. If the original LMF instance is not reachable, the AMF selects an alternative LMF instance using the binding indication and delivers the notification to the selected LMF instance.

Figure 4-50 Edit Default Notification Subscriptions

4. Click **Save** to save the NF Rule Info.
5. Click **Refresh** to view the updated values on the screen.

4.2.13.1 Fetching NF Rule Profile Data

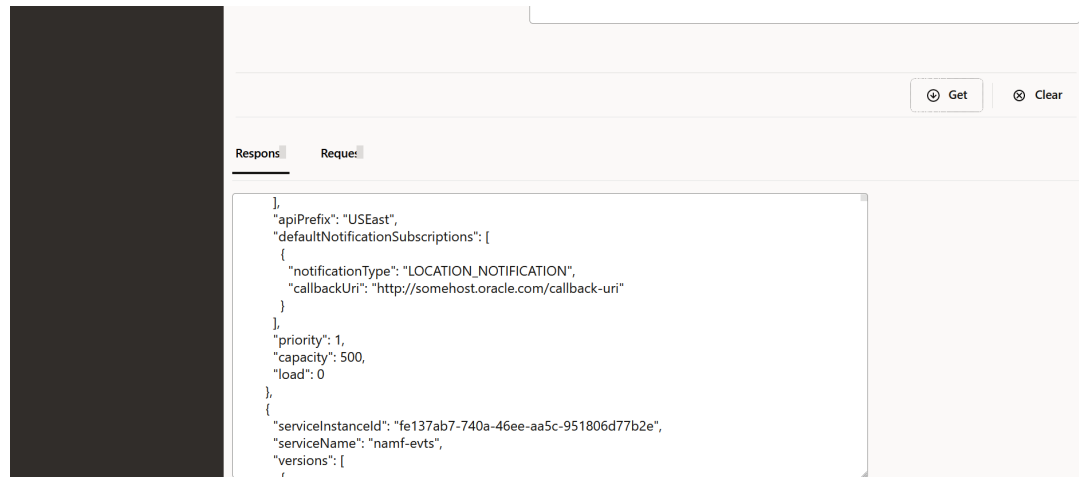
Following is the procedure to retrieve a collection of NF rule profile data configured at SCP:

1. From the left navigation menu, navigate to **SCP** and then click **NF Rule Profile**.
2. To retrieve the default notification subscription data for a specific NF rule profile, click on the **NF Rule Profile Data** tab in the left navigation pane.
3. The NF Rule Profile Data screen is displayed as follows:

Figure 4-51 NF Rule Profile Data

4. Enter the **Query Parameters** in the field. These parameters may include NFType, ServiceName, NF instance ID, or Maximum Number of NF Profiles. For more information, see the "Configuring NF Topology Groups" section in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
5. Click **Get** to fetch the response data. The data is displayed in the **Response** pane.

Figure 4-52 Response data



6. Click **Clear** to clear the response data.

4.2.14 Configuring Routing Options

Perform the following procedure to configure Routing Options parameters.

1. From the left navigation menu, navigate to **SCP**, and then click the **Routing Options** tab.
2. In the Routing Options section, click **Edit** from the Actions column for the required service that must be modified.
The Edit Routing Options screen appears.

Figure 4-53 Edit Routing Options

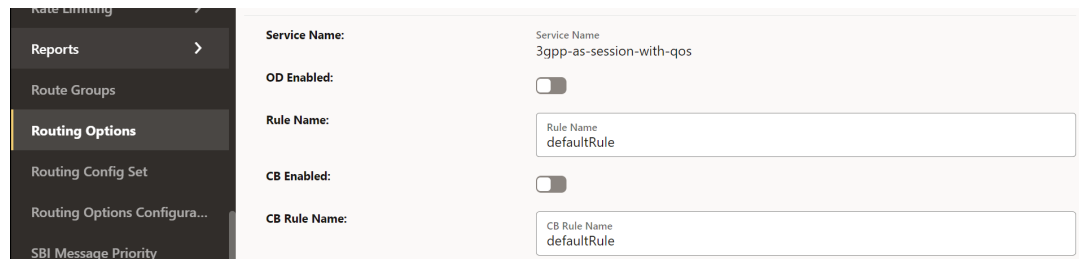
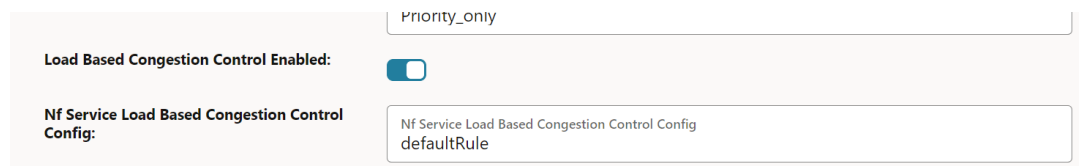


Figure 4-54 Load based congestion control enabled



3. Enter or modify the required values as described in "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Note

Ensure that the following options are set as required to manage the "3gpp-Sbi-Discovery-preferred-locality" header as part of the Model D Indirect 5G SBI Communication feature:

- **Assign Preferred Locality**
- **Override Preferred Locality**
- **Forward Revised Preferred Locality**

4. In the Reroute Condition List, configure the list of error codes for rerouting:
 - Click **Add** to add new status code.
 - Click **Edit** or **Delete** icon to edit or remove any existing error code from the reroute condition list.
5. In the Alternate Route section, set the **Mode** of the specific services as required:
 - a. **NF_SET**: SCP performs alternate routing based on the Model C Indirect 5G SBI Communication format. This is the default mode.
 - b. **DNS_SRV**: SCP performs alternate routing based on the DNS SRV query.
 - c. **NF_SET_FOLLOWED_BY_DNSSRV**: SCP performs alternate routing based on the NF Set, followed by DNS SRV.
 - d. **STATIC_CONFIG**: SCP performs alternate routing based on the static configuration.
 - e. **NF_SET_FOLLOWED_BY_STATIC_CONFIG**: SCP performs alternate routing based on NF Set, and then using static configuration.
6. Click **Save**.
7. Click **Refresh** to view the updated values on the screen.

4.2.15 Configuring SCP Profile

Following is the procedure to configure SCP Profile parameters:

Editing SCP Profile

1. From the left navigation menu, navigate to **SCP** and then select **SCP Profile**.

Figure 4-55 SCP Profile

SCP Profile

FQDN:

FQDN

NF Instance Id:

NF Instance Id

NF Status:

NF Status

NF Type:

NF Type

Locality:

Locality

NF Service Persistence:

NF Service Persistence
false

Custom Info

Remaining Localities:

Remaining Localities

Serving Localities:

Serving Localities

Mate SCP Info

NF Services

Capacity

↑↓

FQDN

↑↓

Load

↑↓

Port

↑↓

Priority

↑↓

Scheme

↑↓

No data to display.

2. Click **Edit** under **NF Services** section. The **Edit SCP Profile** Screen appears. For more information about configuration parameters, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Note

To add a new location to Remaining Localities and Serving Localities, type the new location name, and then click **Enter**.

Note

Use **Edit** or **Delete** buttons available in the next column to update or delete the NF services.

4.2.16 Configuring Topology Source Info

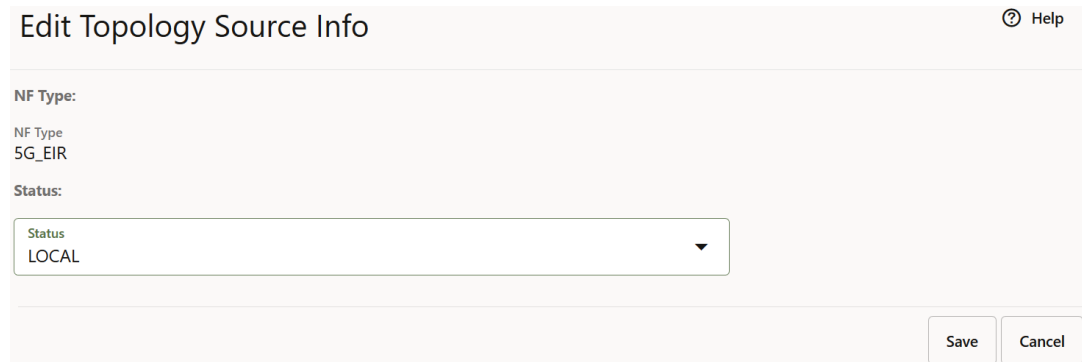
Following is the procedure to configure Topology Group Info parameters:

1. From the left navigation menu, navigate to **SCP** and then select **Topology Source Info**.

Figure 4-56 Topology Source Info

NF Type	Status	Actions
5G_EIR	NRF	
AF	NRF	
AMF	NRF	
AUSF	NRF	
BSF	NRF	
CHF	NRF	
CUSTOM_ORACLE_SCP	NRF	
GMLC	NRF	
LMF	NRF	

2. Click **Edit** from the **Actions** column for the required service that must be modified. The **Edit Topology Source Info** Screen appears.
3. Update the **Status** of the specific **NF Type** in the **Edit Topology Source Info** as per your requirement.

Figure 4-57 Edit Topology Source Info

Edit Topology Source Info

NF Type:

NF Type
5G_EIR

Status:

Status
LOCAL

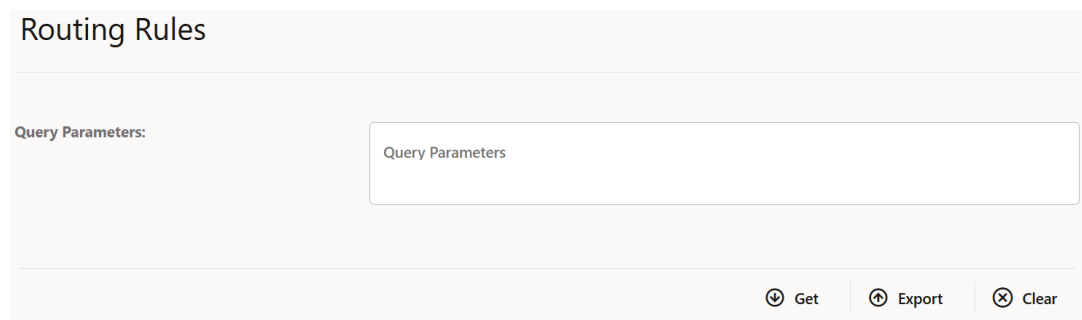
Save Cancel

4. Click **Save** to save the **Topology Source Info**.
5. Click **Refresh** to view the updated values on the screen.

4.2.17 Fetching Routing Rules

Following is the procedure to retrieve a collection of routing rules configured at SCP:

1. From the left navigation menu, navigate to **SCP** and then select **Reports**.
2. Select **Routing Rules**. The **Routing Rules** screen is displayed as follows:

Figure 4-58 Routing Rules

Routing Rules

Query Parameters:

Query Parameters

Get Export Clear

3. Enter the **Query Parameters** in the field.
4. Click **Get** to fetch the response data. The data is displayed in the **Response** pane.

Figure 4-59 Response data

Routing Rules

Query Parameters:

Query Parameters

Get Export Clear

Responses

```
[
  {
    "spec": {
      "hosts": [
        "12.34.56"
      ],
      "http": [
        {
          "match": [
            {
              "uri": {
                "prefix": "/"
              },
              "app_name": "END_NODE"
            }
          ]
        }
      ]
    }
  }
]
```

Note

In case the response data is above the configured display limit, a message is displayed stating *"Response data has crossed the configured display limit (5 MB), please click on **Export** to download it as a file."*. Currently, the display limit cannot be modified, it is set to 5 MB.

5. Click **Export** to download of the response data file.
6. Click **Clear** to clear the response data.

4.2.18 Configuring Ingress Rate Limiting

Following is the procedure to configure ingress rate limiting parameters:

1. From the left navigation menu, navigate to **SCP** and then select **Rate Limiting**.
2. Select **Ingress Rate Limiting**. The screen is displayed.

Figure 4-60 Ingress Rate Limiting

The screenshot shows the 'Ingress Rate Limiting' configuration page. On the left, a sidebar menu has 'Rate Limiting' expanded, with 'Ingress Rate Limiting' selected. The main content area has a header 'Ingress Rate Limiting'. Below it, there is a 'Query Parameters' label and a text input field. At the bottom, there is a row of action buttons: Get, Add, Edit, Delete, Export, and Clear.

3. Enter the **Query Parameters** in the field.
4. Click **Get** to fetch the response data. The data is displayed in the **Response** pane.

Note

In case the response data is above the configured display limit, a message is displayed stating "Response data has crossed the configured display limit (5 MB), please click on **Export** to download it as a file.". Currently, the display limit cannot be modified, it is set to 5 MB.

5. Click **Add** to add the ingress rate limiting configuration using the Request Body.
6. Click **Edit** to edit the Request Body of the ingress rate limiting configuration.
7. Click **Delete** to delete the ingress rate limiting configuration based on the query parameters.

4.2.19 Configuring Egress Rate Limiting

Following is the procedure to configure egress rate limiting parameters:

1. From the left navigation menu, navigate to **SCP** and then select **Rate Limiting**.
2. Select **Egress Rate Limiting**. The screen is displayed.

Figure 4-61 Egress Rate Limiting

The screenshot shows the 'Egress Rate Limiting' configuration page. On the left, a sidebar menu has 'Rate Limiting' expanded, with 'Egress Rate Limiting' selected. The main content area has a header 'Egress Rate Limiting'. Below it, there is a 'Query Parameters' label and a text input field. At the bottom, there is a row of action buttons: Get, Add, Edit, Delete, Export, and Clear.

3. Enter the **Query Parameters** in the field.
4. Click **Get** to fetch the response data. The data is displayed in the **Response** pane.

Note

In case the response data is above the configured display limit, a message is displayed stating "Response data has crossed the configured display limit (5 MB), please click on **Export** to download it as a file.". Currently, the display limit cannot be modified, it is set to 5 MB.

5. Click **Add** to add the egress rate limiting configuration using the Request Body.
6. Click **Edit** to edit the Request Body of the egress rate limiting configuration.
7. Click **Delete** to delete the egress rate limiting configuration based on the query parameters.

4.2.20 Configuring Server Header

Perform the following procedure to configure server header parameters to include server header in the producer NF error responses.

1. From the left navigation pane, click the **SCP** tab, and then click the **Global Options** tab.
2. Click the **Server Header** tab.
3. In the Server Header section, click **Edit**.

The Edit Server Header screen appears.

Figure 4-62 Edit Server Header

Configuration Item	Value
Enable Enhance Server Header Behavior:	false
Side Car Proxy Server Header:	en*y
Add Server Header In Producer Response:	true
Enable Enhance Server Header Behavior V2:	true
Side Car Proxy Server Header Status Code:	503,4xx

4. To enable the enhanced server header behavior introduced in SCP 1.15.0, set **Enable Enhance Server Header Behavior** to true.

By default, this option is disabled. For information about server header configurations, see "Configuring Server Header" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

5. In the **Side Car Proxy Server Header** field, enter the strings to identify server header values if error response is generated by side car proxy.

A sample value is envoy.

6. To enable addition of server header in producer NF generated error responses, set **Enable Enhance Server Header Behavior** to true.

By default, this option is disabled.

7. To enable the enhanced server header behavior introduced in SCP 23.1.0, set **Enable Enhance Server Header Behavior V2** to true.

8. Enter status codes in the **Side Car Proxy Server Header Status Code** field.
9. Click **Save**.

4.2.21 Configuring Pod Overload Control Policies

Perform the following procedure to configure pod overload policies to control and discard request messages sent to scp-worker.

For more information about pod overload policies, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

1. From the left navigation pane, navigate to **SCP** and click **Pod Overload Control**, and then click **Worker**.
2. To configure Pod Overload Control Policy, in the left navigation pane, click the **Overload Control Policy** tab.
 - a. In the Overload Control Policy section, click **Get** to retrieve the scp-worker Pod Overload Control Policy data configured based on the threshold level.

Figure 4-63 Overload Control Policy

The screenshot shows the 'Overload Control Policy' page. On the left, a navigation pane has 'Worker' selected. The main content area has a title 'Overload Control Policy'. Below the title is a 'Query Parameters' section with a text input field containing 'thresholdLevel=CRITICAL'. To the right of this field are buttons: 'Get', 'Add', 'Edit', 'Delete', 'Export', and 'Clear'. Below these buttons is a 'Response' field showing a JSON object:

```
{
  "thresholdLevel": "CRITICAL",
  "data": {
    "cpuOverloadConfig": {
      "onSetThreshold": 95,
      "abatementThreshold": 92,
      "abatementTimeInMilliseconds": 90
    },
    "pendingTransactionOverloadConfig": {
      "onSetThreshold": 95,
      "abatementThreshold": 92,
      "abatementTimeInMilliseconds": 50
    }
  }
}
```

- b. To add a new request, click **Add** and enter the request body information in the **Request** field.
- c. Click **Submit**.
The request body message appears in the **Response** field.
- d. To edit, export, delete, or clear the data, click the required buttons as described in the following table:

Table 4-43 Buttons and Field Descriptions on Overload Control Policy page

Button and Field Names	Description
Query Parameters	Allows to enter parameters to retrieve, add, and update different pod overload policies data.
Get	Retrieve the scp-worker Pod Overload Policy data configured based on the threshold level. You can modify this data using the Edit button.

Table 4-43 (Cont.) Buttons and Field Descriptions on Overload Control Policy page

Button and Field Names	Description
Add	Adds new message requests in the Request field.
Edit	Modifies any existing data by updating mandatory query parameters in the Query Parameters field.
Delete	Removes any message request.
Export	Exports existing records in the JSON format.
Clear	Refreshes the existing screen by removing the data.

3. To configure Pod Overload Action Policy, in the left navigation pane, click the **Overload Action Policy** tab.
 - a. In the Overload Action Policy section, click **Get** to retrieve the scp-worker Pod Overload Action Policy data configured based on the threshold level.
 - b. To add a new request, click **Add** and enter the request body information in the **Request** field.
 - c. Click **Submit**.
The request body message appears in the **Response** field.
 - d. To edit, export, delete, or clear the data, click the required buttons as described in [Table 4-43](#).
4. To configure Pod Overload Discard Policy, in the left navigation pane, click the **Overload Discard Policy** tab.
 - a. In the Overload Discard Policy section, click **Get** to retrieve the scp-worker Pod Overload Discard Policy data configured based on the threshold level.
 - b. To add a new request, click **Add** and enter the request body information in the **Request** field.
 - c. Click **Submit**.
The request body message appears in the **Response** field.
 - d. To edit, export, delete, or clear the data, click the required buttons as described in [Table 4-43](#).

4.2.22 Configuring SBI Message Priority

Perform the following procedure to configure Service Based Interface (SBI) Message Priority parameters supported by the Pod Overload Control feature. You can retrieve SCP_SBI Message Priority configured in SCP and modify the data. Also, add a new SBI message request.

For more information about SBI Message Priority parameters, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

1. From the left navigation pane, navigate to **SCP** and click the **SBI Message Priority** tab.
2. In the **SBI Message Priority** section, click **Get** to retrieve the SCP_SBI Message Priority configured in SCP.

Figure 4-64 SBI Message Priority

The screenshot shows the 'SBI Message Priority' configuration page. On the left is a sidebar with a menu including 'SCP', 'Alternate NF Group', 'Canary Release', 'Global Options', 'Mediation', 'NF Rule Profile', 'Pod Overload Control', 'Rate Limiting', 'Reports', 'Routing Options', 'SBI Message Priority' (highlighted), 'SCP Profile', 'Service Groups', 'Topology Source Info', and 'Unassigned Objects'. The main content area is titled 'SBI Message Priority'. It features a 'Query Parameters' input field at the top. Below it is a toolbar with buttons: Get, Add, Edit, Delete, Export, and Clear. There are two tabs: 'Response' and 'Request'. The 'Request' tab is active, showing a JSON object in a text area:

```
{
  "ruleName": "default",
  "data": {
    "nfServiceName": "+",
    "httpMethods": [
      ""
    ],
    "messageType": "+",
    "enableAssignPriority": false,
    "assignPriority": 16,
    "enableOverridePriority": false,
    "overridePriority": 16
  }
}
```

3. To add a new SBI message request, click **Add** and enter the request body information in the **Request** field.
4. Click **Submit**.
The request body message appears in the **Response** field.
5. To edit, export, delete, or clear the data, click the required buttons as described in the following table:

Table 4-44 Buttons and Field Descriptions of SBI Message Priority

Button and Field Names	Description
Query Parameters	Allows to enter parameters to retrieve, add, and update SBI message priority information.
Get	Retrieves the SCP_SBI Message Priority configured in SCP. You can modify this data using the Edit button.
Add	Adds new message requests in the Request field.
Edit	Modifies any existing data by updating mandatory query parameters in the Query Parameters field.
Delete	Removes any message request.
Export	Exports existing records in the JSON format.
Clear	Refreshes the existing screen by removing the data.

4.2.23 Configuring DNS SRV based Alternate Routing

This section provides information to enable different options to use the Alternate Routing based on the DNS SRV Records feature.

You must log in to the CNC Console while performing the procedures described in the subsequent subsections. Using these procedures, you can perform the followings tasks:

- Set the routing mode.
- Configure the default Time To Live (TTL) value.
- Refresh the DNS SRV records in DB.

- Add and fetch DNS SRV records from DB.

4.2.23.1 Configuring Routing Modes

Perform the following procedure to enable alternate routing and set the routing mode to NF Set, DNS SRV, or both when the forward routing fails.

1. From the left navigation pane, navigate to **SCP** and click the **Routing Options** tab.
The Routing Options screen appears with the list of services.
2. To set alternate route modes, in the **Actions** column, click the **Edit** icon for the corresponding service name.

Figure 4-65 Setting Alternate Routing Mode

3. In the **DNS SRV Based Alternate Route** section, set the **Mode** of the selected specific services as required:
 - **NF_SET**: SCP performs alternate routing based on the Model C Indirect 5G SBI Communication format. This is the default mode.
 - **DNS_SRV**: SCP performs alternate routing based on the DNS SRV query.
 - **NF_SET_FOLLOWED_BY_DNSSRV**: SCP performs alternate routing based on NF Set and then with DNS SRV.
4. Click **Save**.

4.2.23.2 Configuring the Default TTL Value

Perform the following procedure to configure the default Time To Live (TTL) value.

1. From the left navigation pane, navigate to **SCP** and click the **Global Options** tab.
2. To configure the **TTL** value, click the **System Config** tab.
3. In the **System Config** section, click the **Edit** icon and set the **TTL** value as required.
By default, this value is set to 900s. The range is between 30s to 86400s.

Figure 4-66 Setting TTL Value

4. Click **Submit**.

4.2.23.3 Configuring Alternate NF Group

Perform the following procedure to add and fetch DNS SRV records from DB.

1. From the left navigation pane, navigate to **SCP** and click the **Alternate NF Group** tab.
2. To configure DNS records using query parameters, click the **Configuration** tab.
3. In the **Configuration** section, click **Add** to add query parameters in the **Request** field.

Figure 4-67 Alternate NF Group Options

4. Click **Submit**.
Query parameters automatically appear in the **Response** field.
5. Click **Get** to view all the records of the query parameter in the **Response** field.
6. To export, delete, or clear the data, click the required buttons as described in the following table:

Table 4-45 Configuration Query Parameters

Button and Field Names	Description
Query Parameters	FQDN of an NF profile. Example: serviceProtoName=_http._tcp.pcf4svc.scpsvc.svc.cluster.local&apiPrefix=USEast
Get	Fetches DNS SRV records from DB.
Add	Adds DNS SRV records to DB by querying the DNS server with the combination of apiPrefix and Service Proto Name. Example: { "apiPrefix": "USEast", "serviceProtoName": "_http._tcp.pcf2svc.scpsvc.svc.cluster.local" }
Delete	Removes a particular record from DB based on Service Proto Name (SPN). Example: serviceProtoName=_http._tcp.udm1svc.scpsvc.svc.cluster.local
Export	Exports DNS SRV records in the JSON format.
Clear	Resets all the fields in the Configuration section.
Response	Displays the required DNS SRV records.
Request	Allows you to add a query parameter when you click the Add button.

7. Click **Submit**.

4.2.23.4 Configuring Refresh DNS SRV Data

Perform the following procedure to refresh the DNS SRV records in DB.

1. From the left navigation pane, navigate to **SCP** and click the **Alternate NF Group** tab.
2. To refresh DB records, click the **Refresh DNS SRV Data** tab in the left navigation pane.
3. Click **Refresh**.
4. In the **Request** field, enter one of the following parameters:

Table 4-46 Request Parameters

Parameter	Description
{ "spnlist": [] }	This parameter refreshes only specific records that are provided within []. For example: { "spnlist": ["_http._tcp.pcf4svc.scpsvc.svc.cluster.local", "_http._tcp.nf1stub.scpsvc.svc"] }
{ "refreshAll": "true" }	This parameter refreshes all the records.

5. Click **Submit**.

4.2.24 Configuring SBI Message Priority and Error Profiles for Egress Congestion Control

Perform the following procedure to configure values for determining the priority of ingress messages and create error profiles.

1. From the left navigation pane, navigate to **SCP** and click the **Routing Options** tab.
2. In the Routing Options section, select a service name from the Service Name column and click the corresponding edit icon.
3. In the NFService Load Based Congestion Control section, in the **SBI Message Priority Discard From** field, enter a value to determine the priority of the ingress message.

Note

- The value of the **SBI Message Priority Discard From** field is compared with the "3gpp-Sbi-Message-Priority" header to categorize an ingress message request as high or low priority. By default, SCP considers 24 as the priority for the ingress message request if the message does not have the "3gpp-Sbi-Message-Priority" header. If the priority of the ingress message request is less than the value of **SBI Message Priority Discard From**, then the message is considered as high priority message.
- SCP discards all the 5G SBI messages whose "3gpp-Sbi-Message-Priority" header value is equal to or greater than this configuration.

Figure 4-68 Configuring SBI Message Priority and Error Profiles for Egress Congestion Control

SBI Message Priority Discard From:	24

▼ Error Profile Configuration

Error Code:	503
Error Cause:	NF_CONGESTION
Error Title:	NF service is overloaded/congested
Retry After:	5
Redirect Url:	
Error Description:	NF service is overloaded/congested

4. To configuring nextHopSCP and SEPP, see [Configuring the nextHopSCP Routing Option](#) section.

- To configure the error profiles, complete the following fields:

Table 4-47 Error Profile Configuration

Field Name	Description
Error Code	Indicates configurable error codes to be sent by SCP to consumer NFs when the producer NF is congested or throttled in case of a low priority message. The default value is 500, which indicates: Service Unavailable.
Error Cause	Indicates the error cause that is specific to the occurrence of the problem. The default value is NF_CONGESTION.
Error Title	Indicates the title of the error type. The default value is NF service is overloaded or congested.
Retry After	Indicates the time during which the service is unavailable. It is applicable when the error code is 503. When the error code is 3xx, where x indicates any numeric value, it specifies the minimum time that the user must wait before issuing the redirected request. The default value is 5. The unit of this value can be HTTP-date or delta-seconds. For example: - Fri, 31 Dec 2001 23:59:59 GMT 120 In the latter example, the delay is 2 minutes.
Redirect Url	Retries the request towards the directed resource referred in the Location header using the same request method. Example: https://udm1.udmsvc.domain/USEast/nudm-sdm/v1/path2
Error Description	Indicates an explanation specific to the occurrence of the problem. The default value is NF service is overloaded or congested.

- Click **Save**.

4.2.25 Configuring the nextHopSCP Routing Option

Perform the following procedure to configure the nextHopSCP routing options.

- From the left navigation pane, navigate to **SCP** and click the **Routing Options** tab.
The Routing Options screen appears with the list of services.
- To edit the Next Hop SCP section, click the **Edit** icon for the corresponding service name.
- Expand the **Next Hop SCP Routing Options** option.

Figure 4-69 NextHopSCP

Next Hop SCP Routing Options

Response Timeout: 4s

Total Transaction Lifetime: 7s

Service Level

Max Routing Attempts: 2

Service Endpoint Level

Max Routing Attempts: 1

NFService Load Based Congestion Control

Alternate Routing Onset Threshold Percent: 80

Alternate Routing Abatement Threshold Percent: 75

Throttle Onset Threshold Percent: 90

Throttle Abatement Threshold Percent: 85

Error Profile Configuration

Reroute On Response Code List:

Exceptions:

Name	Action	Error Code	Error Response
No_Host	Send_Answer	503	Service Unavailable

4. Configure the following fields as required:

Table 4-48 Configuring Next Hop SCP Routing Options

Field Name	Description
Response Timeout	Indicates the allotted time to respond to a message request. When the response timeout expires, SCP performs alternate rerouting to the available alternate NF or pod. If no alternate NFs or pods are available, SCP sends an error message. The default value is 4 seconds.
Total Transaction lifetime	Indicates the total time allowed to forward a request, including initial and all subsequent routing attempts. Note: The total time consumed in processing all retries must not exceed the total transaction lifetime. The default value is 7 seconds.
Max Routing Attempts (Service Level)	Indicates the maximum number of forward routing attempts made at a service level. - Maximum number of times SCP is allowed to forward a request message. If the Max Routing attempts value is set to 1 for both service and pod level, the total transaction lifetime field value is not required. If the Max Routing attempts value is set to greater than 1 for both service and pod level, then the total transaction lifetime field value is considered for rerouting processing. - Rerouting of request messages is always considered as Max Routing attempts value minus one (-1), that is, Rerouting of request messages = (Max Routing attempts value) -1 The default routing attempt is 2.

Table 4-48 (Cont.) Configuring Next Hop SCP Routing Options

Field Name	Description
Max Routing Attempts (Service Endpoint Level)	Indicates the maximum number of forward routing attempts at NF or pod level. - Maximum number of times SCP is allowed to forward a request message. If the Max Routing attempts value is set to 1 for both service and pod level, the total transaction lifetime field value is not required. If the Max Routing attempts value is set to greater than 1 for both service and pod level, then the total transaction lifetime field value is considered for rerouting processing. - Rerouting of request messages is always considered as Max Routing attempts value minus one (-1), that is, Rerouting of request messages = (Max Routing attempts value) -1 The default routing attempt is 1.
Alternate Routing Onset Threshold Percent (NFService Load Based Congestion Control Level)	Indicates the percentage of Alternate Routing Onset Threshold Percent. The default value is 80%.
Alternate Routing Abatement Threshold Percent (NFService Load Based Congestion Control Level)	Indicates the percentage of Alternate Routing Abatement Threshold Percent. The default value is 75%.
Throttle Onset Threshold Percent (NFService Load Based Congestion Control Level)	Indicates the percentage of Throttle Onset Threshold Percent. The default value is 90%.
Throttle Abatement Threshold Percent (NFService Load Based Congestion Control Level)	Indicates the percentage of Throttle Abatement Threshold Percent. The default value is 85%.

- Click **Save**.

4.2.26 Configuring SEPP Inter PLMN Routing

Perform the following procedure to add an NF instance ID and configure Public Land Mobile Network (PLMN) IDs supported by Security Edge Protection Proxy (SEPP).

- From the left navigation pane, navigate to **SCP** and click the **SEPP Info** tab.
- To filter or add an NF instance ID in the SEPP Info section, click the required buttons as described in the following table:

Table 4-49 Filtering and Adding NF Instance IDs

Button, Column, and Field Names	Description
Type to Filter	This field allows you to search for an NF instance ID and displays it in the NF Instance ID column.
Refresh	This button refreshes information in the SEPP Info section.
Add	This button allows you to add a new NF instance ID.
NF Instance ID	This column displays the available NF instance IDs.

Table 4-49 (Cont.) Filtering and Adding NF Instance IDs

Button, Column, and Field Names	Description
Actions	This column allows you to edit an existing NF instance ID using the edit option.

3. To add a new NF instance ID, click **Add**.

Figure 4-70 Adding a New NF Instance ID

4. In the **NF Instance ID** field, enter an instance ID, for example, 8faf1bbc-6e4a-4454-a507-aef01a101a06.
5. Expand the **SEPP Info** option to configure SEPP ports as follows:
 - a. In the **Http** field, enter a value between 0 and 65535.
This field is applicable for the HTTP connection.
 - b. In the **Https** field, enter a value between 0 and 65535.
This field is applicable for the HTTPS connection.

Note

If both the fields are configured, then the HTTPS connection is preferred.

6. To configure PLMN IDs supported by SEPP, in the Remote PLMN List section, click **Add**.

Figure 4-71 Supported PLMN by SEPP

Add Remote PLMN List

MCC:

MCC

MNC:

MNC

Save

Cancel

- a. In the **MCC** field, enter the required mobile country code.
Mobile Country Code (MCC) is a three digit number ranging from 0 to 9.
 - b. In the **MNC** field, enter the required mobile network code.
Mobile Network Code (MNC) can be of two or three digits ranging from 0 to 9.
 - c. Click **Save**.
7. Click **Save**.

4.2.27 Configuring Route Groups

Perform the following procedure to configure Route Groups parameters.

1. From the left navigation pane, click the **SCP** tab, and then click the **Route Groups** tab.
The Route Groups screen appears.

Figure 4-72 Route Groups

Route Groups

Query Parameters:

Query Parameters

Get

Add

Delete

Export

Clear

2. In the Route Groups section, click **Add** to add request body in the **Request** field.
Query parameters are applicable only for DELETE and GET resource types.
3. Click **Submit**.
Response body automatically appears in the **Response** field.

4. Click **Get** to view all the records of the response body in the **Response** field.
5. To export, delete, or clear the data, click the required buttons as described in the following table, and then click **Submit**.

Table 4-50 Route Groups Buttons and Field Descriptions

Button and Field Name	Description
Response	Displays the required Route Groups records.
Request	Allows you to add a query parameter when you click the Add button.
Query Parameters	Processes routeGroupId, routeGroupType, or primaryRoute parameters.
Get	Retrieves configured route groups based on the supplied query parameters as filtering criteria.
Export	Exports Route Groups records in the JSON format.
Delete	Removes the configuration of an existing route group.
Clear	Resets all the fields in this screen.
Add	Configures a new route group or replaces an existing route group by providing a route group configuration in the request body.

4.2.28 Configuring Ingress Rate Limiter Consumer Info

Perform the following procedure to configure consumer information for ingress rate limiting.

1. From the left navigation pane, click the **SCP** tab, and then click the **Ingress Rate Limiter Consumer Info** tab.

The Ingress Rate Limiter Consumer Info screen appears:

Figure 4-73 Ingress Rate Limiter Consumer Info

The screenshot shows the 'Consumer Info Configuration' screen. The left sidebar has a dark theme with the following menu items: < SCP, Alternate NF Group, Canary Release, Consumer Info Configurati..., Global Options, Mediation, NF Rule Profile, Outlier Detection, Pod Overload Control, Rate Limiting, Reports, Route Groups, and Routing Options. The main content area has a light gray header with 'Consumer Info Configuration' and 'Edit' and 'Refresh' buttons. Below the header, there are several rows of configuration fields:

XFCC Certificate Extract Index:	XFCC Certificate Extract Index
	0
XFCC Extract Field:	XFCC Extract Field
	DNS
XFCC Extract Index:	XFCC Extract Index
	0
User Agent Format:	User Agent Format
Primary Header Name:	Primary Header Name
	X-Forwarded-Client-Cert
User Agent Separator:	User Agent Separator
Secondary Header Name:	Secondary Header Name

2. In the Ingress Rate Limiter Consumer Info section, click **Edit**, and then configure the fields as described in the following table:

Table 4-51 Configuring Consumer Information

Field Name	Description
XFCC Certificate Extract Index	The certificate extract index contains certificate information in the XFCC header. This parameter is used to decode "X-Forwarded-Client-Cert" for the FQDN. The default value is 0.
XFCC Extract Field	This is the name of the field that has to be retrieved from the XFCC header. The default value is DNS.
XFCC Extract Index	Index where the field configured in 'extractField' is present in the XFCC header for extraction. The default value is 0.
User Agent Format	The header format to identify the consumer NF according to the ingress rate limiting configurations. It supports the following formats: • Format1: NFTYPE-NFINSTANCEID FQDN or NFTYPE-NFINSTANCEID-FQDN • Format2: NFTYPE-FQDN NFINSTANCEID or NFTYPE-FQDN-NFINSTANCEID • Format3: NFTYPE-NFINSTANCEID • Format4: NFTYPE-FQDN Where, • NFTYPE indicates the type of consumer NF. • NFINSTANCEID indicates the instance ID of the consumer NF. • FQDN indicates the FQDN of the consumer NF. In the aforementioned example, "-" is a separator and NFInstanceId, FQDN, and Hostname are supported IDs.
Primary Header Name	The header name to identify the consumer NF according to the ingress rate limiting configurations. This is a mandatory parameter. This parameter is given priority over the secondaryHeaderName parameter. The default value is "X-Forwarded-Client-Cert". It can use the following values: • "User-Agent" • "X-Forwarded-Client-Cert"
User Agent Separator	This is the separator that the User-Agent header uses to separate NFINSTANCEID and FQDN. The default value is "SPACE". It can use the following values: • "SPACE" • "HYPHEN"
Secondary Header Name	The header name to identify the consumer NF according to the ingress rate limiting configurations. This is an optional parameter. The default value is "". It can use the following values: • "User-Agent" • "X-Forwarded-Client-Cert"

3. Click **Save**.

4.2.29 Configuring SCP Services

Perform the following procedure to view thread watchdog and queue alerts threshold configurations required for SCP microservices.

1. From the left navigation pane, click the SCP tab, and then click the SCP Service Configuration tab.
2. In the left navigation pane, click the SCPC-AlternateResolution tab and do the following:
 - a. In the SCPC-AlternateResolution Service Configuration section, expand the SCP Service Configuration option.
 - b. To view the Thread Watchdog configuration values, expand the Thread Watchdog Configuration option.
 - c. To view the Connectivity Watchdog configuration values, expand the Connectivity Watchdog Configuration option.
3. In the left navigation pane, click the SCPC-Audit tab and do the following:
 - a. In the SCPC-Audit Service Configuration section, expand the SCP Service Configuration option.
 - b. To view the Thread Watchdog configuration values, expand the Thread Watchdog Configuration option.
 - c. To view the Connectivity Watchdog configuration values, expand the Connectivity Watchdog Configuration option.
4. In the left navigation pane, click the SCPC-Configuration tab and do the following:
 - a. In the SCPC-Configuration Service Configuration section, expand the SCP Service Configuration option.
 - b. To view the Thread Watchdog configuration values, expand the Thread Watchdog Configuration option.
 - c. To view the Connectivity Watchdog configuration values, expand the Connectivity Watchdog Configuration option.
5. In the left navigation pane, click the SCPC-Configuration tab and do the following:
 - a. In the SCPC-Configuration Service Configuration section, expand the SCP Service Configuration option.
 - b. To view the Thread Watchdog configuration values, expand the Thread Watchdog Configuration option.
 - c. To view the Connectivity Watchdog configuration values, expand the Connectivity Watchdog Configuration option.
6. In the left navigation pane, click the SCPC-Notification tab and do the following:
 - a. In the SCPC-Notification Service Configuration section, expand the SCP Service Configuration option.
 - b. To view the threshold values of different queues, expand the Queue Utilization Alert Thresholds option.

The SCPC-Notification Service Configuration screen displays the queue values for critical, major, and minor threshold.

Figure 4-74 SCPC-Notification Service Configuration

Queue Name	Critical Queue Threshold	Major Queue Threshold	Minor Queue Threshold	
notificationQueue	86%	76%	66%	View
waitingQueue	85%	75%	65%	View
runQueue	85%	75%	65%	View

The following table describes the significance of each column to determine the values of critical, major, and minor threshold.

Note

The critical, major, and minor threshold values are automatically populated in the CNC Console and cannot be modified. These values are configured using REST APIs as described in "Configuring SCP Services" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Table 4-52 Queue Utilization Alert Thresholds

Column Name	Description
Queue Name	Indicates the name of the queue for which these configurations are applicable.
Critical Queue Threshold	Indicates the percentage (%) value for queue usage threshold upon reaching which a "critical" alert is raised.
Major Queue Threshold	Indicates the percentage (%) value for queue usage threshold upon reaching which a "major" alert is raised.
Minor Queue Threshold	Indicates the percentage (%) value for queue usage threshold upon reaching which a "minor" alert is raised.

7. In the left navigation pane, click the SCPC-Subscription tab and do the following:
 - a. In the SCPC-Subscription Service Configuration section, expand the SCP Service Configuration option.
 - b. To view the Thread Watchdog configuration values, expand the Thread Watchdog Configuration option.
 - c. To view the Connectivity Watchdog configuration values, expand the Connectivity Watchdog Configuration option.
8. In the left navigation pane, click the SCP-NrfProxy tab and do the following:
 - a. In the SCP-NrfProxy Service Configuration section, expand the SCP Service Configuration option.

- b. To view the threshold values of different queues, expand the Queue Utilization Alert Thresholds option.

The SCP-NrfProxy Service Configuration screen displays the queue values for critical, major, and minor threshold.

Figure 4-75 SCP-NrfProxy Service Configuration

Queue Name	Critical Queue Threshold	Major Queue Threshold	Minor Queue Threshold	
scpDownstreamExecutorQueue	85%	75%	65%	View
scpUpstreamExecutorQueue	85%	75%	65%	View

The queue utilization alerts threshold values are automatically populated in the CNC Console as described in [Table 4-52](#).

- c. Expand the Thread Watchdog Configuration option to view the values of thread watchdog configurations:

The following table describes the fields required to configure the Thread Watchdog feature:

Note

Thread watchdog field values are automatically populated in the CNC Console and cannot be modified. These values are configured using REST APIs as described in "Configuring SCP Services" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

Table 4-53 Thread Watchdog Configuration

Field Name	Description
Enable Thread Watchdog	Specifies whether to enable or disable liveness failures. If this parameter is set to <code>true</code> , liveness fails whenever the watchdog reports a stuck or deadlocked thread.
Watchdog Monitoring Interval	Indicates the time interval used by the watchdog to monitor if any threads are stuck.
Watchdog Interval	Indicates the maximum time allowed for threads to be nonresponsive. Watchdog uses this time to determine if a thread is stuck. For example, if this value is set to 10 seconds and a thread does not respond for 10 seconds, then the watchdog marks that thread as stuck or nonresponsive.
Watchdog Failure Count	Indicates the maximum number of times thread watchdog attempts to check if a thread is stuck before marking the thread as hung or stuck. For example, if this value is set to 3, then the watchdog checks if a thread is stuck 3 times continuously before marking it as stuck or hung.

9. In the left navigation pane, click the SCP-Worker tab and do the following:
 - a. In the SCP-Worker Service Configuration section, expand the SCP Service Configuration option.
 - b. To view the threshold values of different queues, expand the Queue Utilization Alert Thresholds option.

The SCP-Worker Service Configuration screen displays the queue values for critical, major, and minor threshold.

Figure 4-76 SCP-Worker Service Configuration

SCP-Worker Service Configuration				Refresh																				
SCP Service Configuration Queue Utilization Alert Thresholds Value: <table> <tr> <th>Queue Name</th><th>Critical Queue Threshold</th><th>Major Queue Threshold</th><th>Minor Queue Threshold</th><th></th></tr> <tr> <td>scpDownstreamExecutorQueue</td><td>85%</td><td>75%</td><td>65%</td><td>View</td></tr> <tr> <td>scpNrfProxyDownstreamExecutorQueue</td><td>85%</td><td>75%</td><td>65%</td><td>View</td></tr> <tr> <td>scpUpstreamExecutorQueue</td><td>85%</td><td>75%</td><td>65%</td><td>View</td></tr> </table>					Queue Name	Critical Queue Threshold	Major Queue Threshold	Minor Queue Threshold		scpDownstreamExecutorQueue	85%	75%	65%	View	scpNrfProxyDownstreamExecutorQueue	85%	75%	65%	View	scpUpstreamExecutorQueue	85%	75%	65%	View
Queue Name	Critical Queue Threshold	Major Queue Threshold	Minor Queue Threshold																					
scpDownstreamExecutorQueue	85%	75%	65%	View																				
scpNrfProxyDownstreamExecutorQueue	85%	75%	65%	View																				
scpUpstreamExecutorQueue	85%	75%	65%	View																				

The queue utilization alerts threshold values are automatically populated in the CNC Console as described in [Table 4-52](#).

- c. Expand the Thread Watchdog Configuration option to view the values of thread watchdog configurations that are automatically populated in the CNC Console as described in [Table 4-53](#).

4.2.30 Configuring Outlier Detection

Perform the following procedure to add, edit, or remove any existing rule name for outlier detection.

1. From the left navigation pane, click the SCP tab, and then click the Outlier Detection tab.
The Outlier Detection screen appears.

Figure 4-77 Outlier Detection

Outlier Detection	
Type to Filter ✕ ↻ Refresh ⊕ Add	
Rule Name ↕	Actions
defaultRule	✎ 🗑
nextHopScpODRule	✎ 🗑

2. If you want to edit or remove any existing rule name from the Rule Name column, click **Edit** or **Delete** icon in the Actions column.
3. To add new rule names, click **Add**.
4. In the Add Outlier Detection section, configure the following fields:
 - **Rule Name:** Enter a unique rule name that is available in the outlier detection table.
 - **Consecutive Errors:** Enter a value between from 5 to 500 to define the number of consecutive errors after which an endpoint is considered unhealthy.
 - **Interval:** Enter a value from 30 seconds to 86400 seconds to set the interval after which stale records are removed from the outlier data cache.
 - **Base Ejection Time:** Enter a value between 1 second and 500 seconds to set the of duration endpoint ejection.
 - **Error List:** Select the required error name from the available error list to be considered for outlier detection.
5. Click **Save**.

4.2.31 Configuring Traffic Feed

Perform the following procedure to configure OCNADD and Trigger Points.

1. From the left navigation pane, click the SCP tab, and then click the Traffic Feed tab.
2. To configure OCNADD fields, in the left navigation pane, click the Data Director tab.
3. In the Traffic Feed Data Director Configuration section, click **Edit** and configure the fields as described in the "Configuring Traffic Feed Data Director" section of the *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
4. To configure Trigger Points, in the left navigation pane, click the Trigger Points tab.
5. In the Trigger Points section, click **Add** and configure the fields as described in the "Configuring Traffic Feed Trigger Point Config" section of the *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
6. Click **Save**.

4.2.32 Congestion Control Configurations

Perform the following steps to configure congestion control parameters:

1. From the left navigation menu, navigate to **SCP** and then select **Congestion Control**.

- Click **Edit** icon from the **Actions** column for the required service that must be modified.

Note

The default rule name cannot be deleted. The new rule name can be added, modified, or deleted.

The **Edit Congestion Control** screen appears.

Figure 4-78 Congestion Control

The screenshot shows the 'Edit Congestion Control' interface. On the left is a sidebar with a menu including 'SCP', 'Alternate NF Group', 'Congestion Control' (highlighted), 'Canary Release', 'Circuit Breaking', 'Consumer Info Configurati...', 'Error Profile', 'Global Options', 'Mediation', 'NF Rule Profile', 'NRF Configuration', 'OAuth Services', 'Outlier Detection', 'Pod Overload Control', and 'Data Limitation'. The main panel has a title 'Edit Congestion Control' and a 'Rule Name' field with the value 'defaultRule'. Below this is a section 'Congestion Control Configuration Data' with five input fields: 'Alternate Routing Onset Threshold Percent' (85), 'Alternate Routing Abatement Threshold Percent' (75), 'Throttle Onset Threshold Percent' (90), 'Throttle Abatement Threshold Percent' (85), and 'Sbi Msg Priority Discard From' (24). At the bottom of this section is a link for 'Error Profile Configuration'. At the very bottom right are 'Save' and 'Cancel' buttons.

- Make the required modifications for the following parameters:
 - Alternate Routing Onset Threshold Percent:** Set the threshold percentage for onset alternate routing.
 - Alternate Routing Abatement Threshold Percent:** Set the threshold percentage for alternate routing abatement.
 - Throttle Onset Threshold Percent:** Set the threshold percentage for onset throttling.
 - Throttle Abatement Threshold Percent:** Set the threshold percentage for throttle abatement.
 - SBI Msg Priority Discard From:** Set the priority for message, after which the messages can be discarded. For example, if the value is 30, then messages with priority 0-29 are high priority, and other low-priority messages will be throttled.

Note

All the parameters are mandatory.

For more information about the configuration parameters, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.

- Click **Save**.
- Click **Cancel** to reset all the fields.
- Click **Delete** from the **Actions** column to delete the record.

4.2.33 Routing Options Configurations

Perform the following procedure to view or edit Routing Options Configuration parameters.

1. From the left navigation menu, navigate to **SCP**, and then click the **Routing Options Configuration** tab.
2. In the Routing Options Configuration section, click **Edit** from the Actions column for the required service that must be modified.
The Edit Routing Options Configuration screen appears.

Figure 4-79 Routing Options Configuration

The screenshot shows the 'Edit Routing Options Configuration' interface. On the left is a dark sidebar with the Oracle logo and a list of navigation items: Pod Overload Control, Rate Limiting, Reports, Route Groups, Routing Options, Routing Options Configura..., SBI Message Priority, SCP Features, SCP Profile, SCP Service Configurati..., SEPP Info, Service Groups, Topology Source Info, Traffic Feed, and Upgrade Status. The 'Routing Options' item is highlighted. The main content area has a header 'Edit Routing Options Configuration' and a sub-header 'Routing Options Config Name: defaultConfig'. Below this are four input fields: 'NF Type' with value 'ALL', 'Service Name' with value 'ALL', 'Message Type List' with value 'notification-message' and a close button, and 'Routing Config SetName' with value 'defaultRoutingConfigSet'. At the bottom right are 'Save' and 'Cancel' buttons. The top right of the console shows 'Cluster1.SCP.Cluster1-scp-instance1', 'About', and 'Sign Out' buttons.

3. In the Routing Options Config Name, configure the following:
 - NF Type: Set the NF type for which routing options are to be configured.
 - Service Name: Set the NF service name for the routing options to be configured.
 - Message Type List: Set the routing options for notification or SBI messages.
 - Routing Options Config SetName: Enter the name of the routing configuration used to retrieve routing options.
4. Enter or modify the required values as described in "Routing Options Configuration" and "Routing Config Set" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
5. Click **Save**.

4.2.34 Routing Config Set

Perform the following procedure to define custom routing options for notification.

1. From the left navigation menu, navigate to **SCP**, and then click the **Routing Config Set** tab.
2. In the Routing Config Set section, click **Edit** from the Actions column for the required service that must be modified.

The Edit Routing Config Set screen appears.

Figure 4-80 Edit Routing Config Set

3. Enter or modify the required values as described in "Routing Config Set" and "Routing Options Configuration" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
4. In the Reroute Condition List, configure the list of error codes for rerouting:
 - Click **Add** to add new status code.
 - Click **Edit** or **Delete** icon to edit or remove any existing error code from the reroute condition list.
5. Click **Save**.
6. Click **Cancel** to reset all the fields.

4.2.35 Circuit Breaking Configurations

Perform the following procedure to configure circuit breaking data.

1. From the left navigation menu, navigate to **SCP**, and then click the **Circuit Breaking** tab.
2. In the Circuit Breaking section, click **Edit** icon from the Actions column for the required service that must be modified.
The Edit Circuit Breaking screen appears.

Figure 4-81 Edit Circuit Breaking

The screenshot shows the 'Edit Circuit Breaking' interface in the Oracle CNC Console. The left sidebar contains a list of navigation items: SCP, Alternate NF Group, Canary Release, Circuit Breaking, Congestion Control, Consumer Info Configurati..., Error Profile, Global Options, Mediation, NF Rule Profile, and NRF Configuration. The main panel is titled 'Edit Circuit Breaking' and displays the 'Rule Name' as 'defaultRule'. Under the 'Circuit Breaking Configuration Data' section, the 'HTTP2 Max Requests' field is set to '1000'. 'Save' and 'Cancel' buttons are located at the bottom right of the configuration area.

3. In the **HTTP2 Max Requests** field, set the maximum number of requests SCP routes to a NF service instance, waiting for a response before stopping further routing requests to it. The maximum allowed value is 1000.
4. Enter or modify the required values as described in "Circuit Breaking Configurations" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*.
5. Click **Save**.
6. Click **Cancel** to reset the field.

4.2.36 Viewing SCP Feature Status

Perform the following procedure to fetch the runtime status of NRF migration task.

1. From the left navigation pane, click the **SCP** tab, and then click the **SCP Feature Status** tab.
2. On the SCP Feature Status page, the following columns display the SCP feature status.

Table 4-54 SCP Feature Status

Column Name	Description
Feature Name	Displays the names of the features, such as nrf_bootstrap_info.
Admin State	Displays the admin state of the feature. Possible statuses are mentioned below: - ENABLED: This indicates that the user has enabled the feature. - DISABLED: This indicates that the user has disabled the feature.

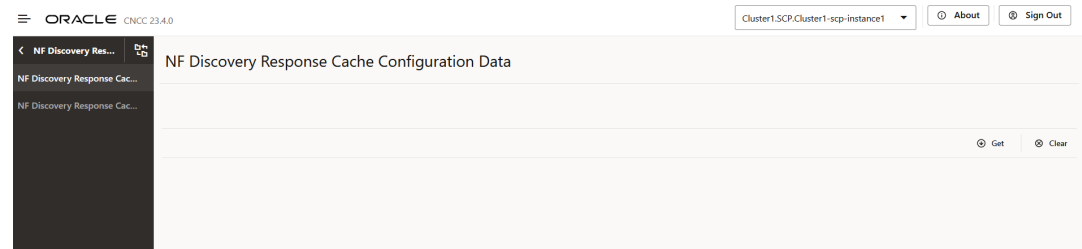
Table 4-54 (Cont.) SCP Feature Status

Column Name	Description
Runtime Status	Displays the current runtime status of the feature. Possible statuses are mentioned below: • DISABLED: This indicates that the feature is disabled. For example, in the <code>nrf_bootstrap_info</code> feature, this state means SCP is running with a statically configured NRF. If the user has changed the 'Admin State' from ENABLED to DISABLED, this state means the feature is successfully disabled. • IN_PROGRESS: This indicates the enabling or disabling of the feature is currently in progress when the 'Admin State' is set to ENABLED or DISABLED. For example, in the <code>nrf_bootstrap_info</code> feature, when the 'Admin State' is ENABLED, the migration from static to DNS SRV based NRF topology is in progress. In the <code>nrf_bootstrap_info</code> feature, when the 'Admin State' is DISABLED, the migration from a DNS SRV based NRF topology to static NRF topology is in progress. • FAILED_RETRY: This indicates that the process of enabling or disabling the feature (as indicated by the 'Admin State') has encountered failure and is undergoing retries. For example, in the <code>nrf_bootstrap_info</code> feature, this state means either of the migration tasks has failed and going through retries. • ENABLED: This indicates the feature is successfully enabled. For example, in the <code>nrf_bootstrap_info</code> feature, this state means the migration from static to a DNS SRV based NRF topology is successful.
Creation Timestamp	Displays the time of creation of the event, for example, <code>creationTimestamp</code> : "2021-05-26T01:17:15.000+00:00"
Update Timestamp	Displays the time of updation of the event, for example, <code>creationTimestamp</code> : "2021-05-26T06:24:15.000+00:00"

4.2.37 Get NF Discovery Response Cache Configuration Data

Perform the following procedure to get the delegated discovery response caching configuration data using Cloud Native Configuration Console (CNC Console).

1. Log in to the CNC Console using your login credentials as described in *Oracle Communications Cloud Native Core, Service Communication Proxy User Guide*.
2. On the CNC Console Home page, in the upper pane, select the required SCP instance from the **Please Select Instance** drop-down list.
3. From the left navigation pane, click the **SCP** tab, and then click the **NF Discovery Response Cache Configuration** tab. The SCP Features section displays the list of two subtabs to be configured using the CNC Console:
 - a. NF Discovery Response Cache Configuration
 - b. NF Discovery Response Cache Configuration Data
4. In the **NF Discovery Response Cache Configuration Data** subtab, click **Get** to retrieve NF discovery cache configuration data based on the query parameters.

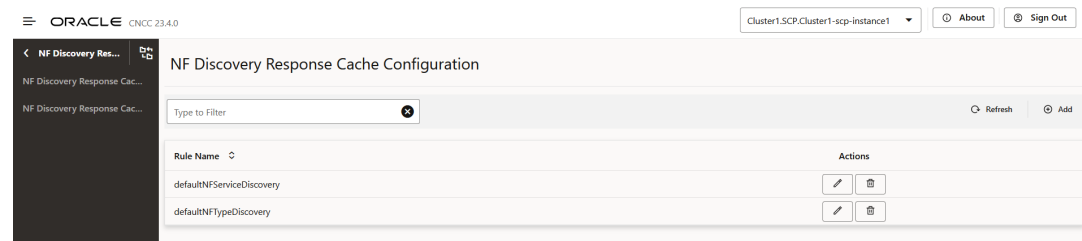
Figure 4-82 NF Discovery Response Cache Configuration Data**Table 4-55 Button and Field Description**

Button and Field Names	Description
Get	Retrieves NF discovery response cache configuration data. You can modify this data using the Edit button.
Clear	Refreshes the existing screen by removing the data.

4.2.38 Configuring NF Delegated Discovery Response Cache

Perform the following procedure to create, edit, and delete the rules for delegated discovery response caching configuration using the Cloud Native Configuration Console (CNC Console).

1. Log in to the CNC Console using your login credentials as described in *Oracle Communications Cloud Native Core, Service Communication Proxy User Guide*.
2. On the CNC Console Home page, in the upper pane, select the required SCP instance from the **Please Select Instance** drop-down list.
3. From the left navigation pane, click the **SCP** tab, and then click the **NF Discovery Response Cache Configuration** tab. The SCP Features section displays the list of two subtabs to be configured using the CNC Console:
 - a. NF Discovery Response Cache Configuration
 - b. NF Discovery Response Cache Configuration Data
4. From the left navigation pane, click the **NF Discovery Response Cache Configuration** subtab.
5. The **NF Discovery Response Cache Configuration** screen displays a list of rules saved in database.

Figure 4-83 NF Discovery Response Cache Configuration

6. Click **Add** to create NF discovery response cache configuration rule. The **Add NF Discovery Response Cache Configuration Rule** screen allows you to configure the fields to create new rule as described in the following table:

Table 4-56 Field Description

Field Names	Description
Rule Name	Indicates the name of the rule. This value should be unique.
Target NF Type	Indicates the NF type of producer for which delegated discovery is being made. The allowed values for this field are: * : This value denotes all the valid NF types. - or - a valid NF type that SCP supports.
Service Names	Indicates the list of service names supported by producer NF for which delegated discovery is being made. The allowed values for this field are: * : This value denotes all the service names for that target NF type. - or - a valid NF service that a NF type supports.
Exclude Discovery Query Parameters	Indicates the list of Exclude Query Info objects. Maximum size of this list is 20.

Note

For more information about valid NF type and services supported by SCP, see [Supported NF Types and Services](#) chapter in the *Oracle Communications Cloud Native Core, Service Communication Proxy User Guide*.

7. In the **Exclude Discovery Query Parameters** section, click **Add** to define the query header to exclude as described in the following table:

Table 4-57 Field Description

Field	Description
Exclude Query Info	Exclude Query Info has one field called query Header. The value of this query header is one of the query parameters that is forwarded along with the discovery request to NRF.
Query Header	Indicates the query parameters that is forwarded along with the discovery request to NRF. The allowed values for this field are: - None: This value denotes that caching is done for all the discovery responses. *: This value denotes that caching will not be done for all the discovery responses. - SubscriberIds: An UE ID-related NF discovery query parameters. - Supported Query Parameters: The query parameters that is forwarded along with the discovery request to NRF. For more information, see Supported Query Parameters chapter.

The following table lists the eight subscriber IDs covered by the keyword "SubscriberIds":

Table 4-58 SubscriberIds

SubscriberIds	Description
supi	If included, this Information Element (IE) shall contain the SUPI of the requester UE to search for an appropriate NF. SUPI may be included if the target NF type is "PCF," "CHF," "AUSF," "UDM," or "UDR."

Table 4-58 (Cont.) SubscriberIds

SubscriberIds	Description
ue-ipv4-address	The IPv4 address of the UE for which a BSF or P-CSCF needs to be discovered.
ip-domain	The IPv4 address domain of the UE for which a BSF needs to be discovered.
ue-ipv6-prefix	The IPv6 prefix of the UE for which a BSF or P-CSCF needs to be discovered.
gpsi	If included, this IE shall contain the GPSI of the requester UE to search for an appropriate NF. GPSI may be included if the target NF type is "CHF," "PCF," "UDM," or "UDR".
external-group-identity	If included, this IE shall contain the external group identifier of the requester UE to search for an appropriate NF. This may be included if the target NF type is "UDM" or "UDR".
routing-indicator	Routing indicator information that allows to route network signaling with SUCI (see 3GPP 23.003 [12]) to an AUSF and UDM instance capable of serving the subscriber. May be included if the target NF type is "AUSF" or "UDM". Pattern: "[0-9]{1,4}\$"
msisdn	If included, this IE shall contain the external group identifier of the requester UE to search for an appropriate NF. This may be included if the target NF type is "UDM" or "UDR".

8. Click **Add** to add the created rules.
9. Click **Remove** to remove the created or existing rules.
10. Click **Save** to save the updates.
11. Click **Cancel** to reset all the fields.
12. Click the **Edit** icon to edit the existing rule.
The Edit NF Discovery Response Cache Configuration Rule screen allows you to configure the fields as described in the following table:

Table 4-59 Field Description

Field	Description
Rule Name	Indicates the name of the rule. This value should be unique.
Target NF Type	Indicates the NF type of producer for which delegated discovery is being made. The default value for this field is *, which denotes that caching will not be done for all the discovery responses.
Service Names	Indicates the list of service names supported by producer NF for which delegated discovery is being made. The default value for this field is *, which denotes that caching will not be done for all the discovery responses.
Exclude Discovery Query Parameters	Indicates the list of ExcludeQueryInfo objects. Maximum size of this list is 20.

Note

- **defaultNFTypeDiscovery** rule is applicable to any Model D request where discovery is happening without service names (in these requests, the 3gpp-sbi-discovery-service-names header is not present) and no other custom rule match is found. For this rule, caching will not happen for the list of query headers mentioned in the ExcludeQueryInfo.
- **defaultNFServiceDiscovery** rule is applicable to a Model D request where the request 3gpp-sbi-discovery-service-names header is also present in the incoming request and no other custom rule match is found. For this rule, caching will not happen for the list of query headers mentioned in the ExcludeQueryInfo.
- You cannot delete the default rules, but the exclude query info for these rules can be edited.

13. Click **Save**.

Note

By default, users can configure up to 100 rules using this API.

5

Metrics

This chapter provides information about newly introduced and updated metrics and its dimensions.

5.1 SCP OCI Metrics Dashboard

To upload the `ocscp_oci_metric_dashboard_24.2.5.zip` file on OCI Logging Analytics Dashboard Service, see *Oracle Communications Cloud Native Core, OCI Adaptor Deployment Guide*. OCI does not support Grafana, OCI uses the Logging Analytics Dashboard Service for visualizing the metrics and logs.

5.2 Dimension Description

The following table provides information on Dimension of Metrics.

Table 5-1 SCP Dimension Descriptions

Dimension	Description
ocscp_nf_type	Includes the producer NF type that is extracted from the URI of the message. The sample values are UDM, PCF, AMF, and so on.
ocscp_nf_service_type	Includes the producer NF Service type that is extracted from the URI of the message. The sample values are nudm-uecm, nudm-sdm, scpc-notification, and so on. Note: Notifications from NRF to scpc-notification will be pegged with the service type as scpc-notification.
ocscp_req_msg_size	Indicates the request message size buckets. The sample values are 1(size < 1k bytes), 2 (1k < size < 2k bytes), 4 (2k < size < 4k), and so on.
ocscp_locality	Indicates the SCP site name or locality name with which SCP is deployed.
ocscp_fqdn	Includes the FQDN of SCP, which is used to deploy SCP. The sample values are Loc6, Loc7, and so on.
plmn_id	Includes the serving PLMN ID of SCP. The sample value is mnc123-mcc254.
ocscp_authority	Includes the authority of SCP. The sample value is FQDN:PORT or IP:PORT. Note: This dimension will get pegged only when it is enabled using <code>values.yaml</code> . <code>scpAuthorityMetricLabelDisabled: false</code>

Table 5-1 (Cont.) SCP Dimension Descriptions

Dimension	Description
ocscp_message_type	Includes the message type received at SCP. It can be of any of the following types: - SERVICE_SPECIFIC_REQUEST - NOTIFICATION_OR_CALLBACK_REQUEST
ocscp_consumer_host	Includes the consumer host details extracted from the user-agent header. SCP should be able to decode that particular format only if the user-agent header is available in certain allowed configurable formats. Note: The ocscp_consumer_host dimension will include the value OCSCP-INITIATED-MSG only in the case of SCP self-generated requests. These requests can be identified based on the header (x-ocscp-initiated-msg=true).
ocscp_consumer_nf_instance_id	Includes the consumer NF Instance ID details extracted from the user-agent header. SCP should be able to decode that particular format only if the user-agent header is available in certain allowed configurable formats.
ocscp_consumer_nf_type	Includes consumer NF type details extracted from the user-agent header. SCP should be able to decode that particular format only if the user-agent header is available in certain allowed configurable formats.
ocscp_consumer_nf_unique_id	Includes the consumer NF's Unique ID details extracted from the user-agent header. SCP should be able to decode that particular format only if the user-agent header is available in certain allowed configurable formats.
ocscp_peer_nf_instance_id	NF Instance ID of the peer to which message (request or response) is routed. For requests, the NF instance ID is obtained from information on a known NF producer or consumer. For responses, the NF Instance ID is obtained from the User-Agent/XFCC header. SCP should be able to decode that particular format only if the user-agent header is available in certain allowed configurable formats.
ocscp_peer_nf_service_type	NF service type of peer to which message (request or response) is routed. For requests, the NF service type is obtained from information on a known NF producer or consumer. For responses, service type is obtained from learned information about NF extracted from the User-Agent/XFCC header. SCP should be able to decode that particular format only if the user-agent header is available in certain allowed configurable formats.
action	Action taken ("error_sent" or "rerouted") on request throttling because of OCI enforcement.
scope	The OCI scope received in the OCI header.
auditMode	The mode of Audit in which SCP performs with NRF using NRF discovery or management services. The possible values are "nnrf-disc" Or "nnrf-mgmt"

Table 5-1 (Cont.) SCP Dimension Descriptions

Dimension	Description
auditType	Type of audit triggered by the audit microservice. It can be an on-demand audit or a regular audit, depending on what triggered the audit. Example: A change in TSI from local to NRF will trigger an on-demand audit. The possible values are "OnDemand" and "Scheduled".
bodyModified	This dimension is used to know if the response received from mediation has any changes in the body that is sent in the request towards mediation.
bucketKey	The key used to maintain the value of the message rate in egress or ingress rate limit functionality.
cause	The reason for the event's occurrence. The value can vary from one metric to another.
cca_consumer_nflInstanceId	The NF instance ID of the consumer sending the CCA header.
certificate_type	The type of certificate used, such as primary server, secondary server, etc.
certificate_type	Indicates the type of certificates that are about to expire. The following are the valid applicable certificate types: 5G_SBI_SERVER_PRIMARY - Applicable for 5G SBI Server Primary certificate, used for Ingress traffic. 5G_SBI_SERVER_SECONDARY - Applicable for 5G SBI Server Secondary certificate, used for Ingress traffic. 5G_SBI_CLIENT_PRIMARY - Applicable for 5G SBI Client Primary certificate, used for Egress traffic. 5G_SBI_CLIENT_SECONDARY - Applicable for 5G SBI Client Secondary certificate, used for Egress traffic. 5G_SBI_TRAFFIC_FEED_PRIMARY - Applicable for 5G SBI Traffic Feed primary certificate, used for Data Director (kafka). 5G_SBI_TRAFFIC_FEED_SECONDARY - Applicable for 5G SBI Traffic Feed secondary certificate, used for Data Director (kafka).
coherenceCluster	The name of the coherence cluster used in the SCP deployment.
configurationName	The name of the configuration that has changed.
cpu_level	The level for CPU utilization, whether it is under minor, major, or critical utilization levels based on the configurations.
dbOperation	The DB operation that happened on a DB table. Example: find, findAll, save, saveAll, get, count, delete, deleteAll.
destination	Indicates the destination SVC, like scpc-notification, scp-nrfproxy, and so on.
destination (TLS)	Contains the destination address details involved in HTTPS connection.
detail	This dimension contains details about the issue that has happened.
duplicateTargetFqdn	When a duplicate target fqdn is detected across different NRF SRV fqdns.
error-reason	Reason for ingress or egress connection failures.
errorType	The error reason that caused notification processing to fail.

Table 5-1 (Cont.) SCP Dimension Descriptions

Dimension	Description
eventName	Showing the migration state of the NRF DNS SRV feature. Possible values are: - FEATURE_ENABLED - START_NRF_RULE_PROCESSING
eventType	It shows the status of the egress requests on the Jetty client or the event type received in notification requests from NRF. This is used in different metrics, thus refer to the respective metric for interpretation.
failureReason	Reasons for the failure of the NRF DNS SRV feature migration.
failureReason (TLS)	Contains HTTPS connection failure reason (Handshake failure reason)
headerModified	Indicates whether any header is modified by the mediation service in response.
host_type	Indicates the host type used for sending the request. For example, "IP" or "FQDN".
http_method	Indicates the HTTP method of the request getting treatment for overload. For example, PUT, PATCH, POST, and so on.
interScpHost	Contains the host of the SCP to which the request is being sent.
is_from_local_cache	Defines whether the NRF discovery response is retrieved from local cache (true) or NRF (false)
is_inter_plmn	Defines whether NRF Delegated Discovery made is for inter plmn (true) or for intra plmn (false)
locality	The locality of the foreign NF.
lockEvent	The notification cache lock event (released or acquired)
msName	The name of the scp microservice.
nfInstanceId	The producer NF instance ID.
nfProfileProcessingStatus	The status of the notification profile processing.
ocscp_nf_service_type	The NF service type.
nf_status	The status of NF, for example, REGISTERED.
nfType	The NF type, for example, UDM.
notificationSource	Indicates the source of notification received at the Notification microservice. It can be NRF or some SCP microservice.
rule_name	Denotes the rule that matches one of the discovery headers in the excludeDiscoveryQuery List, and hence caching is skipped.
ocscp_producer_nf_type	Includes the producer NF type.
nrfHost	NRF FQDN or IP address configured for audit.
nrfRegionOrSetId	NRF region or SetId configured for audit
nrfSrvFqdn	NRF Fqdn configured in DNS SRV
ocscp_alternate_nf_end_point	Selected alternate producer NF's end point.
ocscp_alternate_producer_fqdn	Selected alternate producer NF FQDN.
ocscp_alternate_producer_nf_instance_id	Selected alternate producer NF's instance Id.
ocscp_alternate_producer_service_instance_id	Selected alternate producer NF's service instance Id.
ocscp_alternate_route_type	Alternate routing mechanism selected.

Table 5-1 (Cont.) SCP Dimension Descriptions

Dimension	Description
ocscp_attempted_producers	List of attempted producers before sending back an error response to the consumer.
ocscp_sub	Requester NF instance ID in the oauth2 token request.
ocscp_aud	Target NF instance ID in the oauth2 token request.
ocscp_cause	Traffic Feed Metrics: Provides information on the cause of traffic feed failures (for example, unreachable, noleader, incorrect data director configuration). Routing Metrics: Provides information on the cause of routing failures. (for example, UnknowHostException, Circuit Breaking) Rate limit Metrics: Provides information on the cause for the rate limit not being applied (for example, configuration not found).
ocscp_configured_egress_rate	Configured rate of egress rate limit for a NF.
ocscp_configured_ingress_rate	Configured rate of ingress rate limit for a NF.
ocscp_egress_rl_bucketkey	Key name for egress rate limit bucket.
ocscp_error_condition	Error Condition resulting in routing attempt failure.
ocscp_error_group	Error group for error condition resulting in Routing Attempt Failure (for example, Destination webclient connection failure).
ocscp_ingress_rl_bucketkey	Key name for the ingress rate limit bucket.
ocscp_ingress_rl_headerformat	The User Agent header format used by SCP to extract consumer host information to apply an ingress rate limit.
ocscp_ingress_rl_headername	Headers used to apply the ingress rate limit.
ocscp_med_upstream_service_time	Time taken for mediation processing in milliseconds.
ocscp_mediation_filtername	Name of the filter applied to send a request towards mediation.
ocscp_mediation_triggerpoint	Trigger point used to send requests toward mediation.
ocscp_nexthop_scp_fqdn	FQDN of the selected SCP for interscp requests.
ocscp_nf_end_point	- The endpoint details of producer NF or SEPP for interplmn scenario. - The endpoint details of producer NF or SCP for interscp scenario. The endpoint can be a fqdn or IP address.
ocscp_nf_instance_id	Producer NF instance ID for health check metrics.
ocscp_nf_service_type	Service name of the producer NF in delegated discovery metrics.
ocscp_nf_setid	The NF setid of selected producer NF. If nfsetid is present in the discovery headers or routing binding headers, retrieve it from there; otherwise, it should be extracted from the nfprofile of producer.
ocscp_nrf_service_type	Service name of the NRF NF in delegated discovery metrics.
ocscp_peer_scp_fqdn	The FQDN of peer SCP.
ocscp_peer_scp_instance_id	The NF instance Id of peer SCP.
ocscp_primary_producer_fqdn	The FQDN of the selected primary producer NF, in the case of alternate routing.
ocscp_processing_time	The bucket list of SCP requests or response processing times.

Table 5-1 (Cont.) SCP Dimension Descriptions

Dimension	Description
ocscp_producer_host	The FQDN of selected producer NF.
ocscp_producer_nf_fqdn	The FQDN of producer NF for InterSCP Metrics.
ocscp_producer_nf_instance_id	- The NF Instance Id of producer NF or SEPP for interplmn scenario. - The NF Instance Id of producer NF or SCP for interscp scenario.
ocscp_producer_nf_type	The NF Type of producer NF selected by SCP.(For example, UDM, PCF and so on)
ocscp_producer_service_instance_id	- The service instance Id of producer NF or SEPP for interplmn scenario. - The service instance Id of producer NF or SCP for interscp scenario.
ocscp_res_msg_size	Message size of the response sent by SCP to the consumer. Message size buckets. for example, 1 (size < 1k bytes), 2 (1k < size < 2k bytes), 4 (2k < size < 4k), etc.
ocscp_response_code	Response code sent by SCP to the consumer (for example, a success response such as 200 or 201, an error response such as 503, 400, and so on.)
ocscp_response_code_class	Response code class of response code sent by SCP to the consumer. (For example, response code class 2xx for success responses, 5xx, 4xx, or 3xx for error responses.)
ocscp_scope	Scope NF Load received is applicable to nfInstance or serviceInstance.
ocscp_scp_end_point	Producer SCP endpoint details for interSCP metrics. Endpoint details can contain a FQDN or IP address.
ocscp_scp_instance_id	Producer SCP instance ID for interSCP metrics.
ocscp_sepp_end_point	Selected SEPP endpoint details for InterPLMN metrics. Endpoint details can contain a FQDN or IP address.
ocscp_sepp_fqdn	Selected SEPP FQDN details for InterPLMN metrics.
ocscp_sepp_instance_id	Selected SEPP instance ID for InterPLMN metrics.
ocscp_server_header	Server header received in case of failure received from side car.
ocscp_threshold_level	The CPU threshold level.
ocscp_upstream_service_time	The time taken to receive a response from producer NF.
peer_info	The NF instance ID of the peer for which the load is reported.
pod_name	The pod name for which metric is pegged.
PrimaryEndPoint	Primary end point selected by SCP for routing before alternate routing.
processingTimeInMs	Total time taken for notification processing in milliseconds.
queueCapacity	Notification Metrics: Maximum Notification queue size (runQueue:40, notificationQueue:5000, waitingQueue: 5000) Nrf Proxy Metrics: Maximum size is 10000. Load Manager Metrics: Maximum size is 10000. Cache Metrics: The maximum size of the queue is 30000 for scpCacheExecutorQueue and 10000 for others. Worker Metrics: Metrics: The maximum size is 10000.

Table 5-1 (Cont.) SCP Dimension Descriptions

Dimension	Description
queueName	Notification Queue Name (runQueue, notificationQueue, waitingQueue) NRF Proxy Metrics: Name of the nrf-proxy queue (scpDownstreamExecutorQueue, scpUpstreamExecutorQueue) Load Manager Metrics: Name of the queue (sutLciCohEvExecutorQueue, sutLciAppEvExecutorQueue, peerLciCohEvExecutorQueue, peerLciAppEvExecutorQueue, loadCohEvExecutorQueue, loadAppEvExecutorQueue) Cache Metrics: Name of the queue (scpCacheExecutorQueue, accessTokenAppEvExecutorQueue, accessTokenCohEvExecutorQueue) Worker Metrics: Name of the queue (scpDownstreamExecutorQueue, scpUpstreamExecutorQueue, scpNrfProxyDownstreamExecutorQueue, scpClientExecutorQueue, scpUpdateSSLCertsQueue, scpMediationDownstreamExecutorQueue, scpReconfigureEgressRLExecutorQueue)
reqType	The request type for the oauth token request (for example, ondemand or proactive).
requester_nfType	The requester NF Type received the request for oauth token.
requestType	The HTTP method of request received on configuration (for example, GET, PUT, or DELETE).
responseCode	The HTTP status code in the response from SCP.
responseCodeClass	The HTTP status code class in response to SCP (For example, the response code is 2xx for success responses, 5xx, 4xx, or 3xx for error responses.) The possible values are "OnDemand" and "Scheduled".
sbi_message_priority	The value of the sbi_message_priotiy header in the request received.
scheme	The scheme received in the rx request or the scheme of the selected producer for the tx request.
scp_local_coh_cluster_name	The SCP local coherence cluster name.
scp_nf_instance_id	The NF instance ID of SCP.
scp_remote_coh_cluster_fqdn	The remote SCP coherence cluster FQDN.
scp_remote_coh_cluster_name	The remote SCP coherence cluster name.
scp_remote_coh_cluster_nf_instance_id	The remote SCP coherence cluster NF instance ID.
scp_remote_coh_cluster_port	The remote SCP coherence cluster port.
scp_service	The SCP microservice name for which metric is pegged.
service_proto_name	The service proto-name used for DNS SRV resolution.
severity	The severity of the condition.
source	The Microservice from which the request was originated.
source (TLS)	Contains the source address details involved in HTTPS connection.
status	The health check status code.
status_code	The response status code.

Table 5-1 (Cont.) SCP Dimension Descriptions

Dimension	Description
subscriber_info_present	If any of the following discovery headers are present in the request: 3gpp-sbi-discovery-supi, 3gpp-sbi-discovery-gpsi, 3gpp-sbi-discovery-external-group-identity, 3gpp-sbi-discovery-routing-indicator, 3gpp-sbi-discovery-msisd, or 3gpp-sbi-discovery-imsi.
subscriber_info_present_in_discovery_headers	If any of the following discovery headers are present in the request: 3gpp-sbi-discovery-supi, 3gpp-sbi-discovery-gpsi, 3gpp-sbi-discovery-external-group-identity, 3gpp-sbi-discovery-routing-indicator, 3gpp-sbi-discovery-msisd, or 3gpp-sbi-discovery-ims.
successRspType	The success response code received for the health check.
tableName	The table name indicates which operation is performed. For example, SERVICE_ENTRIES.
target_nfType	The NF type in oAuth access token request.
targetFqdn	The NF FQDN in an oAuth access token request.
taskId	The NRF DNS SRV migration task ID.
taskName	The NRF DNS SRV migration task name.
taskState	The NRF DNS SRV migration task state.
timeInMs	The time taken in Notification queues.
title	The error reason for Health check failure.
triggerpoint	The mediation triggerpoint
tls_version	Indicates the TLS version used to establish the HTTPS connection. Note: In some failure cases, the TLS version will not be available and will be displayed as "NA"
valid_NRF_response_to_fail	Whether SCP converts 200 OK (NRF) to 400 error due to an empty profile list.
validation_type	The validation type for CCA.
kafka_partition_id	Partition ID of the Kafka partition, which is selected for traffic feed messages. For the partition selection strategy "RoundRobin," partitions are not known to the SCP application, thus the partition ID is used as "NA."
kafka_broker_id	The broker ID of the Kafka partition leader is selected for the traffic feed message. For the partition selection strategy "RoundRobin," broker IDs are not known to the SCP application, thus the broker ID is used as "NA."
ocscp_object_pool_name	The name of the object pool in the case of multiple object pool implementations.
ocscp_worker_client_name	The name of the async client for identification of pool objects paraphrase.
ocscp_object_trans_state	The transaction status of pool objects during the transaction timeout for pool objects.
clash_type	The cause of clash for pool objects.
ocscp_attempted_producers	This is a list of producer FQDNs or IPs attempted for a particular SBI message.

Note

- The values for these dimensions, such as `ocscp_consumer_host`, `ocscp_consumer_nf_instance_id`, and `ocscp_consumer_nf_type` are available based on the availability of the User-Agent header in incoming message requests. SCP should be able to decode that particular format only if the User-Agent header is available in certain allowed configurable formats.
- In the notification message type, certain dimension values such as `ocscp_nf_service_type`, `ocscp_nf_setid`, and `ocscp_nf_type` may not qualify for pegging. In the scenario where the service name cannot be derived from the PATH URI, pegging will not take place, and if these dimensions are identified through headers, pegging will occur.

5.2.1 SCP Metrics

The following table provides information for metrics for Service Communication Proxy (SCP).

Caution

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

Note

- The `ocscp_nf_service_type` dimension for notifications can be `n<nfType>-unknown` or `NA-unknown`, where `<nfType>` is any NF name.
- Please note that Kubernetes-based dimensions, such as pod name or namespace, require updates based on the Prometheus deployment.
- All the metrics with HTTP in their names are pegged for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.

Table 5-2 `ocscp_metric_http_rx_req_total`

Field	Details
Description	The total number of incoming HTTP requests at the scp-worker service. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter

Table 5-2 (Cont.) ocscp_metric_http_rx_req_total

Field	Details
Dimension	Non OCI Dimensions - ocscp_nf_type (for example, UDM, PCF, AMF, and so on) - ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on) - ocscp_req_msg_size (The request message size buckets, for example, 1 (size < 1k bytes), 2 (1k < size < 2k bytes), 4 (2k < size < 4k), and so on.) - ocscp_locality (SCP site name, for example, Loc6, Loc7, and so on) - ocscp_fqdn (contains fqdn of scp) - plmn_id(mnc123-mcc254) - ocscp_authority - ocscp_message_type - ocscp_consumer_host - ocscp_consumer_nf_instance_id - ocscp_consumer_nf_type - ocscp_consumer_nf_unique_id - These dimensions are available based on the availability of the User-Agent header in incoming message requests. SCP should be able to decode that particular format only if the User-Agent header is available in certain allowed configurable formats. - The ocscp_consumer_host dimension for the metric can have an "OCSCP-INITIATED-MSG" value if the request is an SCP self-generated request. OCI Dimensions - NFSERVICETYPE - REQUESTMSGSIZE - PLMNID - SCPAUTHORITY - MSGTYPE - consumerInfo - CONSUMERUNIQUEID
Example	- ocscp_metric_http_rx_req_total{ocscp_nf_type="UDM"} - ocscp_metric_http_rx_req_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_http_rx_req_total{ocscp_req_msg_size="2"} - ocscp_metric_http_rx_req_total{ocscp_locality="Loc7"} - ocscp_metric_http_rx_req_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_http_rx_req_total{ocscp_authority="ocscp-scp-worker.scpsvc.svc.cluster.local:8000"} - ocscp_message_type="SERVICE_SPECIFIC_REQUEST"} - ocscp_metric_http_rx_req_total{ocscp_consumer_host="smf1.scpsvc.svc.cluster.local"} - ocscp_metric_http_rx_req_total{ocscp_consumer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_metric_http_rx_req_total{ocscp_consumer_nf_type="SMF"} - ocscp_metric_http_rx_req_total{ocscp_consumer_nf_unique_id="SE1SMSF1"}

Table 5-3 ocscp_metric_http_tx_req_total

Field	Details
Description	The total number of HTTP requests forwarded by SCP to the upstream cluster. Each SCP-worker pod will peg this metric. Note: The ocscp_authority dimension will get pegged only when it is enabled using the value.yaml file. scpAuthorityMetricLabelDisabled: false Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimension	Non OCI Dimensions • ocscp_nf_type (for example, UDM, PCF, AMF, and so on) • ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on) • ocscp_nf_end_point (Default value = NA) • ocscp_locality (SCP site name for example, Loc6, Loc7, and so on) • ocscp_producer_nf_instance_id (for example, 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, and so on. Default value = NA) • ocscp_producer_service_instance_id (for example, fe137ab7-740a-46ee-aa5c-951806d77b02, and so on. Default value = NA) • ocscp_producer_host (default value = NA, Contains fqdn of producer) • ocscp_nf_setid (set1.udmset.5gc.mnc012.mcc345, default value = "UNKNOWN") • plmn_id(mnc123-mcc254) • ocscp_fqdn(contains fqdn of scp) • ocscp_authority • ocscp_message_type • ocscp_consumer_host • ocscp_consumer_nf_instance_id • ocscp_consumer_nf_type • ocscp_consumer_nf_unique_id • These dimensions are available based on the availability of the User-Agent header in incoming message requests. SCP should be able to decode that particular format only if the User-Agent header is available in certain allowed configurable formats. • The ocscp_consumer_host dimension for the metric can have an "OCSCP-INITIATED-MSG" value if the request is an SCP self-generated request. OCI Dimensions • NFSERVICETYPE • producerHost • Nfid • NFSETID • PLMNID • SCPAUTHORITY • MSGTYPE • consumerInfo • CONSUMERUNIQUEID

Table 5-3 (Cont.) ocscp_metric_http_tx_req_total

Field	Details
Example	- ocscp_metric_http_tx_req_total{ocscp_nf_type="UDM"} - ocscp_metric_http_tx_req_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_http_tx_req_total{ocscp_nf_end_point="10.96.166.65:80"} - ocscp_metric_http_tx_req_total{ocscp_locality="Loc7"} - ocscp_metric_http_tx_req_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_http_tx_req_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_http_tx_req_total{ocscp_producer_host="udm1svc.default.svc.cluster.local"} - ocscp_metric_http_tx_req_total{ocscp_nf_setid="set1.udmset.5gc.mnc012.mcc345"} - ocscp_metric_http_tx_req_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_http_tx_req_total{ocscp_authority="ocscp-scp-worker.scpsvc.svc.cluster.local:8000"} - ocscp_message_type="NOTIFICATION_OR_CALLBACK_REQUEST"} - ocscp_metric_http_tx_req_total{ocscp_consumer_host="smf1.scpsvc.svc.cluster.local"} - ocscp_metric_http_tx_req_total{ocscp_consumer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_metric_http_tx_req_total{ocscp_consumer_nf_type="SMF"} - ocscp_metric_http_tx_req_total{ocscp_consumer_nf_unique_id="SE1SMSF1"}

Table 5-4 ocscp_metric_http_rx_res_total

Field	Details
Description	Total number of HTTP responses received by SCP with specific HTTP response codes (e.g., 201, 503, etc.). Each SCP-worker pod will peg this metric. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter

Table 5-4 (Cont.) ocscp_metric_http_rx_res_total

Field	Details
Dimension	Non OCI Dimensions - ocscp_response_code (for example, 201, 503, 404, and so on.) - ocscp_nf_type (for example, UDM, PCF, AMF, and so on.) - ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on.) - ocscp_nf_end_point (Default value = 0.0:00) - ocscp_locality (SCP site name for example, Loc6, Loc7, and so on.) - ocscp_producer_nf_instance_id (for example, 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, and so on. Default value = NA) - ocscp_producer_service_instance_id (for example, fe137ab7-740a-46ee-aa5c-951806d77b02, and so on. Default value = NA) - ocscp_res_msg_size (The response message size buckets. For example, 1 (size < 1k bytes), 2 (1k < size < 2k bytes), 4 (2k < size < 4k), and so on.) - ocscp_response_code_class (e.g., 2, 5, and so on.) - ocscp_producer_host (default value = NA, Contains fqdn of producer) - ocscp_nf_setid(e.g, set1.udmset.5gc.mnc012.mcc345, default value = "UNKNOWN") - plmn_id(mnc123-mcc254) - ocscp_fqdn(contains fqdn of scp) - ocscp_authority Note: These dimensions are available based on the availability of the User-Agent header in incoming message requests. SCP should be able to decode that particular format only if the User-Agent header is available in certain allowed configurable formats. OCI Dimensions - NFSERVICETYPE - producerHost - Nfid - RESPONSECODE - RESPONSEMSGSIZE - NFSETID - PLMNID - SCPAUTHORITY
Example	- ocscp_metric_http_rx_res_total{ocscp_response_code="201"} - ocscp_metric_http_rx_res_total{ocscp_nf_type="UDM"} - ocscp_metric_http_rx_res_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_http_rx_res_total{ocscp_nf_end_point="10.96.166.65:80"} - ocscp_metric_http_rx_res_total{ocscp_locality="Loc7"} - ocscp_metric_http_rx_res_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_http_rx_res_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_http_rx_res_total{ocscp_res_msg_size="2"} - ocscp_metric_http_rx_res_total{ocscp_response_code_class="2"} - ocscp_metric_http_rx_res_total{ocscp_producer_host="udm1svc.default.svc.cluster.local"} - ocscp_metric_http_rx_res_total{ocscp_nf_setid="set1.udmset.5gc.mnc012.mcc345"} - ocscp_metric_http_rx_res_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_http_rx_res_total{ocscp_authority="ocscp-scp-worker.scpsvc.svc.cluster.local:8000"}

Table 5-5 ocscp_metric_http_tx_res_total

Field	Details
Description	Total number of HTTP responses forwarded by SCP back to the consumer or request originator with specific HTTP response codes (For example, 201, 503, and so on) Each SCP-worker pod will peg this metric when sending the response back to the consumer. Note: The ocscp_authority dimension will get pegged only when it is enabled using the value.yaml file. scpAuthorityMetricLabelDisabled: false Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimension	Non OCI Dimensions - ocscp_response_code (for example, 201, 503, 404, and so on.) - ocscp_nf_type (for example, UDM, PCF, AMF, and so on.) - ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on.) - ocscp_locality (SCP site name for example, Loc6, Loc7, and so on.) - ocscp_response_code_class (for example, 2, 5, and so on.) - ocscp_fqdn(contains fqdn of scp) - ocscp_authority - plmn_id(mnc123-mcc254) - ocscp_consumer_host - ocscp_consumer_nf_instance_id - ocscp_consumer_nf_type - ocscp_consumer_nf_unique_id Note: The ocscp_consumer_host dimension for the metric can have an "OCSCP-INITIATED-MSG" value if the request is an SCP self-generated request. OCI Dimensions - NFSERVICETYPE - RESPONSECODE - PLMNID - SCPAUTHORITY - consumerInfo - CONSUMERUNIQUEID
Example	- ocscp_metric_http_tx_res_total{ocspf_response_code="201"} - ocscp_metric_http_tx_res_total{ocspf_nf_type="UDM"} - ocscp_metric_http_tx_res_total{ocspf_nf_service_type="nudm-uecm"} - ocscp_metric_http_tx_res_total{ocscp_locality="Loc7"} - ocscp_metric_http_tx_res_total{ocscp_response_code_class="2"} - ocscp_metric_http_tx_res_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_http_tx_res_total{ocscp_authority="ocscp-scp-worker.scpsvc.svc.cluster.local:8000"} - ocscp_metric_http_tx_res_total{ocscp_consumer_host="smf1.scpsvc.svc.cluster.local"} - ocscp_metric_http_tx_res_total{ocscp_consumer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_metric_http_tx_res_total{ocscp_consumer_nf_type="SMF"} - ocscp_metric_http_tx_res_total{ocscp_consumer_nf_unique_id="SE1SMSF1"}

Table 5-6 ocscp_metric_attempts_to_forward_route_total

Field	Details
Description	Total number of requests that matched the catch-all-route routing rule.
Type	Counter
Dimension	- ocscp_nf_type (for example, UDM, PCF, AMF, and so on.) - ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on.) - ocscp_nf_end_point (Default value = 0.0:00) - ocscp_locality (SCP site name for example, Loc6, Loc7, and so on.) - ocscp_producer_nf_instance_id (for example, 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc. Default value = NA) - ocscp_producer_service_instance_id (for example, fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_fqdn (contains fqdn of scp)
Example	- ocscp_metric_attempts_to_forward_route_total{ocscp_nf_type="UDM"} - ocscp_metric_attempts_to_forward_route_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_attempts_to_forward_route_total{ocscp_nf_end_point="10.96.166.65:80"} - ocscp_metric_attempts_to_forward_route_total{ocscp_locality="Loc7"} - ocscp_metric_attempts_to_forward_route_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_attempts_to_forward_route_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_attempts_to_forward_route_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-7 ocscp_metric_routing_attempt_fail_total

Field	Details
Description	Total number of request that SCP failed to route due to any reason.
Type	Counter
Dimension	- ocscp_nf_type (for example, UDM, PCF, AMF, and so on.) - ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on.) - ocscp_nf_end_point (default value = NA) - ocscp_locality (SCP site name, for example, Loc6, Loc7, and so on.) - ocscp_producer_nf_instance_id (for example, 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc., default value = NA) - ocscp_producer_service_instance_id (for example, fe137ab7-740a-46ee-aa5c-951806d77b01, etc., default value = NA) - ocscp_error_group - ocscp_producer_host (default value = NA, contains the fqdn of the producer) - ocscp_fqdn (contains the fqdn of scp)

Table 5-7 (Cont.) ocscp_metric_routing_attempt_fail_total

Field	Details
Example	- ocscp_metric_routing_attempt_fail_total{ocscp_nf_type="UDM"} - ocscp_metric_routing_attempt_fail_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_routing_attempt_fail_total{ocscp_nf_end_point="10.96.166.65:80"} - ocscp_metric_routing_attempt_fail_total{ocscp_locality="Loc7"} - ocscp_metric_routing_attempt_fail_total{ocscp_producer_nf_instance_id="6faf1b-bc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_routing_attempt_fail_total{ocscp_producer_service_instance_id="f-e137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_routing_attempt_fail_total{ocscp_error_group="ocscp.http.routing_failure.connect_failure"} - ocscp_metric_routing_attempt_fail_total(ocscp_producer_host="udm1svc.default.svc.cluster.local") - ocscp_metric_routing_attempt_fail_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-8 ocscp_metric_request_processing_time_total

Field	Details
Description	This metric captures the processing time by SCP for ingress requests into the time buckets, for example, 1ms, 2ms, 4ms, 8ms and so on.
Type	Histogram
Dimension	- ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on.) - ocscp_processing_time (for example, 1ms, 2ms, 4ms, 8ms, 16ms, 32ms, 64ms, 128ms, 256ms, 500ms, 1000ms, 2000ms, 3000ms, 5000ms, 10000ms, 15000ms, >15000ms) - ocscp_fqdn (contains fqdn of scp)
Example	- ocscp_metric_request_processing_time_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_request_processing_time_total{ocscp_processing_time="1ms"} - ocscp_metric_request_processing_time_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-9 ocscp_metric_response_processing_time_total

Field	Details
Description	This metric captures the processing time by SCP for ingress responses into the time buckets (for example, 1ms, 2ms, 4ms, 8ms and so on).
Type	Histogram
Dimension	- ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on.) - ocscp_processing_time (for example, 1ms, 2ms, 4ms, 8ms, 16ms, 32ms, 64ms, 128ms, 256ms, 500ms, 1000ms, 2000ms, 3000ms, 5000ms, 10000ms, 15000ms, >15000ms.) - ocscp_fqdn (contains fqdn of scp)
Example	- ocscp_metric_response_processing_time_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_response_processing_time_total{ocscp_processing_time="1ms"} - ocscp_metric_response_processing_time_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-10 ocscp_metric_request_per_try_timeout_total

Field	Details
Description	This metric captures the number of incoming request whose per try timeout expired.
Type	Counter
Dimension	- ocscp_nf_type (for example, UDM, PCF, AMF, and so on.) - ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on.) - ocscp_nf_end_point (Default value = NA) - ocscp_locality (SCP site name for example, Loc6, Loc7, etc.) - ocscp_producer_nf_instance_id (for example, 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc. Default value = NA) - ocscp_producer_service_instance_id (for example, fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_producer_host (default value = NA, Contains fqdn of producer) - ocscp_nf_setid(e.g, set1.udmset.5gc.mnc012.mcc345, default value = "UNKNOWN") - ocscp_fqdn (contains fqdn of scp)
Example	- ocscp_metric_request_per_try_timeout_total{ocscp_nf_type="UDM"} - ocscp_metric_request_per_try_timeout_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_request_per_try_timeout_total{ocscp_nf_end_point="10.96.166.65:80"} - ocscp_metric_request_per_try_timeout_total{ocscp_locality="Loc7"} - ocscp_metric_request_per_try_timeout_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_request_per_try_timeout_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_request_per_try_timeout_total{ocscp_producer_host="udm1svc.default.svc.cluster.local"} - ocscp_metric_request_per_try_timeout_total{ocscp_nf_setid="set1.udmset.5gc.mnc012.mcc345"} - ocscp_metric_request_per_try_timeout_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-11 ocscp_metric_transaction_timeout_total

Field	Details
Description	This metric captures the total number of request whose transaction timed out.
Type	Counter
Dimension	- ocscp_nf_type (for example, UDM, PCF, AMF, and so on.) - ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on.) - ocscp_nf_end_point (Default value = NA) - ocscp_locality (SCP site name for example, Loc6, Loc7, and so on.) - ocscp_producer_nf_instance_id (for example, 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, and so on. Default value = NA) - ocscp_producer_service_instance_id (for example, fe137ab7-740a-46ee-aa5c-951806d77b02, and so on. Default value = NA) - ocscp_producer_host (default value = NA, Contains fqdn of producer) - ocscp_fqdn (contains fqdn of scp)

Table 5-11 (Cont.) ocscp_metric_transaction_timeout_total

Field	Details
Example	- ocscp_metric_transaction_timeout_total{ocscp_nf_type="UDM"} - ocscp_metric_transaction_timeout_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_transaction_timeout_total{ocscp_nf_end_point="10.96.166.65:80"} - ocscp_metric_transaction_timeout_total{ocscp_locality="Loc7"} - ocscp_metric_transaction_timeout_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_transaction_timeout_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_transaction_timeout_total(ocscp_producer_host="udm1svc.default.svc.cluster.local") - ocscp_metric_transaction_timeout_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-12 ocscp_metric_max_routing_attempts_exhausted_total

Field	Details
Description	This metric captures the total number of requests whose maximum routing attempts expired during alternate routing.
Type	Counter
Dimension	- ocscp_nf_type (for example, UDM, PCF, AMF, and so on.) - ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on.) - ocscp_locality (SCP site name for example, Loc6, Loc7, and so on.) - ocscp_producer_nf_instance_id (for example, 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, and so on. Default value = NA) - ocscp_producer_service_instance_id (for example, fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_fqdn (contains fqdn of scp)
Example	- ocscp_metric_max_routing_attempts_exhausted_total{ocscp_nf_type="UDM"} - ocscp_metric_max_routing_attempts_exhausted_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_max_routing_attempts_exhausted_total{ocscp_locality="Loc7"} - ocscp_metric_max_routing_attempts_exhausted_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_max_routing_attempts_exhausted_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_max_routing_attempts_exhausted_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-13 ocscp_notifications_rejected_topologysource_local_total

Field	Details
Description	Number of NF notification messages rejected because "learning from NRF" was configured as "not allowed". Metrics Pegging Condition: This metrics will be pegged whenever notification is received for NF configured as LOCAL.
Type	Counter

Table 5-13 (Cont.) ocscp_notifications_rejected_topologysource_local_total

Field	Details
Dimension	- nfType (for example: UDM, AMF) - ocscp_fqdn
Example	- scp_local_topology_source_total{nfType = "UDM"} - scp_local_topology_source_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-14 ocscp_topologysource_toggle_nrf_to_local_total

Field	Details
Description	Number of times topology source changed from "NRF" to "Local" for given NF.
Type	Counter
Dimension	- nfType (for example: UDM, AMF) - ocscp_fqdn
Example	- scp_topologysource_toggle_nrf_to_local_total{nfType = "UDM"} - scp_topologysource_toggle_nrf_to_local_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-15 ocscp_topologysource_toggle_local_to_nrf_total

Field	Details
Description	Number of times topology source changed from "Local" to "NRF" for given NF.
Type	Counter
Dimension	- nfType (for example: UDM, AMF) - ocscp_fqdn
Example	- scp_topologysource_toggle_local_to_nrf_total{nfType = "UDM"} - scp_topologysource_toggle_local_to_nrf_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-16 ocscp_db_operation_failure_total

Field	Details
Description	Number of times audit failed due to database failure.
Type	Counter
Dimension	- ocscp_fqdn - dbOperation (eg: find, findAll, save, saveAll, get, count, delete, deleteAll) - tableName (eg: VIRTUAL_SERVICES, SERVICES_ENTRIES, NF_PROFILES.)

Table 5-16 (Cont.) ocscp_db_operation_failure_total

Field	Details
Example	- ocscp_db_operation_failure_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_db_operation_failure_total{dbOperation="findAll"} - ocscp_db_operation_failure_total{tableName="SERVICE_ENTRIES"}

Table 5-17 ocscp_notification_nf_profile_processing_failure_total

Field	Details
Description	Number of times notification process failure.
Type	Counter
Dimension	- nfType (eg., AMF) - nfInstanceId - ocscp_fqdn
Example	ocscp_notification_nf_profile_processing_failure_total{nfType="UDM",nfInstanceId="6faf3abc-6e4a-4454-a507-a14ef8e1bc4b",ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-18 ocscp_patch_subscription_nf_success_total

Field	Details
Description	Number of successful patch operation of subscription based on nfType and nrfRegionOrSetId.
Type	Counter
Dimension	- nfType (eg: UDM, AMF) - nrfRegionOrSetId (eg:Reg1,Reg2) - ocscp_fqdn
Example	- ocscp_patch_subscription_nf_success_total{nfType="A USF" - ocscp_patch_subscription_nf_success_total{nrfRegionOrSetId="Reg1"} - ocscp_patch_subscription_nf_success_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-19 ocscp_patch_subscription_nf_failure_total

Field	Details
Description	Number of failure patch operation of subscription based on nfType and nrfRegionOrSetId.
Type	Counter
Dimension	- nfType (eg: UDM, AMF) - nrfRegionOrSetId (eg:Reg1,Reg2) - ocscp_fqdn

Table 5-19 (Cont.) ocscp_patch_subscription_nf_failure_total

Field	Details
Example	- ocscp_patch_subscription_nf_failure_total{nfType="AUS F"} - ocscp_patch_subscription_nf_failure_total{nrfRegionOrSetId="Reg1"} - ocscp_patch_subscription_nf_failure_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-20 ocscp_unsubscription_nf_success_total

Field	Details
Description	Number of successful operations of unsubscription based on nfType and nrfRegionOrSetId.
Type	Counter
Dimension	- nfType (for example: UDM, AMF) - nrfRegionOrSetId(eg:Reg1,Reg2) - ocscp_fqdn
Example	- ocscp_unsubscription_nf_success_total{nfType="AUSF"} - ocscp_unsubscription_nf_success_total{nrfRegionOrSetId="Reg1"} - ocscp_unsubscription_nf_success_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-21 ocscp_unsubscription_nf_failure_total

Field	Details
Description	Number of failure operations of unsubscription based on nfType and nrfRegionOrSetId.
Type	Counter
Dimension	- nfType (for example: UDM, AMF) - nrfRegionOrSetId(eg:Reg1,Reg2) - ocscp_fqdn
Example	- ocscp_unsubscription_nf_failure_total{nfType="AUSF"} - ocscp_unsubscription_nf_failure_total{nrfRegionOrSetId="Reg1"} - ocscp_unsubscription_nf_failure_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-22 ocscp_subscription_nf_success_total

Field	Details
Description	Number of successful operations of subscription based on nfType and nrfRegionOrSetId.
Type	Counter
Dimension	- nfType (for example: UDM, AMF) - nrfRegionOrSetId(eg:Reg1,Reg2) - ocscp_fqdn

Table 5-22 (Cont.) ocscp_subscription_nf_success_total

Field	Details
Example	- ocscp_subscription_nf_success_total{nftype="AUSF"} - ocscp_subscription_nf_success_total{nrfRegionOrSetId="Reg1"} - ocscp_subscription_nf_success_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-23 ocscp_subscription_nf_failure_total

Field	Details
Description	Number of failure operations of subscription based on nfType and nrfRegionOrSetId.
Type	Counter
Dimension	- nfType (for example: UDM, AMF) - nrfRegionOrSetId (eg: Reg1, set1.nrfset.5gc.mnc014.mcc310) - ocscp_fqdn
Example	- ocscp_subscription_nf_failure_total{nftype="AUSF"} - ocscp_subscription_nf_failure_total{nrfRegionOrSetId="Reg1"} - ocscp_subscription_nf_failure_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-24 ocscp_audit_2xx_empty_nf_array

Field	Details
Description	The state of metrics is 1 when SCP audit receives empty NF array response for nfType. Then, state of metrics is changes to 0 when audit receives non empty NF array.
Type	Gauge
Dimension	- ocscp_fqdn - nfType (eg: UDM, AMF) - auditMethod - nrfRegionOrSetId(eg:Reg1,Reg2)
Example	ocscp_audit_2xx_empty_nf_array_rx_total{nrfRegionOrSetId="Reg2"} ocscp_audit_2xx_empty_nf_array_rx_total{nftype="UDM"} ocscp_audit_2xx_empty_nf_array_rx_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-25 ocscp_audit_error_response

Field	Details
Description	The state of metrics is 1 when SCP audit receives error response for nfType. Then, state of metrics is changed to 0 when audit receives non error response for nfType.
Type	Gauge

Table 5-25 (Cont.) ocscp_audit_error_response

Field	Details
Dimension	- ocscp_fqdn - nfType (eg: UDM, AMF) - auditMethod - nrfRegionOrSetId(eg:Reg1,Reg2)
Example	ocscp_audit_error_response{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-26 ocscp_subscription_nrf_registration_success_total

Field	Details
Description	This metrics pegs the number of times registration successful for a particular nrfRegionOrSetId.
Type	Counter
Dimension	nrfRegionOrSetId For example: (Reg11,Reg1,set1.nrfset.5gc.mnc014.mcc310) ocscp_fqdn
Example	- ocscp_subscription_nrf_registration_success_total{nrfRegionOrSetId="Reg11"} - ocscp_subscription_nrf_registration_success_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-27 ocscp_subscription_nrf_registration_failure_total

Field	Details
Description	This metrics pegs the number of times registration failed for a particular nrfRegionOrSetId.
Type	Counter
Dimension	nrfRegionOrSetId For example: (Reg11,Reg1,set1.nrfset.5gc.mnc014.mcc310) ocscp_fqdn
Example	- ocscp_subscription_nrf_registration_failure_total{nrfRegionOrSetId="Reg11"} - ocscp_subscription_nrf_heartbeat_failures_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-28 ocscp_subscription_nrf_heartbeat_success_total

Field	Details
Description	This metrics pegs the number of times heartbeat successful for a particular nrfRegionOrSetId.
Type	Counter
Dimension	nrfRegionOrSetId For example: (Reg11,Reg1,set1.nrfset.5gc.mnc014.mcc310) ocscp_fqdn

Table 5-28 (Cont.) ocscp_subscription_nrf_heartbeat_success_total

Field	Details
Example	- ocscp_subscription_nrf_heartbeat_success_total{nrfRegionOrSetId="Reg11"} - ocscp_subscription_nrf_heartbeat_success_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-29 ocscp_subscription_nrf_heartbeat_failures_total

Field	Details
Description	This metrics pegs the number of times heartbeat failed for a particular nrfRegionOrSetId.
Type	Counter
Dimension	nrfRegionOrSetId For example: (Reg11,Reg1,set1.nrfset.5gc.mnc014.mcc310) ocscp_fqdn
Example	- ocscp_subscription_nrf_heartbeat_failures_total{nrfRegionOrSetId="Reg11"} - ocscp_subscription_nrf_heartbeat_failures_total{scpFqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-30 ocscp_metric_upstream_service_time_total

Field	Details
Description	This metric captures the time taken by upstream host is responding the request in time buckets, for example, 1ms, 2ms, 4ms, 8ms, and so on.
Type	Histogram
Dimension	- ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on.) - ocscp_nf_end_point (Default value = NA) - ocscp_producer_nf_instance_id (for example, 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, and so on. Default value = NA) - ocscp_producer_service_instance_id (for example, fe137ab7-740a-46ee-aa5c-951806d77b02, and so on. Default value = NA) - ocscp_upstream_service_time (for example, 1ms, 2ms, 4ms, 8ms, etc.16ms, 32ms, 64ms, 128ms, 256ms, 500ms, 1000ms, 2000ms, 3000ms, 5000ms, 10000ms, 15000ms, >15000ms.) - ocscp_producer_host (default value = NA, Contains fqdn of producer) - ocscp_fqdn

Table 5-30 (Cont.) ocscp_metric_upstream_service_time_total

Field	Details
Example	- ocscp_metric_upstream_service_time{ocscp_nf_type="UDM"} - ocscp_metric_upstream_service_time{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_upstream_service_time{ocscp_nf_end_point="10.96.166.65:80"} - ocscp_metric_upstream_service_time{ocscp_locality="LOC7"} - ocscp_metric_upstream_service_time{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_upstream_service_time{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_upstream_service_time{ocscp_upstream_service_time="1ms"} - ocscp_metric_upstream_service_time{ocscp_producer_host="udm1svc.default.svc.cluster.local"} - ocscp_metric_upstream_service_time_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_upstream_service_time_total{ocscp_auth_arity="ocscp-scp-worker.scpsvc.svc.cluster.local:8000"}

Table 5-31 ocscp_metric_scp_generated_response_total

Field	Details
Description	Captures the error response generated by SCP.
Type	Counter
Dimension	- ocscp_nf_type (for example: UDM, PCF, AMF, etc.) - ocscp_nf_service_type (for example: nudm-uecm, nudm-sdm, etc.) - ocscp_response_code (for example: 503, 404, 408, etc.) - ocscp_error_group (for example: ocscp.http.config_error.cluster_not_found, ocscp.http.routing_failure.connect_failure, etc) - ocscp_fqdn - ocscp_nf_end_point(nrf2svc.scpsvc.svc.cluster.local:8080) - ocscp_producer_nf_instance_id(3faf3bbc-6e4a-2828-a507-a14ef8e1bc6a) - ocscp_producer_host(nrf2svc.scpsvc.svc.cluster.local)

Table 5-31 (Cont.) ocscp_metric_scp_generated_response_total

Field	Details
Example	- ocscp_metric_scp_generated_response_total{ocscp_nf_type="UDM",ocscp_nf_service_type="nudm-uecm", - ocscp_metric_scp_generated_response_total{ocscp_response_code="503",ocscp_error_group="ocscp.http.routing_failure.connect_failure",ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local",ocscp_authority="ocscp-scp-worker.scpsvc.svc.cluster.local:8000"} - ocscp_metric_scp_generated_response_total{ocscp_error_group="Destination webclient Connection Failure",ocscp_nf_end_point="nfr2l2svc.scpsvc.svc.cluster.local:8080",ocscp_producer_nf_instance_id="3faf2bb-c-6e4a-2828-a507-a14ef8e1bc7b",ocscp_producer_host="nfr2l2svc.scpsvc.svc.cluster.local",ocscp_response_code="503"}

Table 5-32 ocscp_notification_failure_type_total

Field	Details
Description	Number of Notification failed to process.
Type	Counter
Dimension	- errorType (for example: Invalid Supi Range, Invalid Identity Range, Notification Failure) - ocscp_fqdn
Example	- ocscp_notification_failure_type_total{errorType="Invalid Identity Range"} Note: errorType like Invalid Supi Range, Invalid Identity Range, Notification Failure - ocscp_notification_failure_type_total{ocscp_fqdn="ocscpworker.scpsvc.svc.cluster.local"}

Table 5-33 ocscp_nrf_notifications_requests_nf_total

Field	Details
Description	Number of Notifications Requests from NRF specific to NFs per NF Type.
Type	Counter
Dimension	- nftype (for example: UDM, AMF) - ocscp_fqdn
Example	- ocscp_nrf_notifications_requests_nf_total{nftype="UDM"} - ocscp_nrf_notifications_requests_nf_total{ocscp_fqdn="ocscp-scpworker.scpsvc.svc.cluster.local"}

Table 5-34 ocscp_notification_nf_profile_processing_success_total

Field	Details
Description	Number of notifications processed successfully.
Type	Counter

Table 5-34 (Cont.) ocscp_notification_nf_profile_processing_success_total

Field	Details
Dimension	- nftype (for example: UDM, AMF) - ocscp_fqdn
Example	- ocscp_notification_nf_profile_processing_success_total{nftype="UDM"} - ocscp_notification_nf_profile_processing_success_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-35 ocscp_config_tsi_change_ondemand_audit_invoke_success_total

Field	Details
Description	Number of successful On-Demand Audit invocations when TSI is changed.
Type	Counter
Dimension	nrfRegionOrSetId For example: (Reg11,Reg1,Reg2) ocscp_fqdn
Example	- ocscp_config_tsi_change_ondemand_audit_invoke_success_total{nrfRegionOrSetId="Reg11"} - ocscp_config_tsi_change_ondemand_audit_invoke_success_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-36 ocscp_config_tsi_change_ondemand_audit_invoke_failure_total

Field	Details
Description	Number of failed On-Demand Audit invocations when TSI is changed.
Type	Counter
Dimension	nrfRegionOrSetId For example: (Reg11,Reg1,Reg2) ocscp_fqdn
Example	- ocscp_config_tsi_change_ondemand_audit_invoke_failure_total{nrfRegionOrSetId="Reg11"} - ocscp_config_tsi_change_ondemand_audit_invoke_failure_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-37 ocscp_duplicate_audit_profile_total

Field	Details
Description	Number of duplicate profiles added by audit due to notification slowness
Type	Counter

Table 5-37 (Cont.) ocscp_duplicate_audit_profile_total

Field	Details
Dimension	nfType (for example: UDM, AMF) nfInstanceId (for example: 4947a69a-f61b-4bc1-b9da-47c9c5d14b64) ocscp_fqdn
Example	- ocscp_duplicate_audit_profile_total{nfType="UDM"} - ocscp_duplicate_audit_profile_total{nfInstanceId="4947a69a-f61b-4bc1-b9da-47c9c5d14b64"} - ocscp_duplicate_audit_profile_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-38 ocscp_notification_foreign_nf_locality_absent

Field	Details
Description	Sets to 1 if a foreign NF is registered from an unknown locality that is locality is not present in the foreign NF
Type	Gauge
Dimension	locality (for example: UNKNOWN_LOCALITY) nfInstanceId (for example: 1faf1bbc-6e4a-4454-a507-a14ef8e1bb6a) nfType (for example: UDM, AMF) nrregionorsetid (for example: Reg1,Reg2) ocscp_fqdn
Example	- ocscp_notification_foreign_nf_locality_absent{locality="UNKNOWN_LOCALITY"} - ocscp_notification_foreign_nf_locality_absent{nfInstanceId="4947a69a-f61b-4bc1-b9da-47c9c5d14b64"} - ocscp_notification_foreign_nf_locality_absent{nfType="UDM"} - ocscp_notification_foreign_nf_locality_absent{nrregionorsetid="Reg2"} - ocscp_notification_foreign_nf_locality_absent{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-39 ocscp_notification_foreign_nf_locality_unserved

Field	Details
Description	Sets to 1 if a foreign NF is registered and its locality is not served by any SCP
Type	Gauge
Dimension	locality (for example: Loc80) nfInstanceId (for example: 1faf1bbc-6e4a-4454-a507-a14ef8e1bb6a) nfType (for example: UDM, AMF) nrregionorsetid (for example: Reg1, Reg2) ocscp_fqdn

Table 5-39 (Cont.) ocscp_notification_foreign_nf_locality_unserved

Field	Details
Example	- ocscp_notification_foreign_nf_locality_unserved{locality="Loc80"} - ocscp_notification_foreign_nf_locality_unserved{nflnstanceId="4947a69a-f61b-4bc1-b9da-47c9c5d14b64"} - ocscp_notification_foreign_nf_locality_unserved{nftype="UDM"} - ocscp_notification_foreign_nf_locality_unserved{nrfregionorsetid="Reg2"} - ocscp_notification_foreign_nf_locality_unserved{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-40 ocscp_notification_duplicate_foreign_location

Field	Details
Description	Sets to 1 if Locality of foreign NF(mate/serving locality in case of foreign SCP) is same as one of locality present in the local SCP.
Type	Gauge
Dimension	locality (for example: Loc83) nflnstanceId (for example: 1faf1bbc-6e4a-4454-a507-a14ef8e1bb6a) nftype (for example: UDM, AMF) nrfregionorsetid (for example: Reg1, Reg2) ocscp_fqdn
Example	- ocscp_notification_duplicate_foreign_location{locality="Loc83"} - ocscp_notification_duplicate_foreign_location{nflnstanceId="4947a69a-f61b-4bc1-b9da-47c9c5d14b64"} - ocscp_notification_duplicate_foreign_location{nftype="UDM"} - ocscp_notification_duplicate_foreign_location{nrfregionorsetid="Reg2"} - ocscp_notification_duplicate_foreign_location{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-41 ocscp_interscp_downstream_rq_total

Field	Details
Description	Keeps track of ingress message received from another SCP
Type	Counter
Dimension	ocscp_fqdn (for example: site2-ocscp-scp-worker.scpsvc.svc.cluster.local") ocscp_nf_service_type (for example:"nudm-uecm") ocscp_nf_type (for example:"UDM") ocscp_fqdn

Table 5-41 (Cont.) ocscp_interscp_downstream_rq_total

Field	Details
Example	- ocscp_interscp_downstream_rq_total{ocscp_fqdn="site 2-ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_interscp_downstream_rq_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_interscp_downstream_rq_total{ocscp_nf_type="UDM"}

Table 5-42 ocscp_metric_rule_engine_processing_time_total

Field	Details
Description	Captures the processing time by SCP for rule engine
Type	Counter
Dimension	- ocscp_nf_type (for example, UDM, PCF, AMF, etc.) - ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, etc.) - ocscp_locality (SCP site name for example, Loc6, Loc7, etc.) - ocscp_processing_time (for example, 1ms, 2ms, 4ms, 8ms, etc) - ocscp_fqdn (contains fqdn of scp)
Example	ocscp_metric_rule_engine_processing_time_total{ocscp_locality="Loc7"} ocscp_metric_rule_engine_processing_time_total{ocscp_nf_service_type=" nudm-uecm"} ocscp_metric_rule_engine_processing_time_total{ ocscp_nf_type="UDM"} ocscp_metric_rule_engine_processing_time_total{ocscp_processing_time="29"}

Table 5-43 ocscp_metric_upstream_response_timeout_total

Field	Details
Description	Contains information of timeout for every attempt made by producer to get a response.
Type	Counter
Dimension	- ocscp_nf_type (for example, UDM, PCF, AMF, etc.) - ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, etc.) - ocscp_locality (SCP site name for example, Loc6, Loc7, etc.) - ocscp_producer_host (for example, udm1svc.default.svc.cluster.local) - ocscp_response_code (for example, 503 SERVICE_UNAVAILABLE, 504 GATEWAY_TIMEOUT) - ocscp_nf_setid(e.g, set1.udmset.5gc.mnc012.mcc345, default value = "UNKNOWN") - ocscp_fqdn

Table 5-43 (Cont.) `ocscp_metric_upstream_response_timeout_total`

Field	Details
Example	<code>ocscp_metric_upstream_response_timeout_total{ ocscp_nf_type="UDM"}</code> <code>ocscp_metric_upstream_response_timeout_total{ocscp_location="Loc7"}</code> <code>ocscp_metric_upstream_response_timeout_total{ocscp_nservice_type=" nudm-uecm"}</code> <code>ocscp_metric_upstream_response_timeout_total{ocscp_processing_time="29"}</code> <code>ocscp_metric_upstream_response_timeout_total{ocscp_response_code="504 GATEWAY_TIMEOUT"}</code> <code>ocscp_metric_upstream_response_timeout_total{ocscp_producer_host="udm1svc.default.svc.cluster.local"}</code> <code>ocscp_metric_upstream_response_timeout_total{ocscp_nfsetid="set1.udmset.5gc.mnc012.mcc345"}</code>

Table 5-44 `ocscp_metric_http_streaminfo_create_total`

Field	Details
Description	Captures information of application, microservice, fqdn and vendor whenever StreamInfo Object is initialized. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimension	<code>ocscp_fqdn</code>
Example	<code>ocscp_metric_http_streaminfo_create_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}</code>

Table 5-45 `ocscp_metric_http_streaminfo_delete_total`

Field	Details
Description	Captures information of application, microservice, fqdn and vendor just before StreamInfo Object is destroyed before the garbage collection. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimension	<code>ocscp_fqdn</code>
Example	<code>ocscp_metric_http_streaminfo_delete_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}</code>

Table 5-46 ocscp_upstream_rq_pending_overflow_total

Field	Details
Description	Captures total number of requests for which circuit has been broken due to http2MaxLimit reached for outstanding requests.
Type	Counter
Dimension	- ocscp_nf_type(for example, UDM, PCF, AMF, etc.) - ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, etc.) - ocscp_locality(SCP site name for example, Loc6, Loc7, etc.) - ocscp_producer_host(for example, udm1svc.default.svc.cluster.local) - ocscp_producer_nf_instance_id (for example, 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc. Default value = NA) - ocscp_producer_service_instance_id (for example, fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_fqdn (contains fqdn of scp)
Example	<pre>ocscp_upstream_rq_pending_overflow_total{ ocscp_nf_type="UDM"}</pre> <pre>ocscp_upstream_rq_pending_overflow_total{ocscp_nf_service_type=" nudm-uecm"}</pre> <pre>ocscp_upstream_rq_pending_overflow_total{ocscp_locality="Loc7"}</pre> <pre>ocscp_upstream_rq_pending_overflow_total{ocscp_producer_host="udm1svc.default.svc.cluster.local"}</pre> <pre>ocscp_upstream_rq_pending_overflow_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"}</pre> <pre>ocscp_upstream_rq_pending_overflow_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"}</pre> <pre>ocscp_upstream_rq_pending_overflow_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}</pre>

Table 5-47 ocscp_circuit_breaking_applied

Field	Details
Description	Indicates the circuit breaking state. If outstanding request limit has been reached then gauge will show 1 otherwise 0.
Type	Gauge

Table 5-47 (Cont.) ocscp_circuit_breaking_applied

Field	Details
Dimension	- ocscp_nf_type(for example, UDM, PCF, AMF, etc.) - ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, etc.) - ocscp_locality (SCP site name for example, Loc6, Loc7, etc.) - ocscp_producer_host (for example, udm1svc.default.svc.cluster.local) - ocscp_producer_nf_instance_id (for example, 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc. Default value = NA) - ocscp_producer_service_instance_id (for example, fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_fqdn (contains fqdn of scp)
Example	<pre>ocscp_circuit_breaking_applied{ ocscp_nf_type="UDM"} ocscp_circuit_breaking_applied{ocscp_nf_service_type=" nudm-uecm"} ocscp_circuit_breaking_applied{ocscp_locality="Loc7"} ocscp_circuit_breaking_applied{ocscp_producer_host="udm 1svc.default.svc.cluster.local"} ocscp_circuit_breaking_applied{ocscp_producer_nf_instanc e_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} ocscp_circuit_breaking_applied{ocscp_fqdn="ocscp-scp- worker.scpsvc.svc.cluster.local"}</pre>

Table 5-48 ocscp_configuration_tx_res_total

Field	Details
Description	Indicates the number of times the configuration microservice receives request from users.
Type	Counter
Dimension	- configurationName(RoutingOptions, CanaryRelease, MessagePriority, NfServiceGroup, MediationConfiguration, SystemOptions, LogLevels, TracingOptions, IngressRateLimiter, EgressRateLimiter, WorkerPodOverloadCtrlPolicy, WorkerPodOverloadActionPolicy, WorkerPodOverloadDiscardPolicy) - requestType(GET, PUT, DELETE) - responseCode - ocscp_fqdn
Example	<pre>ocscp_configuration_tx_res_total{configurationName="Syste moptions"} ocscp_configuration_tx_res_total{requestType="PUT"} ocscp_configuration_tx_res_total{responseCode="200"} ocscp_configuration_tx_res_total{ocscp_fqdn="ocscp-scp- worker.scpsvc.svc.cluster.local"}</pre>

Table 5-49 ocscp_rx_configuration_total

Field	Details
Description	Indicates the number of times the configuration microservice responds to users either with a successful or failure response.
Type	Counter
Dimension	- configurationName (RoutingOptions, CanaryRelease, MessagePriority, NfServiceGroup, MediationConfiguration, SystemOptions, LogLevels, TracingOptions, IngressRateLimiter, EgressRateLimiter, WorkerPodOverloadCtrlPolicy, WorkerPodOverloadActionPolicy, WorkerPodOverloadDiscardPolicy) - requestType(GET, PUT, DELETE) - ocscp_fqdn
Example	<pre>ocscp_rx_configuration_total{configurationName="Systemoptions"} ocscp_rx_configuration_total{requestType="PUT"} ocscp_rx_configuration_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}</pre>

Table 5-50 ocscp_notification_all_routing_rules_deleted_total

Field	Details
Description	Indicates no of times all routing rules are deleted.
Type	Counter
Dimension	- nfType (for example: UDM, AMF) - ocscp_fqdn
Example	- ocscp_notification_all_routing_rules_deleted_total{nfType="UDM"} - ocscp_notification_all_routing_rules_deleted_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-51 ocscp_no_nf_instance

Field	Details
Description	Indicates the state of number of instances for NFType.
Type	Gauge
Dimension	- nftype (for example: UDM, AMF) - ocscp_fqdn
Example	- ocscp_no_nf_instance{nftype="UDM"} - ocscp_no_nf_instance{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-52 ocscp_metric_egress_rate_limiting_not_applied_req_total

Field	Details
Description	Number of requests routed without egress rate limiting treatment.
Type	Counter
Dimension	- ocscp_nf_type - ocscp_nf_service_type - ocscp_producer_nf_instance_id - ocscp_producer_host - ocscp_cause - ocscp_egress_rl_bucketkey - ocscp_fqdn
Example	- ocscp_metric_egress_rate_limiting_not_applied_req_total{ocscp_nf_type="UDM"} - ocscp_metric_egress_rate_limiting_not_applied_req_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_egress_rate_limiting_not_applied_req_total{ocscp_producer_host="udm1svc.default.svc.cluster.local"} - ocscp_metric_egress_rate_limiting_not_applied_req_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_egress_rate_limiting_not_applied_req_total{ocscp_cause="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_egress_rate_limiting_not_applied_req_total{ocscp_egress_rl_bucketkey="egress_6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_egress_rate_limiting_not_applied_req_total{ocscp_fqdn="ocscp-scp-worker.scpvc.svc.cluster.local"}

Table 5-53 ocscp_metric_egress_rate_limiting_alternate_routed_req_total

Field	Details
Description	Number of requests alternate routed because of egress rate limiting.
Type	Counter
Dimension	- ocscp_nf_type - ocscp_nf_service_type - ocscp_producer_nf_instance_id - ocscp_producer_host - ocscp_egress_rl_bucketkey - ocscp_fqdn - ocscp_consumer_host - ocscp_consumer_nf_instance_id - ocscp_consumer_nf_type

Table 5-53 (Cont.) ocscp_metric_egress_rate_limiting_alternate_routed_req_total

Field	Details
Example	- ocscp_metric_egress_rate_limiting_alternate_routed_req_total{ocscp_nf_type="UDM"} - ocscp_metric_egress_rate_limiting_alternate_routed_req_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_egress_rate_limiting_alternate_routed_req_total{ocscp_producer_host="udm1svc.default.svc.cluster.local"} - ocscp_metric_egress_rate_limiting_alternate_routed_req_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_egress_rate_limiting_alternate_routed_req_total{ocscp_egress_rl_bucketkey="egress_6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_egress_rate_limiting_alternate_routed_req_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_egress_rate_limiting_alternate_routed_req_total{ocscp_consumer_host="smf1.scpsvc.svc.cluster.local"} - ocscp_metric_egress_rate_limiting_alternate_routed_req_total{ocscp_consumer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_metric_egress_rate_limiting_alternate_routed_req_total{ocscp_consumer_nf_type="SMF"}

Table 5-54 ocscp_metric_egress_rate_limiting_throttle_req_total

Field	Details
Description	Number of requests throttled due to egress rate limiting
Type	Counter
Dimension	- ocscp_nf_type - ocscp_nf_service_type - ocscp_producer_nf_instance_id - ocscp_producer_host - ocscp_consumer_fqdn - ocscp_configured_egress_rate - ocscp_egress_rl_bucketkey - ocscp_fqdn - ocscp_consumer_host - ocscp_consumer_nf_instance_id - ocscp_consumer_nf_type

Table 5-54 (Cont.) ocscp_metric_egress_rate_limiting_throttle_req_total

Field	Details
Example	• ocscp_metric_egress_rate_limiting_throttle_req_total{ ocscp_nf_type="UDM"} • ocscp_metric_egress_rate_limiting_throttle_req_total{ ocscp_nf_service_type="nudm-uecm"} • ocscp_metric_egress_rate_limiting_throttle_req_total{ ocscp_producer_host="udm1svc.default.svc.cluster.local"} • ocscp_metric_egress_rate_limiting_throttle_req_total{ ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} • ocscp_metric_egress_rate_limiting_throttle_req_total{ ocscp_egress_rl_bucketkey="egress_6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} • ocscp_metric_egress_rate_limiting_throttle_req_total{ ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} • ocscp_metric_egress_rate_limiting_throttle_req_total{ ocscp_consumer_host="smf1.scpsvc.svc.cluster.local"} • ocscp_metric_egress_rate_limiting_throttle_req_total{ ocscp_consumer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} • ocscp_metric_egress_rate_limiting_throttle_req_total{ ocscp_consumer_nf_type="SMF"}

Table 5-55 ocscp_metric_ingress_rate_limiting_not_applied_req_total

Field	Details
Description	Number of requests routed without ingress rate limiting treatment
Type	Counter
Dimension	• ocscp_consumer_nf_type • ocscp_consumer_host • ocscp_consumer_nf_instance_id • ocscp_ingress_rl_headername • ocscp_ingress_rl_headerformat • ocscp_cause • ocscp_ingress_rl_bucketkey • ocscp_fqdn

Table 5-55 (Cont.) ocscp_metric_ingress_rate_limiting_not_applied_req_total

Field	Details
Example	- ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_nf_type="UDM"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_consumer_host="smf1.scpsvc.svc.cluster.local"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_consumer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_ingress_rl_headername="User-Agent"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_ingress_rl_headerformat="NFTYPE-FQDN NFINSTANCEID"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_cause="CONFIGURATION_NOT_FOUND"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp.ingress.rl.bucketkey="ingress_6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{app="ocscp-scp-worker",app_kubernetes_io_component="microService",app_kubernetes_io_instance="ocscp",app_kubernetes_io_managed_by="Helm",app_kubernetes_io_name="scp-worker",app_kubernetes_io_part_of="ocscp",app_kubernetes_io_version="22.4.0.0",application_name="ocscp",coherenceCluster="scp-coherence-cluster",exported_vendor="Oracle",helm_sh_chart="scp-worker-22.4.0",instance="192.168.141.38:8091",job="kubernetes-pods",kubernetes_namespace="scpsvc",kubernetes_pod_name="ocscp-scp-worker-5897bd5b76-wvbm",microservice="scp-worker",ocscp_cause="CONFIGURATION_NOT_FOUND",ocscp_consumer_fqdn="NA",ocscp_consumer_nf_instance_id="NA",ocscp_consumer_nf_type="NA",ocscp_ingress_rl_bucketkey="NA",ocscp_ingress_rl_headerformat="NFTYPE-FQDN NFINSTANCEID",ocscp_ingress_rl_headername="User-Agent",pod_template_hash="5897bd5b76",ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local",vendor="oracle"}

Table 5-56 ocscp_metric_ingress_rate_limiting_throttle_req_total

Field	Details
Description	Number of requests throttled due to ingress rate limiting
Type	Counter

Table 5-56 (Cont.) ocscp_metric_ingress_rate_limiting_throttle_req_total

Field	Details
Dimension	• ocscp_consumer_nf_type • ocscp_consumer_nf_instance_id • ocscp_consumer_host • ocscp_configured_ingress_rate • ocscp_ingress_rl_bucketkey • ocscp_fqdn • ocscp_ingress_rl_headername • ocscp_ingress_rl_headerformat

Table 5-56 (Cont.) ocscp_metric_ingress_rate_limiting_throttle_req_total

Field	Details
Example	- ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_nf_type="UDM"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_consumer_host="smf1.scpsvc.svc.cluster.local"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_consumer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_ingress_rl_headername="User-Agent"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_ingress_rl_headerformat="NFTYPE-FQDN-NFINSTANCEID"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_configured_ingress_rate="40 messages per 1 seconds with Burst of 100 %"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp.ingress.rl.bucketkey="ingress_6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_ingress_rate_limiting_not_applied_req_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_ingress_rate_limiting_throttle_req_total{app="ocscp-scp-worker",app_kubernetes_io_component="microService",app_kubernetes_io_instance="ocscp",app_kubernetes_io_managed_by="Helm",app_kubernetes_io_name="scp-worker",app_kubernetes_io_part_of="ocscp",app_kubernetes_io_version="22.4.0.0",application_name="ocscp",coherenceCluster="scp-coherence-cluster",exported_vendor="Oracle",helm_sh_chart="scp-worker-22.4.0",instance="192.168.141.38:8091",job="kubernetes-pods",kubernetes_namespace="scpsvc",kubernetes_pod_name="ocscp-scp-worker-5897bd5b76-wvbm",microservice="scp-worker",ocscp_configured_ingress_rate="40 messages per 1 seconds with Burst of 100 %",ocscp_consumer_fqdn="smf1svc.scpsvc.svc.cluster.local",ocscp_consumer_nf_instance_id="2faf1bbc-6e4a-4454-acd7-a14ef8efbc6b",ocscp_consumer_nf_type="SMF",ocscp_ingress_rl_bucketkey="ingress_2faf1bbc-6e4a-4454-acd7-a14ef8efbc6b",ocscp_ingress_rl_headerformat="NFTYPE-FQDN",ocscp_ingress_rl_headername="User-Agent",pod_template_hash="5897bd5b76",ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local",vendor="oracle"}

Table 5-57 ocscp_notification_received_different_nrfregionorsetid_total

Field	Details
Description	Number of duplicate notifications skipped with the nrfRegionOrSetId update
Type	Counter
Dimension	• nfType • nfiInstanceid • nrfRegionOrSetId • ocscp_fqdn
Example	ocscp_notification_received_different_nrfregionorsetid_total{nfType="UDM"} ocscp_notification_received_different_nrfregionorsetid_total{nfiInstanceid="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} ocscp_notification_received_different_nrfregionorsetid_total{nrfRegionOrSetId="Reg1"} ocscp_notification_received_different_nrfregionorsetid_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-58 ocscp_audit_2xx_empty_nf_array_rx_total

Field	Details
Description	Number of 2xx responses with empty body
Type	Counter
Dimension	• nrfRegionOrSetId • nfType • ocscp_fqdn
Example	ocscp_audit_2xx_empty_nf_array_rx_total{nrfRegionOrSetId="Reg2"} ocscp_audit_2xx_empty_nf_array_rx_total{nfType="UDM"} ocscp_audit_2xx_empty_nf_array_rx_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-59 ocscp_audit_nrf_failure_response_rx_total

Field	Details
Description	Number of failed responses received from NRF for SCP audit
Type	Counter
Dimension	• nrfRegionOrSetId • nfType • ocscp_fqdn
Example	ocscp_audit_nrf_failure_response_rx_total{nrfRegionOrSetId="Reg1"} ocscp_audit_nrf_failure_response_rx_total{nfType="AUSF"} ocscp_audit_nrf_failure_response_rx_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-60 ocscp_notification_request_total

Field	Details
Description	This metric pegs the number of times notifications received on the notification module.
Type	Counter
Dimension	• eventType • nfStatus • nfType • nfInstanceId • notificationSource • nrfRegionOrSetId • ocscp_fqdn
Example	<pre>ocscp_notification_request_total {eventType="NF_REGISTERED"} ocscp_notification_request_total {nfStatus="REGISTERED"} ocscp_notification_request_total {nfType="UDM"} ocscp_notification_request_total {nfInstanceId="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} ocscp_notification_request_total {notificationSource="NRF"} ocscp_notification_request_total {nrfRegionOrSetId="Reg1"} ocscp_notification_request_total{ocscp_fqdn="ocscp-scp- worker.scpsvc.svc.cluster.local"}</pre>

Table 5-61 ocscp_audit_tx_req_total

Field	Details
Description	This metric pegs the number of times audit requests are initiated for particular NFType, nrfRegionOrSetId, and nrfHost by the scpc-Audit service. This metric is pegged for the scheduled Audit cycle.
Type	Counter
Dimension	• auditType • nrfHost • nfType • auditMode • nrfRegionOrSetId • ocscp_fqdn
Example	<pre>ocscp_audit_tx_req_total{auditType="onDemand"} ocscp_audit_tx_req_total{nrfHost="nrf.svc.cluster.local"} ocscp_audit_tx_req_total{nfType="AMF"} ocscp_audit_tx_req_total{auditMode="nnrf-disc"} ocscp_audit_tx_req_total{nrfRegionOrSetId="Reg1"} ocscp_audit_tx_req_total{ocscp_fqdn="ocscp-scp- worker.scpsvc.svc.cluster.local"}</pre>

Table 5-62 ocscp_audit_rx_res_total

Field	Details
Description	This metric pegs the number of times a response has been received for an initiated audit request. This metric is pegged for the scheduled Audit cycle.
Type	Counter
Dimension	• auditType • nrfHost • nfType • auditMode • nrfRegionOrSetId • responseCodeClass • responseCode • ocscp_fqdn
Example	<pre>ocscp_audit_rx_res_total{auditType="onDemand"} ocscp_audit_rx_res_total{nrfHost="nrf.svc.cluster.local"} ocscp_audit_rx_res_total{nfType="AMF"} ocscp_audit_rx_res_total{auditMode="nnrf-disc"} ocscp_audit_rx_res_total{nrfRegionOrSetId="Reg1"} ocscp_audit_rx_res_total{responseCodeClass="2xx"} ocscp_audit_rx_res_total{responseCode="201"} ocscp_audit_rx_res_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}</pre>

Table 5-63 ocscp_metric_interscp_http_tx_req_total

Field	Details
Description	Number of HTTP requests forwarded to the next hop SCP Here, the dimension ocscp_producer_nf_instance_id indicates the ninstanceid of nexthopscp, and the dimension ocscp_producer_service_instance_id indicates the service instance ID of nexthopscp. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter

Table 5-63 (Cont.) ocscp_metric_interscp_http_tx_req_total

Field	Details
Dimension	Non OCI Dimensions • ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) • ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) • ocscp_scp_end_point (Default value = NA, scp51svc.default.svc.cluster.local:8080) • ocscp_locality (SCP own locality e.g., Loc6, Loc7, etc.) • ocscp_producer_nf_instance_id (e.g., 2fbf1bbc-6e4b-3994-b507-b14ef8e1bc51, etc. Default value = NA) • ocscp_producer_service_instance_id (e.g., fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) • ocscp_producer_nf_fqdn (default value = NA, Contains fqdn of NF pcf1svc.default.svc.cluster.local) • ocscp_nf_setid (default value = "UNKNOWN") • ocscp_nexthop_ocscp_fqdn (contains fqdn of other scp scp51svc.default.svc.cluster.local) • ocscp_fqdn (contains fqdn of scp ocscp-scp-worker.scpsvc.svc.cluster.local) Non OCI Dimensions • NFSERVICETYPE • SCPENDPOINT • NfId • PRODUCERNFFQDN • NEXTHOPSCPFQDN • NFSETID • SCPAUTHORITY
Example	• ocscp_metric_interscp_http_tx_req_total{ocscp_nf_type="PCF"} • ocscp_metric_interscp_http_tx_req_total{ocscp_nf_service_type="npcf-smpolicycontrol"} • ocscp_metric_interscp_http_tx_req_total{ocscp_scp_end_point="10.96.166.65:80"} • ocscp_metric_interscp_http_tx_req_total{ocscp_locality="Loc7"} • ocscp_metric_interscp_http_tx_req_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} • ocscp_metric_interscp_http_tx_req_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} • ocscp_metric_interscp_http_tx_req_total{ocscp_producer_nf_fqdn="pcf1svc.default.svc.cluster.local"} • ocscp_metric_interscp_http_tx_req_total{ocscp_nexthop_ocscp_fqdn="scp51svc.default.svc.cluster.local"} • ocscp_metric_interscp_http_tx_req_total{ocscp_fqdn="scp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-64 ocscp_metric_interscp_http_rx_res_total

Field	Details
Description	Number of HTTP responses received from SCP with specific HTTP response codes, for example, 201, 503, and so on. Here, the dimension ocscp_producer_nf_instance_id indicates the ninstanceid of the nexthopscp, and the dimension ocscp_producer_service_instance_id indicates the service instance ID of the nexthopscp. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimension	Non OCI Dimensions - ocscp_response_code (e.g., 201, 503, 404, etc.) - ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm,npcf-smpolicycontrol etc.) - ocscp_scp_end_point (Default value = NA, scp52svc.default.svc.cluster.local:8080) - ocscp_locality (SCP own locality e.g., Loc6, Loc7, etc.) - ocscp_producer_nf_instance_id (e.g., 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc. Default value = NA) - ocscp_producer_service_instance_id (e.g., fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_res_msg_size (The response message size buckets. e.g., 1 (size < 1k bytes), 2 (1k < size < 2k bytes), 4 (2k < size < 4k), etc.) - ocscp_response_code_class (e.g., 2, 5, etc.) - ocscp_producer_nf_fqdn (default value = NA, Contains fqdn of NF pcf1svc.default.svc.cluster.local) - ocscp_nf_setid(default value ="UNKNOWN") - ocscp_nexthop_ocscp_fqdn(contains fqdn of other scp scp51svc.default.svc.cluster.local) - ocscp_fqdn(contains fqdn of scp) OCI Dimensions - NFSERVICETYPE - SCPENDPOINT - Nfid - RESPONSECODE - RESPONSEMSGSIZE - PRODUCERNFFQDN - NEXTHOPSCPFQDN - NFSETID - SCPAUTHORITY

Table 5-64 (Cont.) ocscp_metric_interscp_http_rx_res_total

Field	Details
Example	- ocscp_metric_interscp_http_rx_res_total{ocscp_response_code="201"} - ocscp_metric_interscp_http_rx_res_total{ocscp_nf_type="PCF"} - ocscp_metric_interscp_http_rx_res_total{ocscp_nf_service_type="npcf-smpolicycontrol"} - ocscp_metric_interscp_http_rx_res_total{ocscp_scp_endpoint="scp52svc.default.svc.cluster.local:8080"} - ocscp_metric_interscp_http_rx_res_total{ocscp_locality="Loc7"} - ocscp_metric_interscp_http_rx_res_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_interscp_http_rx_res_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_interscp_http_rx_res_total{ocscp_res_msg_size="2"} - ocscp_metric_interscp_http_rx_res_total{ocscp_response_code_class="2"} - ocscp_metric_interscp_http_rx_res_total{ocscp_producer_nf_fqdn="pcf1svc.default.svc.cluster.local"} - ocscp_metric_interscp_http_rx_res_total{ocscp_nexthop_ocscp_fqdn="scp51svc.default.svc.cluster.local"} - ocscp_metric_interscp_http_rx_res_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-65 ocscp_metric_jetty_client_requests_total

Field	Details
Description	Number of HTTP requests sent to producer NFs by the Jetty client.
Type	Counter
Dimension	- eventtype(eg., onQueued, onBegin, onHeaders, onCommit, onContent, onSuccess, onFailure) - ocscp_fqdn(contains fqdn of scp)
Example	- ocscp_metric_jetty_client_requests_total(eventtype="onBegin") - ocscp_metric_jetty_client_requests_total(ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local")

Table 5-66 ocscp_worker_pod_overload_discard_req_total

Field	Details
Description	Number of requests discarded with error responses through overload control treatment of SCP-worker service when CPU utilization or Pending Transactions exceeds threshold.
Type	Counter

Table 5-66 (Cont.) ocscp_worker_pod_overload_discard_req_total

Field	Details
Dimension	- Consumer NF type - Consumer NF instance - Producer NF FQDN (if available) - CPU threshold level (Minor/Major/Critical) - HTTP method - SMP priority - ocscp_fqdn - ocscp_consumer_host - ocscp_consumer_nf_instance_id - ocscp_consumer_nf_type - ocscp_consumer_nf_unique_id
Example	- ocscp_metric_overload_discard_with_error_res_total{cpu_level="MINOR"} - ocscp_metric_overload_discard_with_error_res_total{http_method="PATCH"} - ocscp_metric_overload_discard_with_error_res_total{ocscp_producer_host="pcf1svc.scpsvc.svc.cluster.local"} - ocscp_metric_overload_discard_with_error_res_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_overload_discard_with_error_res_total{sbimessage_priority="19"}

Table 5-67 ocscp_alternate_resolution_dnssrv_req_total

Field	Details
Description	Indicates the total number of requests sent to DNS SRV.
Type	Counter
Dimension	- ocscp_fqdn - service_proto_name
Example	- ocscp_alternate_resolution_dnssrv_req_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_alternate_resolution_dnssrv_req_total{service_proto_name="_http._tcp.nf2stub.scpsvc.sv"}

Table 5-68 ocscp_alternate_resolution_dnssrv_res_total

Field	Details
Description	Indicates the total number of DNS SRV queries received from the DNS server.
Type	Counter
Dimension	- ocscp_fqdn - service_proto_name - status(success/failure) - cause(timed_out, host_not_found, network_error, referral, name_too_long, na, error_at_scp, unknown, servfail)

Table 5-68 (Cont.) ocscp_alternate_resolution_dnssrv_res_total

Field	Details
Example	- ocscp_alternate_resolution_dnssrv_res_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_alternate_resolution_dnssrv_res_total{service_protocol_name="_http._tcp.nf2stub.scpsvc.sv"} - ocscp_alternate_resolution_dnssrv_res_total{status="success"} - ocscp_alternate_resolution_dnssrv_res_total{cause="timed_out", "host_not_found", "network_error", "referral", "name_too_long", "na", "error_at_scp", "unknown", "servfail"}

Table 5-69 ocscp_metric_http_ar_tx_req_total

Field	Details
Description	Indicates the total number of alternate route attempts made by SCP. Note: Here, the instanceid of SEPP is used for interplmn scenarios, the instanceid of SCP is used for interscp scenarios, otherwise, the instanceid of the producer is utilized. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimension	- ocscp_nf_type - ocscp_nf_service_type - ocscp_primary_producer_fqdn - ocscp_primary_nf_end_point - ocscp_alternate_producer_fqdn - ocscp_alternate_nf_end_point - ocscp_locality - ocscp_alternate_producer_nf_instance_id - ocscp_alternate_producer_service_instance_id - ocscp_nf_setid - ocscp_alternate_route_type Note: In the notification message type, certain dimension values such as ocscp_nf_service_type, ocscp_nf_setid, and ocscp_nf_type may not qualify for pegging. In the scenario where the service name cannot be derived from the PATH URI, pegging will not take place, and if these dimensions are identified through headers, pegging will occur.

Table 5-69 (Cont.) ocscp_metric_http_ar_tx_req_total

Field	Details
Example	- ocscp_metric_http_ar_tx_req_total{ocscp_alternate_route_type="DNS_SRV"} - ocscp_metric_http_ar_tx_req_total{ocscp_alternate_route_type="STATIC_CONFIG"} - ocscp_metric_http_ar_tx_req_total{ocscp_nf_type="PCF"} - ocscp_metric_http_ar_tx_req_total{ocscp_nf_service_type="npcf-smpolicycontrol"} - ocscp_metric_http_ar_tx_req_total{ocscp_alternate_nf_end_point="pcf2svc.default.svc.cluster.local:8080"} - ocscp_metric_http_ar_tx_req_total{ocscp_locality="Loc7"} - ocscp_metric_http_ar_tx_req_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_http_ar_tx_req_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_http_ar_tx_req_total{ocscp_primary_producer_fqdn="pcf1svc.default.svc.cluster.local"} - ocscp_metric_http_ar_tx_req_total{ocscp_alternate_producer_fqdn="pcf2svc.default.svc.cluster.local"} - ocscp_metric_http_ar_tx_req_total{ocscp_nf_setid="Loc7"}

Table 5-70 ocscp_metric_sidecarproxy_failures_total

Field	Details
Description	Number of error responses received from service mesh sidecar proxy because of transport error towards producer NFs.
Type	Counter
Dimension	- ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_locality (SCP site name e.g., Loc6, Loc7, etc.) - ocscp_producer_nf_instance_id (e.g., 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc. Default value = NA) - ocscp_producer_service_instance_id (e.g., fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_producer_host (default value = NA, Contains fqdn of producer) - ocscp_fqdn - response_code - ocscp_server_header

Table 5-70 (Cont.) ocscp_metric_sidecarproxy_failures_total

Field	Details
Example	- ocscp_metric_sidecarproxy_failures_total{ocscp_nf_type="UDM"} - ocscp_metric_sidecarproxy_failures_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_sidecarproxy_failures_total{ocscp_locality="Loc7"} - ocscp_metric_sidecarproxy_failures_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_sidecarproxy_failures_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_sidecarproxy_failures_total{ocscp_producer_host="udm1svc.default.svc.cluster.local"} - ocscp_metric_sidecarproxy_failures_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_sidecarproxy_failures_total{ocscp_response_code="503"} - ocscp_metric_sidecarproxy_failures_total{ocscp_server_header="envoy"}

Table 5-71 ocscp_metric_remote_response_without_server_header_total

Field	Details
Description	Number of times SCP adds the server header in the error response received from producer NFs without server header.
Type	Counter
Dimension	- ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_locality (SCP site name e.g., Loc6, Loc7, etc.) - ocscp_producer_nf_instance_id (e.g., 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc. Default value = NA) - ocscp_producer_service_instance_id (e.g., fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_producer_host (default value = NA, Contains fqdn of producer) - ocscp_fqdn

Table 5-71 (Cont.) ocscp_metric_remote_response_without_server_header_total

Field	Details
Example	- ocscp_metric_remote_response_without_server_header_total{ocscp_nf_type="UDM"} - ocscp_metric_remote_response_without_server_header_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_remote_response_without_server_header_total{ocscp_locality="Loc7"} - ocscp_metric_remote_response_without_server_header_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_remote_response_without_server_header_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_remote_response_without_server_header_total{ocscp_producer_host="udm1svc.default.svc.cluster.local"} - ocscp_metric_remote_response_without_server_header_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-72 ocscp_metric_add_or_update_server_header_custom_behavior_total

Field	Details
Description	Number of times SCP adds or updates the server header in the error response received from producer NFs.
Type	Counter
Dimension	- ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_locality (SCP site name e.g., Loc6, Loc7, etc.) - ocscp_producer_nf_instance_id (e.g., 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc. Default value = NA) - ocscp_producer_service_instance_id (e.g., fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_producer_host (default value = NA, Contains fqdn of producer) - ocscp_fqdn

Table 5-72 (Cont.) ocscp_metric_add_or_update_server_header_custom_behavior_total

Field	Details
Example	- ocscp_metric_add_or_update_server_header_custom_behavior_total{ocscp_nf_type="UDM"} - ocscp_metric_add_or_update_server_header_custom_behavior_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_add_or_update_server_header_custom_behavior_total{ocscp_locality="Loc7"} - ocscp_metric_add_or_update_server_header_custom_behavior_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_add_or_update_server_header_custom_behavior_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_add_or_update_server_header_custom_behavior_total{ocscp_producer_host="udm1svc.default.svc.cluster.local"} - ocscp_metric_add_or_update_server_header_custom_behavior_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-73 ocscp_metric_add_server_header_default_behavior_total

Field	Details
Description	Number of times SCP adds the server header in the error response received from producer NFs.
Type	Counter
Dimension	- ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_locality (SCP site name e.g., Loc6, Loc7, etc.) - ocscp_producer_nf_instance_id (e.g., 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc. Default value = NA) - ocscp_producer_service_instance_id (e.g., fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_producer_host (default value = NA, Contains fqdn of producer) - ocscp_fqdn

Table 5-73 (Cont.) ocscp_metric_add_server_header_default_behavior_total

Field	Details
Example	- ocscp_metric_add_server_header_default_behavior_total{ocscp_nf_type="UDM"} - ocscp_metric_add_server_header_default_behavior_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_add_server_header_default_behavior_total{ocscp_locality="Loc7"} - ocscp_metric_add_server_header_default_behavior_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c"} - ocscp_metric_add_server_header_default_behavior_total{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"} - ocscp_metric_add_server_header_default_behavior_total{ocscp_producer_host="udm1svc.default.svc.cluster.local"} - ocscp_metric_add_server_header_default_behavior_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-74 ocscp_metric_failure_from_peer_scp_total

Field	Details
Description	Number of times SCP receives error responses with server header containing only the next hop FQDN of SCP.
Type	Counter
Dimension	- ocscp_locality (SCP site name e.g., Loc6, Loc7, etc.) - ocscp_peer_scp_fqdn (default value = NA, Contains fqdn of producer) - ocscp_fqdn
Example	- ocscp_metric_failure_from_peer_scp_total{ocscp_locality="Loc7"} - ocscp_metric_failure_from_peer_scp_total{ocscp_peer_scp_fqdn="udm1svc.default.svc.cluster.local"} - ocscp_metric_failure_from_peer_scp_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-75 ocscp_peer_scp_unhealthy

Field	Details
Description	Number of times (as gauge) the peer SCP is marked as unavailable.
Type	Gauge
Dimension	- ocscp_locality (SCP site name e.g., Loc6, Loc7, etc.) - ocscp_peer_scp_fqdn (default value = NA, Contains fqdn of producer) - ocscp_fqdn

Table 5-75 (Cont.) ocscp_peer_scp_unhealthy

Field	Details
Example	- ocscp_peer_scp_unhealthy{ocscp_locality="Loc7"} - ocscp_peer_scp_unhealthy(ocscp_peer_scp_fqdn="udm1svc.default.svc.cluster.local") - ocscp_peer_scp_unhealthy{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-76 ocscp_alterate_resolution_dnssrv_rx_error_res

Field	Details
Description	Indicates failed DNS SRV query responses.
Type	Gauge
Dimension	- cause (network_error, timed_out) - ocscp_fqdn (ocscp-scp-worker.scpsvc.svc.cluster.local)
Example	- ocscp_alterate_resolution_dnssrv_rx_error_res {cause="timed_out", "host_not_found", "network_error", "referral", "name_too_long", "na", "error_at_scp", "unknown", "servfail"} - ocscp_alterate_resolution_dnssrv_rx_error_res {ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-77 ocscp_load_manager_peer_load_throttled_threshold

Field	Details
Description	Indicates that the peer load has exceeded the Throttled threshold. This metric is pegged by the load manager pod.
Type	Gauge
Dimension	- ocscp_peer_fqdn - ocscp_peer_nf_instance_id - ocscp_peer_service_instance_id - ocscp_peer_service_name
Example	ocscp_load_manager_peer_load_throttled_threshold{ocscp_peer_fqdn="smf1svc.scpsvc.svc.cluster.local",ocscp_peer_nf_instance_id="11111112-aaaa-aaaa-aaaa-111111111111",ocscp_peer_service_instance_id="3aad8c63-0cc0-4b74-9582-4e6475214ae8",ocscp_peer_service_name="nsmf-pdusession"}

Table 5-78 ocscp_load_manager_peer_load_alterateRoute_threshold

Field	Details
Description	Indicates that the peer load has exceeded the Alternate Route state threshold. This metric is pegged by the load manager pod.
Type	Gauge

Table 5-78 (Cont.) ocscp_load_manager_peer_load_alternateRoute_threshold

Field	Details
Dimension	- ocscp_peer_fqdn - ocscp_peer_nf_instance_id - ocscp_peer_service_instance_id - ocscp_peer_service_name
Example	ocscp_load_manager_peer_load_alternateRoute_threshold{ocscp_peer_fqdn="smf1svc.scpsvc.svc.cluster.local",ocscp_peer_nf_instance_id="11111112-aaaa-aaaa-aaaa-111111111111",ocscp_peer_service_instance_id="3aad8c63-0cc0-4b74-9582-4e6475214ae8",ocscp_peer_service_name="nsmf-pdusession"}

Table 5-79 ocscp_metric_ondemand_discovery_tx_req_total

Field	Details
Description	Number of times SCP worker sends on-demand discovery request to nrfproxy.
Type	Counter
Dimension	- ocscp_nf_type (e.g.: UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g.: nudm-uecm, nudm-sdm, etc.) - ocscp_consumer_nf_type (e.g.: UDM, PCF, AMF, etc.) - ocscp_fqdn - subscriber_info_present_in_discovery_headers (If any of the following discovery headers is present in the request: 3gpp-sbi-discovery-supi, 3gpp-sbi-discovery-gpsi, 3gpp-sbi-discovery-external-group-identity, 3gpp-sbi-discovery-routing-indicator, 3gpp-sbi-discovery-msisdn, 3gpp-sbi-discovery-imsi)
Example	- ocscp_metric_ondemand_discovery_tx_req_total{ocscp_nf_type="UDM"} - ocscp_metric_ondemand_discovery_tx_req_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_ondemand_discovery_tx_req_total{ocscp_consumer_nf_type="UDM"}

Table 5-80 ocscp_metric_ondemand_discovery_rx_res_total

Field	Details
Description	Number of times SCP worker receives on-demand discovery response from nrfproxy.
Type	Counter

Table 5-80 (Cont.) ocscp_metric_ondemand_discovery_rx_res_total

Field	Details
Dimension	- ocscp_nf_type (e.g.: UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g.: nudm-uecm, nudm-sdm, etc.) - ocscp_consumer_nf_type (e.g.: UDM, PCF, AMF, etc.) - ocscp_fqdn - ocscp_response_code - ocscp_response_code_class - subscriber_info_present_in_discovery_headers
Example	- ocscp_metric_ondemand_discovery_rx_res_to tal{ocscp_nf_type="UDM"} - ocscp_metric_ondemand_discovery_rx_res_to tal{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_ondemand_discovery_rx_res_to tal{ocscp_consumer_nf_type="AMF"} - ocscp_metric_ondemand_discovery_rx_res_to tal{ocscp_response_code="201"} - ocscp_metric_ondemand_discovery_rx_res_to tal{ocscp_response_code_class="2"}

Table 5-81 ocscp_metric_ondemand_discovery_cache_match_total

Field	Details
Description	Number of times SCP worker is able to resolve discovery requests using local cache.
Type	Counter
Dimension	- ocscp_nf_type (e.g.: UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g.: nudm-uecm, nudm-sdm, etc.) - ocscp_consumer_nf_type (e.g.: UDM, PCF, AMF, etc.) - ocscp_fqdn - subscriber_info_present_in_discovery_headers
Example	- ocscp_metric_ondemand_discovery_cache_match_tota l{ocscp_nf_type="UDM"} - ocscp_metric_ondemand_discovery_cache_match_tota l{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_ondemand_discovery_cache_match_tota l{ocscp_consumer_nf_type="AMF"}

Table 5-82 ocscp_nrf_nf_discovery_tx_req_total

Field	Details
Description	The total number of discovery requests sent to NRF by nrfproxy.
Type	Counter

Table 5-82 (Cont.) ocscp_nrf_nf_discovery_tx_req_total

Field	Details
Dimension	• ocscp_producer_nf_type (e.g.: UDM, PCF, AMF, etc.) • ocscp_consumer_nf_type (e.g.: UDM, PCF, AMF, etc.) • ocscp_nf_service_name (e.g.: nudm-uecm, nudm-sdm, etc.) • ocscp_fqdn • subscriber_info_present (If any of the following discovery headers present in the request: 3gpp-sbi-discovery-supi, 3gpp-sbi-discovery-gpsi, 3gpp-sbi-discovery-external-group-identity, 3gpp-sbi-discovery-routing-indicator, 3gpp-sbi-discovery-msisdn, 3gpp-sbi-discovery-imsi) • is_inter_plmn (e.g.: true for inter plmn; false for intra plmn)
Example	• ocscp_nrf_nf_discovery_tx_req_total{ocscp_producer_nf_type="UDM"} • ocscp_nrf_nf_discovery_tx_req_total{ocscp_consumer_nf_type="AMF"} • ocscp_nrf_nf_discovery_tx_req_total{ocscp_nf_service_name="nudm-uecm"} • ocscp_nrf_nf_discovery_tx_req_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-83 ocscp_nrf_nf_discovery_rx_res_total

Field	Details
Description	The total number of discovery responses received from NRF by nrfproxy.
Type	Counter
Dimension	• ocscp_producer_nf_type (e.g.: UDM, PCF, AMF, etc.) • ocscp_consumer_nf_type (e.g.: UDM, PCF, AMF, etc.) • ocscp_nf_service_name (e.g.: nudm-uecm, nudm-sdm, etc.) • ocscp_fqdn • subscriber_info_present • is_inter_plmn • status_code
Example	• ocscp_nrf_nf_discovery_tx_req_total{ocscp_producer_nf_type="UDM"} • ocscp_nrf_nf_discovery_tx_req_total{ocscp_consumer_nf_type="AMF"} • ocscp_nrf_nf_discovery_tx_req_total{ocscp_nf_service_name="nudm-uecm"} • ocscp_nrf_nf_discovery_tx_req_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} • ocscp_nrf_nf_discovery_tx_req_total{status_code="200"}

Table 5-84 ocscp_nrf_nf_discovery_rx_req_total

Field	Details
Description	The total number of discovery requests received by nrfproxy from SCP worker.
Type	Counter
Dimension	- ocscp_producer_nf_type (e.g.: UDM, PCF, AMF, etc.) - ocscp_consumer_nf_type (e.g.: UDM, PCF, AMF, etc.) - ocscp_nf_service_name (e.g.: nudm-uecm, nudm-sdm, etc.) - ocscp_fqdn - subscriber_info_present - is_inter_plmn
Example	- ocscp_nrf_nf_discovery_tx_req_total{ocscp_producer_nf_type="UDM"} - ocscp_nrf_nf_discovery_tx_req_total{ocscp_consumer_nf_type="AMF"} - ocscp_nrf_nf_discovery_tx_req_total{ocscp_nf_service_name="nudm-uecm"} - ocscp_nrf_nf_discovery_tx_req_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-85 ocscp_nrf_nf_discovery_tx_res_total

Field	Details
Description	The total number of discovery responses sent by nrfproxy to SCP worker.
Type	Counter
Dimension	- ocscp_producer_nf_type (e.g.: UDM, PCF, AMF, etc.) - ocscp_consumer_nf_type (e.g.: UDM, PCF, AMF, etc.) - ocscp_nf_service_name (e.g.: nudm-uecm, nudm-sdm, etc.) - ocscp_fqdn - subscriber_info_present - is_inter_plmn - status_code - is_from_local_cache (Whether the discovery process was made from local cache (true) or NRF (false))
Example	- ocscp_nrf_nf_discovery_tx_req_total{ocscp_producer_nf_type="UDM"} - ocscp_nrf_nf_discovery_tx_req_total{ocscp_consumer_nf_type="AMF"} - ocscp_nrf_nf_discovery_tx_req_total{ocscp_nf_service_name="nudm-uecm"} - ocscp_nrf_nf_discovery_tx_req_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_nrf_nf_discovery_tx_req_total{status_code="200"}

Table 5-86 ocscp_nrfproxy_generated_error_response_total

Field	Details
Description	The total number of error responses generated by nrfproxy excluding NRF returned errors.
Type	Counter
Dimension	- ocscp_producer_nf_type (e.g.: UDM, PCF, AMF, etc.) - ocscp_consumer_nf_type (e.g.: UDM, PCF, AMF, etc.) - ocscp_nf_service_name (e.g.: nudm-uecm, nudm-sdm, etc.) - ocscp_fqdn - subscriber_info_present - is_inter_plmn - status_code - valid_NRF_response_to_fail (Whether SCP convert 200 OK (NRF) to 400 error, due to empty profile list or not)
Example	- ocscp_nrf_nf_discovery_tx_req_total{ocscp_producer_nf_type="UDM"} - ocscp_nrf_nf_discovery_tx_req_total{ocscp_consumer_nf_type="AMF"} - ocscp_nrf_nf_discovery_tx_req_total{ocscp_nf_service_name="nudm-uecm"} - ocscp_nrf_nf_discovery_tx_req_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_nrf_nf_discovery_tx_req_total{status_code="200"}

Table 5-87 ocscp_metric_sepp_http_rx_req_total

Field	Details
Description	The total number of HTTP requests routed towards SEPP. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimension	- ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_nf_end_point (Default value = NA) ocscp_locality (SCP site name e.g., Loc6, Loc7, etc.) - ocscp_producer_nf_instance_id (e.g., 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc. Default value = NA) - ocscp_producer_service_instance_id (e.g., fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_sepp_fqdn (default value = NA, Contains fqdn of producer) - plmn_id(mnc123-mcc254) - ocscp_authority

Table 5-87 (Cont.) ocscp_metric_sepp_http_rx_req_total

Field	Details
Example	- ocscp_metric_sepp_http_tx_req_total{ocscp_nf_type="UDM"} - ocscp_metric_sepp_http_tx_req_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_sepp_http_tx_req_total{ocscp_sepp_fqdn="sepp1svc.scpsvc.svc.cluster.local"} - ocscp_metric_sepp_http_tx_req_total{plmn_id="mnc123-mcc100"}

Table 5-88 ocscp_metric_sepp_http_rx_res_total

Field	Details
Description	The total number of HTTP responses received from SEPP. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimension	Non OCI Dimensions - ocscp_response_code (e.g., 201, 503, 404, etc.) - ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_nf_end_point (Default value = NA) - ocscp_locality (SCP site name e.g., Loc6, Loc7, etc.) - ocscp_producer_nf_instance_id (e.g., 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc. Default value = NA) - ocscp_producer_service_instance_id (e.g., fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_res_msg_size (The response message size buckets. e.g., 1 (size < 1k bytes), 2 (1k < size < 2k bytes), 4 (2k < size < 4k), etc.) - ocscp_response_code_class (e.g., 2, 5, etc.) - ocscp_sepp_fqdn (default value = NA, Contains fqdn of producer) - plmn_id(mnc123-mcc254) - ocscp_authority OCI Dimensions - NFSERVICETYPE - NFENDPOINT - NFINSTANCEID - SERVICEINSTANCEID - RESPONSECODE - RESPONSEMSGSIZE - SEPP_FQDN - PLMNID - SCPAUTHORITY

Table 5-88 (Cont.) ocscp_metric_sepp_http_rx_res_total

Field	Details
Example	- ocscp_metric_sepp_http_rx_res_total{ocscp_nf_type="UDM"} - ocscp_metric_sepp_http_rx_res_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_sepp_http_rx_res_total{ocscp_sepp_fqdn="sepp1svc.scpsvc.svc.cluster.local"} - ocscp_metric_sepp_http_rx_res_total{plmn_id="mnc123-mcc100"}

Table 5-89 ocscp_metric_sepp_http_tx_res_total

Field	Details
Description	The total number of HTTP responses routed towards SEPP. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimension	- ocscp_response_code (e.g., 201, 503, 404, etc.) - ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_locality (SCP site name e.g., Loc6, Loc7, etc.) - ocscp_response_code_class (e.g., 2, 5, etc.) - ocscp_sepp_fqdn (default value = NA, Contains fqdn of producer) - plmn_id(mnc123-mcc254) - ocscp_authority
Example	- ocscp_metric_sepp_http_tx_res_total{ocscp_nf_type="UDM"} - ocscp_metric_sepp_http_tx_res_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_sepp_http_tx_res_total{ocscp_sepp_fqdn="sepp1svc.scpsvc.svc.cluster.local"} - ocscp_metric_sepp_http_tx_res_total{plmn_id="mnc123-mcc100"}

Table 5-90 ocscp_metric_sepp_routing_attempt_fail_total

Field	Details
Description	The total number of inter-PLMN requests routing failed towards SEPP.
Type	Counter

Table 5-90 (Cont.) ocscp_metric_sepp_routing_attempt_fail_total

Field	Details
Dimension	- ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_nf_end_point (Default value = NA) - ocscp_locality (SCP site name e.g., Loc6, Loc7, etc.) - ocscp_producer_nf_instance_id (e.g., 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc. Default value = NA) - ocscp_producer_service_instance_id (e.g., fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_sepp_fqdn (default value = NA, Contains fqdn of producer) - plmn_id(mnc123-mcc254) - ocscp_authority - ocscp_error_group
Example	- ocscp_metric_sepp_routing_attempt_fail_current{ocscp_nf_type="UDM"} - ocscp_metric_sepp_routing_attempt_fail_current{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_sepp_routing_attempt_fail_current{ocscp_sepp_fqdn="sepp1svc.scpsvc.svc.cluster.local"} - ocscp_metric_sepp_routing_attempt_fail_current{plmn_id="mnc123-mcc100"}

Table 5-91 ocscp_metric_sepp_routing_attempt_fail_current

Field	Details
Description	Set to 1 if inter-plmn requests routing failed towards SEPP.
Type	Gauge
Dimension	- ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_nf_end_point (Default value = NA) - ocscp_locality (SCP site name e.g., Loc6, Loc7, etc.) - ocscp_producer_nf_instance_id (e.g., 6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c, etc. Default value = NA) - ocscp_producer_service_instance_id (e.g., fe137ab7-740a-46ee-aa5c-951806d77b02, etc. Default value = NA) - ocscp_sepp_fqdn (default value = NA, Contains fqdn of producer) - plmn_id(mnc123-mcc254) - ocscp_authority - ocscp_error_group

Table 5-91 (Cont.) ocscp_metric_sepp_routing_attempt_fail_current

Field	Details
Example	- ocscp_metric_sepp_routing_attempt_fail_current{ocscp_nf_type="UDM"} - ocscp_metric_sepp_routing_attempt_fail_current{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_sepp_routing_attempt_fail_current{ocscp_sepp_fqdn="sepp1svc.scpsvc.svc.cluster.local"} - ocscp_metric_sepp_routing_attempt_fail_current{plmn_id="mnc123-mcc100"}

Table 5-92 ocscp_metric_sepp_not_configured_total

Field	Details
Description	The total number of inter-PLMN requests failed because SEPP is not configured for target PLMN received in interPlmnFqdn from 3gpp-Sbi-Target-apiRoot HTTP header or interPlmnFqdn after discovery response.
Type	Counter
Dimension	- ocscp_locality (SCP site name e.g., Loc6, Loc7, etc.) - plmn_id(mnc123-mcc254) - ocscp_authority
Example	ocscp_metric_sepp_not_configured_total{plmn_id="mnc123-mcc254"}

Table 5-93 ocscp_metric_sepp_not_configured_current

Field	Details
Description	Set to 1 if inter-plmn requests failed because SEPP is not configured for target PLMN received in interPlmnFqdn from 3gpp-Sbi-Target-apiRoot HTTP header or interPlmnFqdn after discovery response.
Type	Gauge
Dimension	- ocscp_locality (SCP site name e.g., Loc6, Loc7, etc.) - plmn_id(mnc123-mcc254) - ocscp_authority
Example	ocscp_metric_sepp_not_configured_current{plmn_id="mnc123-mcc254"}

Table 5-94 ocscp_metric_5gsbi_rx_req_total

Field	Details
Description	The total number of incoming requests. This metric can be used to segregate the ingress requests based on scheme, such as HTTP or HTTPS.
Type	Counter

Table 5-94 (Cont.) ocscp_metric_5gsbi_rx_req_total

Field	Details
Dimension	- ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_nf_end_point (Default value = NA) - ocscp_authority - scheme(e.g., https)
Example	- ocscp_metric_5gsbi_rx_req_total{ocscp_nf_type="PCF"} - ocscp_metric_5gsbi_rx_req_total{ocscp_nf_service_type="npcf-smpolicycontrol"} - ocscp_metric_5gsbi_rx_req_total{scheme="https"} - ocscp_metric_5gsbi_rx_req_total{ocscp_nf_end_point="NA"} - ocscp_metric_5gsbi_rx_req_total{ocscp_authority="ocscp-scp-worker.scpsvc.svc.cluster.local:9443"}

Table 5-95 ocscp_metric_5gsbi_tx_req_total

Field	Details
Description	The total number of requests forwarded by Service Communication Proxy to the upstream cluster. This metric can be used to segregate the egress requests based on the scheme, such as HTTP or HTTPS.
Type	Counter
Dimension	- ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_nf_end_point (Default value = NA) - ocscp_producer_host (default value = NA, Contains fqdn of producer) - ocscp_authority - scheme(e.g., https)
Example	- ocscp_metric_5gsbi_tx_req_total{ocscp_nf_type="PCF"} - ocscp_metric_5gsbi_tx_req_total{ocscp_nf_service_type="npcf-smpolicycontrol"} - ocscp_metric_5gsbi_tx_req_total{scheme="https"} - ocscp_metric_5gsbi_tx_req_total{ocscp_nf_end_point="NA"} - ocscp_metric_5gsbi_tx_req_total{ocscp_authority="ocscp-scp-worker.scpsvc.svc.cluster.local:9443"}

Table 5-96 ocscp_metric_5gsbi_tx_res_total

Field	Details
Description	The total number of responses forwarded by SCP with specific HTTP response codes (for example, 201, 503, and so on). This metric can be used to segregate the egress responses based on a scheme, such as HTTP or HTTPS. Note:The ocscp_authority dimension will get pegged only when it is enabled using the value.yaml file. scpAuthorityMetricLabelDisabled: false
Type	Counter
Dimension	- ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_response_code(e.g., 201, 503, 404, etc.) - ocscp_response_code_class (e.g., 2, 5, etc.) - ocscp_authority - scheme(e.g., https)
Example	- ocscp_metric_5gsbi_tx_res_total{ocscp_nf_type="PCF"} - ocscp_metric_5gsbi_tx_res_total{ocscp_nf_service_type="npcf-smpolicycontrol"} - ocscp_metric_5gsbi_tx_res_total{scheme="https"} - ocscp_metric_5gsbi_tx_res_total{ocscp_authority="ocscp-scp-worker.scpsvc.svc.cluster.local:9443"}

Table 5-97 ocscp_metric_5gsbi_rx_res_total

Field	Details
Description	The total number of responses received by SCP with specific HTTP response codes (for example, 201, 503, and so on.) This metric can be used to segregate the ingress responses based on a scheme, such as HTTP or HTTPS.
Type	Counter
Dimension	- ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_nf_end_point (Default value = NA) - ocscp_response_code(e.g., 201, 503, 404, etc.) - ocscp_response_code_class (e.g., 2, 5, etc.) - ocscp_authority - scheme(e.g., https)
Example	- ocscp_metric_5gsbi_rx_res_total{ocscp_nf_type="PCF"} - ocscp_metric_5gsbi_rx_res_total{ocscp_nf_service_type="npcf-smpolicycontrol"} - ocscp_metric_5gsbi_rx_res_total{scheme="https"} - ocscp_metric_5gsbi_rx_res_total{ocscp_nf_end_point="NA"} - ocscp_metric_5gsbi_rx_res_total{ocscp_authority="ocscp-scp-worker.scpsvc.svc.cluster.local:9443"}

Table 5-98 ocscp_metric_ssl_certificate_expire_total

Field	Details
Description	The state of metric becomes 1 when the server or client certificate is about to expire.
Type	Gauge
Dimension	- ocscp_fqdn - certificate_type (for example, 5G_SBI_SERVER_PRIMARY, 5G_SBI_CLIENT_PRIMARY, 5G_SBI_SERVER_SECONDARY, 5G_SBI_CLIENT_SECONDARY, 5G_SBI_TRAFFIC_FEED_PRIMARY, 5G_SBI_TRAFFIC_FEED_SECONDARY)
Example	ocscp_metric_ssl_certificate_expire_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-99 ocscp_metric_scp_cache_service_unreachable_total

Field	Details
Description	Number of times SCP worker is unable to find SCP cache service.
Type	Counter
Dimension	- coherenceCluster - ocscp_fqdn
Example	ocscp_metric_scp_cache_service_unreachable_total{coherenceCluster="scp-coherence-cluster"scp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-100 ocscp_missed_nf_notifications_total

Field	Details
Description	Number of times the notifications are missed.
Type	Counter
Dimension	nfType
Example	ocscp_missed_nf_notifications_total{ocscp_nf_type="UDM"}

Table 5-101 ocscp_notifications_rejected_topologysource_nrf_total

Field	Details
Description	Number of notifications rejected from NRF as topology source.
Type	Counter
Dimension	nfType
Example	ocscp_notifications_rejected_topologysource_nrf_total{ocscp_nf_type="UDM"}

Table 5-102 ocscp_nrf_notifications_requests_total

Field	Details
Description	Number of NRF notifications received.
Type	Counter
Dimension	nfType
Example	ocscp_nrf_notifications_requests_total{ocscp_nf_type="UDM"}

Table 5-103 ocscp_soothsayer_audit_db_fetch

Field	Details
Description	Indicates the database fetched for audit.
Type	Counter
Dimension	ocscp_soothsayer_audit_db_fetch
Example	ocscp_nrf_notifications_requests_total{ocscp_nf_type="UDM"}

Table 5-104 ocscp_update_topology_source_failure_total

Field	Details
Description	Number of times the topology source information is toggled.
Type	Counter
Dimension	nfType
Example	ocscp_update_topology_source_failure_total{ocscp_nf_type="UDM"}

Table 5-105 ocscp_worker_pod_overload_control_cpu_utilization_warn

Field	Details
Description	Indicates CPU utilization of SCP worker at Warn level.
Type	Gauge
Dimension	- pod_name - scp_service
Example	ocscp_worker_pod_overload_control_cpu_utilization_warn{pod_name="ocscp-scp-worker-6fcc755f9b-5hckw",scp_service="scp-worker"}

Table 5-106 ocscp_worker_pod_overload_control_cpu_utilization_minor

Field	Details
Description	Indicates CPU utilization of SCP worker at Minor level.
Type	Gauge
Dimension	- pod_name - scp_service

Table 5-106 (Cont.) ocscp_worker_pod_overload_control_cpu_utilization_minor

Field	Details
Example	ocscp_worker_pod_overload_control_cpu_utilization_minor{ pod_name="ocscp-scp- worker-6fcc755f9b-5hckw",scp_service="scp-worker"}

Table 5-107 ocscp_worker_pod_overload_control_cpu_utilization_major

Field	Details
Description	Indicates CPU utilization of SCP worker at Major level.
Type	Gauge
Dimension	- pod_name - scp_service
Example	ocscp_worker_pod_overload_control_cpu_utilization_major{ pod_name="ocscp-scp- worker-6fcc755f9b-5hckw",scp_service="scp-worker"}

Table 5-108 ocscp_notifications_suspended_topologysource_nrf_total

Field	Details
Description	Indicates the number of times SUSPEND notifications are received by SCP.
Type	Counter
Dimension	- pod_name - ocscp_producer_nf_instance_id
Example	ocscp_notifications_suspended_topologysource_nrf_total{st atus="suspended"}

Table 5-109 ocscp_notifications_profile_deleted_deregistration_total

Field	Details
Description	Indicates the number of times NF profile is removed due to DEREGISTER ED notification.
Type	Counter
Dimension	- event - ocscp_producer_nf_instance_id
Example	ocscp_notifications_profile_deleted_deregistration_total{eve nt="deregistered"}

Table 5-110 ocscp_metric_ar_attempt_fail_total

Field	Details
Description	Indicates the total number of failed alternate routing attempts made using static configuration.
Type	Counter

Table 5-110 (Cont.) ocscp_metric_ar_attempt_fail_total

Field	Details
Dimension	- ocscp_alternate_route_type - ocscp_nf_type - ocscp_nf_service_type - ocscp_fqdn
Example	ocscp_metric_ar_attempt_fail_total{ocscp_alternate_route_type="STATIC_CONFIG"}

Table 5-111 ocscp_nrf_nf_discovery_request_processing_time_total

Field	Details
Description	Indicates the NRF request processing time by the scp-nrfproxy microservice.
Type	Counter
Dimension	- ocscp_producer_nf_type - ocscp_nf_service_type - ocscp_consumer_nf_type - subscriber_info_present - ocscp_nrf_service_type - is_inter_plmn
Example	- ocscp_nrf_nf_discovery_request_processing_time_total{ocscp_producer_nf_type="PCF"} - ocscp_nrf_nf_discovery_request_processing_time_total{ocscp_consumer_nf_type="PCF"}

Table 5-112 ocscp_nrf_nf_discovery_response_processing_time_total

Field	Details
Description	Indicates the NRF response processing time by the scp-nrfproxy microservice.
Type	Counter
Dimension	- ocscp_producer_nf_type - ocscp_nf_service_type - ocscp_consumer_nf_type - subscriber_info_present - is_inter_plmn - ocscp_nrf_service_type
Example	- ocscp_nrf_nf_discovery_response_processing_time_total{ocscp_producer_nf_type="PCF"} - ocscp_nrf_nf_discovery_response_processing_time_total{ocscp_consumer_nf_type="PCF"}

Table 5-113 ocscp_global_egress_rl_bucketkey_not_rate_controlled_total

Field	Details
Description	Indicates when the local SCP is not performing global rate control while other SCP instances in the scp-cache cluster are performing the rate control.
Type	Counter
Dimension	- ocscp_fqdn - ocscp_egress_rl_bucketkey
Example	- ocscp_global_egress_rl_bucketkey_not_rate_controlled_total(bucketKey="egress_udm3svc.scpsvc.svc.cluster.local") - ocscp_global_egress_rl_bucketkey_not_rate_controlled_total(ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local")

Table 5-114 ocscp_global_egress_rl_remote_participant_connectivity_failure

Field	Details
Description	Monitors the remote coherence participant connectivity status.
Type	Gauge
Dimension	- ocscp_fqdn - scp_local_coh_cluster_name - scp_remote_coh_cluster_name - scp_remote_coh_cluster_fqdn - scp_remote_coh_cluster_port - scp_remote_coh_cluster_nf_instance_id
Example	ocscp_global_egress_rl_remote_participant_connectivity_failure{applicationName="ocscp",microservice="scp-cache",ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local",scp_local_coh_cluster_name="scp-coherence-cluster",scp_remote_coh_cluster_fqdn="ocscp-scp-cache.fedsvc.svc.cluster.local",scp_remote_coh_cluster_name="scp-coherence-cluster-fedsvc",scp_remote_coh_cluster_port="30001",ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local",scp_remote_coh_cluster_nf_instance_id="2faf1bbc-6e4a-4454-a507-a14ef8e1bc22"}

Table 5-115 ocscp_mediation_http_tx_total

Field	Details
Description	Indicates that the request is forwarded to Mediation. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter

Table 5-115 (Cont.) ocscp_mediation_http_tx_total

Field	Details
Dimension	• ocscp_nf_type • ocscp_nf_service_type • ocscp_mediation_triggerpoint • ocscp_mediation_filtername
Example	ocscp_mediation_http_tx_total(ocscp_mediation_filtername="PCFsmpolicy",ocscp_mediation_triggerpoint="requestIngress")

Table 5-116 ocscp_mediation_http_rx_total

Field	Details
Description	Indicates that the response is received by SCP from Mediation when Mediation is invoked from the request trigger point and producer NF when Mediation is triggered from the response trigger points. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimension	• ocscp_nf_type • ocscp_nf_service_type • ocscp_mediation_triggerpoint • ocscp_mediation_filtername • responseCode • responseCodeClass • headerModified • bodyModified
Example	ocscp_mediation_http_rx_total(ocscp_mediation_filtername="PCFsmpolicy",ocscp_mediation_triggerpoint="requestIngress")

Table 5-117 ocscp_mediation_upstream_service_time_ms_total

Field	Details
Description	Indicates the time taken for Mediation processing.
Type	Counter
Dimension	• nfType • nfServiceType • triggerpoint • ocscp_med_upstream_service_time
Example	ocscp_mediation_upstream_service_time_ms_total(ocscp_mediation_triggerpoint="requestIngress")

Table 5-118 ocscp_mediation_http_not_reachable

Field	Details
Description	Indicates that the request to Mediation has failed or connection is not established. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Gauge
Dimension	• nfType • nfServiceType • triggerpoint • responsecode
Example	ocscp_mediation_http_not_reachable(ocscp_mediation_filter name="PCFsmpolicy",ocscp_mediation_triggerpoint="requestIngress")

Table 5-119 ocscp_notification_queue_utilization

Field	Details
Description	Indicates Notification queue utilization.
Type	Gauge
Dimension	• queueName • queueCapacity
Example	• ocscp_notification_queue_utilization{queueName="waitingQueueUDM"} • ocscp_notification_queue_utilization > 0 • ocscp_notification_queue_utilization{queueName="waitingQueueUDM"} > 0

Table 5-120 ocscp_notification_queue_alert

Field	Details
Description	Indicates Notification queue alert.
Type	Gauge
Dimension	• queueName • severity • queueCapacity
Example	• ocscp_notification_queue_alert{queueName="waitingQueueUDM"} • ocscp_notification_queue_alert > 0 • ocscp_notification_queue_alert{queueName="waitingQueueUDM"} > 0

Table 5-121 ocscp_nrfproxy_queue_utilization

Field	Details
Description	Indicates NrfProxy queue utilization.
Type	Gauge

Table 5-121 (Cont.) ocscp_nrfproxy_queue_utilization

Field	Details
Dimension	- queueName - queueCapacity
Example	- ocscp_nrfproxy_queue_utilization{queueName="scpUpstreamExecutorQueue"} - ocscp_nrfproxy_queue_utilization{queueCapacity="10000"} - ocscp_nrfproxy_queue_utilization > 0

Table 5-122 ocscp_worker_queue_utilization

Field	Details
Description	Indicates Worker queue utilization.
Type	Gauge
Dimension	- queueName - queueCapacity
Example	- ocscp_worker_queue_utilization{queueName="scpClientExecutorQueue"} - ocscp_worker_queue_utilization{queueCapacity="10000"} - ocscp_worker_queue_utilization > 0

Table 5-123 ocscp_nrfproxy_queue_alert

Field	Details
Description	Indicates NrfProxy queue alert Metrics Pegging Condition: This metric value will either be 0 or 1. Based on the severity value, it will be 1 for high severity, and correspondingly, for other severity values, it will be 0.
Type	Gauge
Dimension	- queueName: Name of the queue (scpDownstreamExecutorQueue, scpUpstreamExecutorQueue) - severity: CRITICAL, MAJOR, MINOR, NORMAL - queueCapacity: The maximum size of queue is 10000
Example	- ocscp_nrfproxy_queue_alert{queueName="scpDownstreamExecutorQueue"} - ocscp_nrfproxy_queue_alert > 0 - ocscp_nrfproxy_queue_alert{severity="CRITICAL"}

Table 5-124 ocscp_worker_queue_alert

Field	Details
Description	Indicates Worker queue alert.
Type	Gauge

Table 5-124 (Cont.) ocscp_worker_queue_alert

Field	Details
Dimension	- queueName - severity - queueCapacity
Example	- ocscp_worker_queue_alert{queueName="scpUpdateSSL CertsQueue"} - ocscp_worker_queue_alert > 0 - ocscp_worker_queue_alert{severity="MAJOR"}

Table 5-125 ocscp_metric_nf_lci_rx_total

Field	Details
Description	Indicates the number of messages received with NF's LCI header. Here, "ocscp_peer_nf_instance_id" denotes the instance ID of the producer from which it has received the LCI header.
Type	Counter
Dimension	- ocscp_fqdn - ocscp_peer_nf_instance_id
Example	- ocscp_metric_nf_lci_rx_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_nf_lci_rx_total{ocscp_peer_nf_instance_id="instance-id-example"}

Table 5-126 ocscp_metric_nf_lci_tx_total

Field	Details
Description	Indicates the number of messages forwarded with NF's LCI header. Here, "ocscp_peer_nf_instance_id" denotes the instance ID of the NF where it is sending LCI. When sending a request to the producer, it represents the producer instance ID, while when sending a response to the consumer, it represents the consumer instance ID
Type	Counter
Dimension	- ocscp_fqdn - ocscp_peer_nf_instance_id
Example	- ocscp_metric_nf_lci_tx_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_nf_lci_tx_total{ocscp_peer_nf_instance_id="instance-id-example"}

Table 5-127 ocscp_metric_scp_lci_tx_total

Field	Details
Description	Indicates the number of messages forwarded with SCP's LCI header. Here, "ocscp_peer_nf_instance_id" denotes the instance ID of the producer to which SCP sends this request with LCI header.
Type	Counter
Dimension	- ocscp_fqdn - ocscp_peer_nf_instance_id
Example	- ocscp_metric_scp_lci_tx_total{ocscp_fqdn= "ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_scp_lci_tx_total{ocscp_peer_nf_instance_id="2faf1bbc-6e4a-4454-a507-a14ef8e1bc22"}

Table 5-128 ocscp_metric_nf_lci_cache_update_total

Field	Details
Description	Indicates the number of cache updates done by peer load change.
Type	Counter
Dimension	ocscp_fqdn
Example	ocscp_metric_nf_lci_cache_update_total{ocscp_fqdn= "ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-129 ocscp_scp_load

Field	Details
Description	Metric to record SCP load value.
Type	Gauge
Dimension	- ocscp_fqdn - ocscp_scp_instance_id
Example	- ocscp_nf_load{ocscp_fqdn= "ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_nf_load{ocscp_scp_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"}

Table 5-130 ocscp_nf_load

Field	Details
Description	Metric to record NF load value.
Type	Gauge
Dimension	- ocscp_fqdn - peer_info - ocscp_scope - ocscp_producer_service_instance_id

Table 5-130 (Cont.) ocscp_nf_load

Field	Details
Example	- ocscp_nf_load{ocscp_fqdn= "ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_nf_load{ocscp_producer_instance_id="2faf1bbc-6e4a-4454-a507-a14ef8e1bc22"} - ocscp_nf_load{ocscp_scope="nf-instance"} - ocscp_nf_load{ocscp_producer_service_instance_id="fe137ab7-740a-46ee-aa5c-951806d77b02"}

Table 5-131 ocscp_modeld_cache_key_size

Field	Details
Description	Metric to indicate modelD cache Key size in worker and nrfproxy.
Type	Gauge
Dimension	- scp_service - ocscp_fqdn
Example	- ocscp_modeld_cache_key_size{scp_service="scp-nrfproxy"} - ocscp_modeld_cache_key_size{scp_service="scp-worker"}

Table 5-132 ocscp_cache_egress_grl_local_worker_token_consumption_total

Field	Details
Description	The local SCP worker token consumption in global egress rate limit.
Type	Counter
Dimension	- bucketKey - ocscp_fqdn
Example	- ocscp_cache_egress_grl_local_worker_token_consumption_total{bucketKey="egress_pcf1svc.ny-ocscp-01.svc.thrust2"} - ocscp_cache_egress_grl_local_worker_token_consumption_total{ocscp_fqdn="ocscp-scp-worker.ny-ocscp-01.svc.thrust2"}

Table 5-133 ocscp_cache_egress_grl_remote_worker_token_consumption_total

Field	Details
Description	The remote SCP worker token consumption in global egress rate limit.
Type	Counter
Dimension	- bucketKey - ocscp_fqdn

Table 5-133 (Cont.) ocscp_cache_egress_grl_remote_worker_token_consumption_total

Field	Details
Example	- ocscp_cache_egress_grl_remote_worker_token_consumption_total{bucketKey="6faf1bbc-6e4a-4a54-a507-a14ef8e1bc5d_egress_pcf1svc.ny-ocscp-01.svc.thrust2"} - ocscp_cache_egress_grl_remote_worker_token_consumption_total{ocscp_fqdn="ocscp-scp-worker.ny-ocscp-01.svc.thrust2"}

Table 5-134 ocscp_cache_egress_grl_local_worker_msg_consumption_percentage_total

Field	Details
Description	The local SCP worker token consume percentage in global egress rate limit.
Type	Counter
Dimension	- bucketKey - ocscp_fqdn
Example	- ocscp_cache_egress_grl_local_worker_msg_consumption_percentage_total{bucketKey="egress_pcf1svc.ny-ocscp-01.svc.thrust2"} - ocscp_cache_egress_grl_local_worker_msg_consumption_percentage_total{ocscp_fqdn="ocscp-scp-worker.ny-ocscp-01.svc.thrust2"}

Table 5-135 ocscp_cache_egress_grl_controller_token_consumption_total

Field	Details
Description	The number of tokens consumed by the global rate controller to adjust token availability for cache services.
Type	Counter
Dimension	- bucketKey - ocscp_fqdn
Example	- ocscp_cache_egress_grl_controller_token_consumption_total{bucketKey="egress_pcf1svc.ny-ocscp-01.svc.thrust2"} - ocscp_cache_egress_grl_controller_token_consumption_total{ocscp_fqdn="ocscp-scp-worker.ny-ocscp-01.svc.thrust2"}

Table 5-136 ocscp_metric_svc_unreachable

Field	Details
Description	This metric is pegged whenever there is a communication failure between microservices. Once the communication between the source and the destination is reestablished, this metric is cleared.
Type	Gauge

Table 5-136 (Cont.) ocscp_metric_svc_unreachable

Field	Details
Dimension	- source: indicates the source svc, such as scp-worker, scpc-configuration, and so on. - destination: indicates the destination svc, such as scpc-notification, scp-nrfproxy, and so on. Possible list of values: - source: scp-worker, scp-nrfproxy, scpc-audit, scpc-alternate-resolution, scpc-configuration, scpc-notification, scpc-subscription - destination: scp-worker, scp-nrfproxy, scpc-audit, scpc-alternate-resolution, scpc-configuration, scpc-notification, scpc-subscription, scp-mediation, scp-cache
Example	- ocscp_metric_svc_unreachable{source="scp-worker", destination="scpc-notification"} - ocscp_metric_svc_unreachable{source="scpc-audit", destination="scp-worker"} - ocscp_metric_svc_unreachable{source="scpc-subscription", destination="scp-worker"}

Table 5-137 ocscp_metric_nf_unhealthy_total

Field	Details
Description	Number of times a producer NF is marked unhealthy.
Type	Counter
Dimension	- ocscp_producer_nf_instance_id - ocscp_nf_end_point - ocscp_producer_host - ocscp_nf_service_type - ocscp_locality - ocscp_fqdn
Example	- ocscp_metric_nf_unhealthy_total{ocscp_producer_nf_instance_id="123-ffb"} - ocscp_metric_nf_unhealthy_total{ocscp_nf_end_point="10.23.34.54:8080"} - ocscp_metric_nf_unhealthy_total{ocscp_producer_host="udm1.svc.cluster.local.com"} - ocscp_metric_nf_unhealthy_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_nf_unhealthy_total{ocscp_locality="Loc7"} - ocscp_metric_nf_unhealthy_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-138 ocscp_metric_req_nf_unhealthy_total

Field	Details
Description	Number of message requests not routed to the selected producer NF because it is marked unhealthy.
Type	Counter

Table 5-138 (Cont.) ocscp_metric_req_nf_unhealthy_total

Field	Details
Dimension	- ocscp_producer_nf_instance_id - ocscp_nf_end_point - ocscp_producer_host - ocscp_nf_service_type - ocscp_locality - ocscp_fqdn
Example	- ocscp_metric_req_nf_unhealthy_total{ocscp_producer_nf_instance_id="123-44b"} - ocscp_metric_req_nf_unhealthy_total{ocscp_nf_end_point="10.23.34.54:8080"} - ocscp_metric_req_nf_unhealthy_total{ocscp_producer_host="udm1.svc.cluster.local.com"} - ocscp_metric_req_nf_unhealthy_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_req_nf_unhealthy_total{ocscp_locality="Loc7"} - ocscp_metric_req_nf_unhealthy_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-139 ocscp_metric_req_peer_scp_unhealthy_total

Field	Details
Description	Number of message requests not routed to the selected peer SCP because it is marked unhealthy.
Type	Counter
Dimension	- ocscp_nf_service_type - ocscp_peer_scp_instance_id - ocscp_peer_scp_fqdn - ocscp_fqdn - ocscp_locality
Example	- ocscp_metric_req_peer_scp_unhealthy_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_req_peer_scp_unhealthy_total{ocscp_peer_scp_instance_id="123-23b"} - ocscp_metric_req_peer_scp_unhealthy_total{ocscp_peer_scp_fqdn="scp51svc.cluster.local.com"} - ocscp_metric_req_peer_scp_unhealthy_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_req_peer_scp_unhealthy_total{ocscp_locality="Loc7"}

Table 5-140 ocscp_metric_sepp_unhealthy_total

Field	Details
Description	Number of times SEPP is marked unhealthy.
Type	Counter

Table 5-140 (Cont.) ocscp_metric_sepp_unhealthy_total

Field	Details
Dimension	- ocscp_sepp_intance_id - ocscp_sepp_end_point - ocscp_sepp_fqdn - ocscp_locality - ocscp_fqdn
Example	- ocscp_metric_sepp_unhealthy_total{ocscp_sepp_intance_id="123-55b"} - ocscp_metric_sepp_unhealthy_total{ocscp_sepp_end_point="10.23.34.54:8080"} - ocscp_metric_sepp_unhealthy_total{ocscp_sepp_fqdn="sepp1.svc.cluster.local.com"} - ocscp_metric_sepp_unhealthy_total{ocscp_locality="Loc7"} - ocscp_metric_sepp_unhealthy_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-141 ocscp_metric_req_sepp_unhealthy_total

Field	Details
Description	Number of message requests not routed to the selected SEPP because SEPP is marked unhealthy.
Type	Counter
Dimension	- ocscp_nf_service_type - ocscp_sepp_intance_id - ocscp_sepp_end_point - ocscp_sepp_fqdn - ocscp_locality - ocscp_fqdn
Example	- ocscp_metric_req_sepp_unhealthy_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_metric_req_sepp_unhealthy_total{ocscp_sepp_intance_id="123-455b"} - ocscp_metric_req_sepp_unhealthy_total{ocscp_sepp_end_point="10.23.34.54:8080"} - ocscp_metric_req_sepp_unhealthy_total{ocscp_sepp_fqdn="sepp1.svc.cluster.local.com"} - ocscp_metric_req_sepp_unhealthy_total{ocscp_locality="Loc7"} - ocscp_metric_req_sepp_unhealthy_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-142 ocscp_metric_nf_set_unhealthy

Field	Details
Description	Indicates whether all the producer NFs in the NF SET are unhealthy.
Type	Gauge
Dimension	ocscp_nf_setid

Table 5-142 (Cont.) ocscp_metric_nf_set_unhealthy

Field	Details
Example	ocscp_metric_nf_set_unhealthy{ocscp_nf_setid="123-445-363"}

Table 5-143 ocscp_sepp_unhealthy

Field	Details
Description	Indicate whether all the SEPP instances are unhealthy.
Type	Gauge
Dimension	- ocscp_sepp_instance_id - ocscp_sepp_fqdn - ocscp_locality
Example	- ocscp_sepp_unhealthy{ocscp_sepp_instance_id="123-455b"} - ocscp_sepp_unhealthy{ocscp_sepp_fqdn="sepp1.svc.cluster.local.com"} - ocscp_sepp_unhealthy{ocscp_locality="Loc7"}

Table 5-144 ocscp_metric_trafficfeed_not_attempted_total

Field	Details
Description	Number of messages selected for traffic feed based on the trigger point configuration filter criteria but not attempted to send because of some condition.
Type	Counter
Dimension	- nfServiceType (for example, nudm-uecm, nudm-sdm, and so on.) - ocscp_trafficfeed_triggerpoint (for example, requestIngress, requestEgress, responseIngress, responseEgress.) - ocscp_cause(for example, CRITICAL Pod Overload, Incorrect Data Director Information, Send Failed.) - kafka_partition_id - kafka_broker_id - ocscp_fqdn
Example	ocscp_metric_trafficfeed_not_attempted_total{ocscp_cause="Incorrect Data Director Information",ocscp_nf_service_type="nudm-uecm",ocscp_trafficfeed_triggerpoint="requestEgress"}

Table 5-145 ocscp_metric_trafficfeed_failed_total

Field	Details
Description	Number of messages failed to send to traffic feed.
Type	Counter

Table 5-145 (Cont.) ocscp_metric_trafficfeed_failed_total

Field	Details
Dimension	- ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on.) - ocscp_trafficfeed_triggerpoint (for example, requestIngress, requestEgress, responseIngress, responseEgress.) - ocscp_cause(for example, CRITICAL Pod Overload, Incorrect Data Director Information, Send Failed.) - kafka_partition_id - kafka_broker_id - ocscp_fqdn
Example	ocscp_metric_trafficfeed_failed_total{ocscp_cause="Incorrect Data Director Information",ocscp_nf_service_type="nudm-uecm",ocscp_trafficfeed_triggerpoint="requestEgress"}

Table 5-146 ocscp_metric_trafficfeed_attempted_total

Field	Details
Description	Number of messages sent to traffic feed.
Type	Counter
Dimension	- ocscp_nf_service_type (for example, nudm-uecm, nudm-sdm, and so on.) - ocscp_trafficfeed_triggerpoint (for example, requestIngress, requestEgress, responseIngress, responseEgress.) - kafka_partition_id - kafka_broker_id - ocscp_fqdn
Example	ocscp_metric_trafficfeed_total{ocscp_nf_service_type="nausf-auth",ocscp_trafficfeed_triggerpoint="requestEgress"}

Table 5-147 ocscp_worker_fqdn_resolution_total

Field	Details
Description	Number of messages for which fqdn is resolved by DNS or from nfProfile.
Type	Counter
Dimension	- ocscp_nf_service_type - ocscp_fqdn_resolution - ocscp_fqdn
Example	- ocscp_worker_fqdn_resolution_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_worker_fqdn_resolution_total{ocscp_fqdn_resolution="dns"} - ocscp_worker_fqdn_resolution_total{ocscp_fqdn="ocscp-scp-worker.ny-ocscp-01.svc.thrust2"}

Table 5-148 ocscp_worker_interplmn_fqdn_resolution_total

Field	Details
Description	Number of messages for which inter-plmn fqdn is resolved by DNS or from nfProfile.
Type	Counter
Dimension	- ocscp_nf_service_type - ocscp_fqdn_resolution - ocscp_fqdn
Example	- ocscp_worker_interplmn_fqdn_resolution_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_worker_interplmn_fqdn_resolution_total{ocscp_fqdn_resolution="dns"} - ocscp_worker_interplmn_fqdn_resolution_total{ocscp_fqdn="ocscp-scp-worker.ny-ocscp-01.svc.thrust2"}

Table 5-149 ocscp_worker_http_req_host_type_total

Field	Details
Description	Number of messages with http host as IP or FQDN. Note: When a message request is implicit or of default notification type, the value of the ocscp_worker_http_req_host_type_total metric is updated based on the host of the message request, with value IP or FQDN that is independent of the configuration of the Egress Host Preference feature. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimension	- ocscp_consumer_host - ocscp_consumer_nf_instance_id - ocscp_consumer_nf_type - ocscp_nf_service_type - host_type - ocscp_fqdn Note: These dimensions are available based on the availability of the User-Agent header in incoming message requests. SCP should be able to decode that particular format only if the User-Agent header is available in certain allowed configurable formats.
Example	- ocscp_worker_http_req_host_type_total{ocscp_nf_service_type="nudm-uecm"} - ocscp_worker_http_req_host_type_total{host_type="fqdn"} - ocscp_worker_http_req_host_type_total{ocscp_fqdn="ocscp-scp-worker.ny-ocscp-01.svc.thrust2"}

Table 5-150 ocscp_worker_cpu_pod_overload_control_status

Field	Details
Description	Indicates CPU utilization of SCP worker at any level.
Type	Gauge
Dimension	- pod_name - scp_service - ocscp_threshold_level
Example	- ocscp_worker_cpu_pod_overload_control_status{pod_name="ocscp-scp-worker-647595788d-7tbx9",scp_service="scp-worker",ocscp_threshold_level="major"} - ocscp_worker_cpu_pod_overload_control_status{pod_name="ocscp-scp-worker-647595788d-7tbx9",scp_service="scp-worker"}

Table 5-151 ocscp_worker_pod_overload_control_pendingTrans_utilization_minor

Field	Details
Description	Indicates pending transaction utilization of SCP worker at minor level.
Type	Gauge
Dimension	- pod_name - scp_service
Example	ocscp_worker_pod_overload_control_pendingTrans_utilization_minor{pod_name="ocscp-scp-worker-668c75fbc8-kptfv",scp_service="scp-worker"}

Table 5-152 ocscp_worker_pod_overload_control_pendingTrans_utilization_major

Field	Details
Description	Indicates pending transaction utilization of SCP worker at major level.
Type	Gauge
Dimension	- pod_name - scp_service
Example	ocscp_worker_pod_overload_control_pendingTrans_utilization_major{pod_name="ocscp-scp-worker-668c75fbc8-kptfv",scp_service="scp-worker"}

Table 5-153 ocscp_worker_pod_overload_control_pendingTrans_utilization_critical

Field	Details
Description	Indicates pending transaction utilization of SCP worker at critical level.
Type	Gauge
Dimension	- pod_name - scp_service

Table 5-153 (Cont.) ocscp_worker_pod_overload_control_pendingTrans_utilization_critical

Field	Details
Example	ocscp_worker_pod_overload_control_pendingTrans_utilization_critical{pod_name="ocscp-scp-worker-668c75fbc8-kptfv",scp_service="scp-worker"}

Table 5-154 ocscp_worker_pod_overload_control_pendingTrans_utilization_warn

Field	Details
Description	Indicates pending transaction utilization of SCP worker at warn level.
Type	Gauge
Dimension	- pod_name - scp_service
Example	ocscp_worker_pod_overload_control_pendingTrans_utilization_warn{pod_name="ocscp-scp-worker-668c75fbc8-kptfv",scp_service="scp-worker"}

Table 5-155 ocscp_worker_pendingTrans_pod_overload_control_status

Field	Details
Description	Indicates pending transaction utilization of SCP worker at any level.
Type	Gauge
Dimension	- pod_name - scp_service - ocscp_threshold_level
Example	- ocscp_worker_pendingTrans_pod_overload_control_status{pod_name="ocscp-scp-worker-668c75fbc8-kptfv",scp_service="scp-worker"} - ocscp_worker_pendingTrans_pod_overload_control_status{pod_name="ocscp-scp-worker-668c75fbc8-kptfv",scp_service="scp-worker",ocscp_threshold_level="major"}

Table 5-156 ocscp_worker_pod_overload_control_resource_utilization_minor

Field	Details
Description	Indicates overload control resource utilization of SCP worker at minor level.
Type	Gauge
Dimension	- pod_name - scp_service
Example	ocscp_worker_pod_overload_control_resource_utilization_minor{pod_name="ocscp-scp-worker-668c75fbc8-kptfv",scp_service="scp-worker"}

Table 5-157 ocscp_worker_pod_overload_control_resource_utilization_major

Field	Details
Description	Indicates overload control resource utilization of SCP worker at major level.
Type	Gauge
Dimension	- pod_name - scp_service
Example	ocscp_worker_pod_overload_control_resource_utilization_major{pod_name="ocscp-scp-worker-668c75fbc8-kptfv",scp_service="scp-worker"}

Table 5-158 ocscp_worker_pod_overload_control_resource_utilization_critical

Field	Details
Description	Indicates overload control resource utilization of SCP worker at critical level.
Type	Gauge
Dimension	- pod_name - scp_service
Example	ocscp_worker_pod_overload_control_resource_utilization_critical{pod_name="ocscp-scp-worker-668c75fbc8-kptfv",scp_service="scp-worker"}

Table 5-159 ocscp_worker_pod_overload_control_resource_utilization_warn

Field	Details
Description	Indicates overload control resource utilization of SCP worker at warn level.
Type	Gauge
Dimension	- pod_name - scp_service
Example	ocscp_worker_pod_overload_control_resource_utilization_warn{pod_name="ocscp-scp-worker-668c75fbc8-kptfv",scp_service="scp-worker"}

Table 5-160 ocscp_worker_pod_overload_control_status

Field	Details
Description	Indicates overload control resource utilization of SCP worker at any level.
Type	Gauge
Dimension	- pod_name - scp_service - ocscp_threshold_level

Table 5-160 (Cont.) ocscp_worker_pod_overload_control_status

Field	Details
Example	- ocscp_worker_pod_overload_control_status{pod_name="ocscp-scp-worker-668c75fbc8-kptfv",scp_service="scp-worker"} - ocscp_worker_pod_overload_control_status{pod_name="ocscp-scp-worker-668c75fbc8-kptfv",scp_service="scp-worker",ocscp_threshold_level="major"}

Table 5-161 ocscp_worker_cca_validation_performed_total

Field	Details
Description	Indicates the number of requests on which CCA validation is performed.
Type	Counter
Dimension	ocscp_fqdn
Example	ocscp_worker_cca_validation_performed_total{ocscp_fqdn="ocscp-scp-worker.ny-ocscp-01.svc.thrust2"}

Table 5-162 ocscp_worker_cca_validation_failed_total

Field	Details
Description	Indicates the number of requests that failed CCA validation.
Type	Counter
Dimension	- ocscp_fqdn - cca_consumer_nfInstanceId (not_available in case header is not present) - san_consumer_nfInstanceId - validation_type (subject/headerpresence)
Example	- ocscp_worker_cca_validation_failed_total{ocscp_fqdn="ocscp-scp-worker.ny-ocscp-01.svc.thrust2"} - ocscp_worker_cca_validation_failed_total{cca_consumer_nfInstanceId="2faf1bbc-6e4a-4454-a507-a14ef8e1bc22"} - ocscp_worker_cca_validation_failed_total{san_consumer_nfInstanceId="2faf1bbc-6e4a-4454-a507-a14ef8e1bc22"} - ocscp_worker_cca_validation_failed_total{validation_type="headerpresence"}

Table 5-163 ocscp_worker_cca_validation_feature_enabled_without_https

Field	Details
Description	Indicates the cache queue utilization. Metrics Pegging Condition: This metric value shows the current utilization of the queue.
Type	Gauge

Table 5-163 (Cont.) ocscp_worker_cca_validation_feature_enabled_without_https

Field	Details
Dimension	- queueName : name of the queue (scpCacheExecutorQueue, accessTokenAppEvExecutorQueue, accessTokenCohEvExecutorQueue) - queueCapacity : The maximum size of queue is 30000 for scpCacheExecutorQueue and for others it 10000
Example	- The following example returns the cache queue utilization metric for scpCacheExecutorQueue: - ocscp_cache_queue_utilization{queueName="scpCacheExecutorQueue"} - The following example returns all cache queues utilization metric with queueCapacity: - ocscp_cache_queue_utilization{queueCapacity="30000"} - The following example returns all cache queue utilization metrics with utilization greater than 0: - ocscp_worker_queue_utilization > 0

Table 5-164 ocscp_cache_queue_alert

Field	Details
Description	Indicates the cache queue alert. Metrics Pegging Condition: This metric value is either 0 or 1. Based on severity, the value will be 1 for that severity, and correspondingly, for other severities, the value will be 0.
Type	Gauge
Dimension	- queueName : name of the queue (scpCacheExecutorQueue, accessTokenAppEvExecutorQueue, accessTokenCohEvExecutorQueue) - queueCapacity : The maximum size of queue is 30000 for scpCacheExecutorQueue and for others it 10000 - severity: CRITICAL, MAJOR, MINOR, NORMAL
Example	- The following example returns the cache queue alert metric for scpCacheExecutorQueue; for severity one instance will be 1, and all others will be either 0 or not pegged: - ocscp_cache_queue_alert{queueName="scpUpdateSSLQueue"} - The following example returns all cache queue alert metrics with a value of 1: - ocscp_cache_queue_alert > 0 - The following example returns the cache queue alert metric for severity MAJOR: - ocscp_cache_queue_alert{severity="MAJOR"}

Table 5-165 ocscp_load_manager_queue_utilization

Field	Details
Description	Indicates the load manager queue utilization. Metrics Pegging Condition: This metric value shows the current utilisation of the queue.
Type	Gauge
Dimension	- queueName : name of the queue (sutLciCohEvExecutorQueue, sutLciAppEvExecutorQueue, peerLciCohEvExecutorQueue, peerLciAppEvExecutorQueue, loadCohEvExecutorQueue, loadAppEvExecutorQueue) - queueCapacity : 10000
Example	- The following example returns the load manager queue utilization metric for sutLciCohEvExecutorQueue: - ocscp_load_manager_queue_utilization{queueName="sutLciCohEvExecutorQueue"} - The following example returns all load manager queues utilization metric with queueCapacity: - ocscp_load_manager_queue_utilization{queueCapacity="10000"} - The following example returns all load manager queue utilization metrics with utilization greater than 0: - ocscp_load_manager_queue_utilization > 0

Table 5-166 ocscp_load_manager_queue_alert

Field	Details
Description	Indicates the load manager queue alert. Metrics Pegging Condition: This metric value is either 0 or 1. Based on severity, the value will be 1 for that severity, and correspondingly, for other severities, the value will be 0.
Type	Gauge
Dimension	- queueName : name of the queue (sutLciCohEvExecutorQueue, sutLciAppEvExecutorQueue, peerLciCohEvExecutorQueue, peerLciAppEvExecutorQueue, loadCohEvExecutorQueue, loadAppEvExecutorQueue) - queueCapacity : 10000 - severity: CRITICAL, MAJOR, MINOR, NORMAL

Table 5-166 (Cont.) ocscp_load_manager_queue_alert

Field	Details
Example	- The following example returns the load manager queue alert metric for sutLciCohEvExecutorQueue; for severity one instance will be 1, and all others will be either 0 or not pegged: - ocscp_load_manager_queue_alert{queueName="sutLciCohEvExecutorQueue"} - The following example returns all load manager queue alert metrics with a value of 1: - ocscp_load_manager_queue_alert> 0 - The following example returns the load manager queue alert metric for severity MAJOR: - ocscp_load_manager_queue_alert{severity="MAJOR"}

Table 5-167 ocscp_global_egress_rl_remote_participant_is_duplicate

Field	Details
Description	This metric keeps track of duplicate remote coherence participants.
Type	Gauge
Dimension	- ocscp_fqdn - scp_nf_instance_id - scp_local_coh_cluster_name - scp_remote_coh_cluster_name - scp_remote_coh_cluster_nf_instance_id - scp_remote_coh_cluster_fqdn - scp_remote_coh_cluster_port
Example	ocscp_global_egress_rl_remote_participant_is_duplicate{app="ocscp-scp-cache", app_kubernetes_io_component="microService", app_kubernetes_io_instance="ocscp", app_kubernetes_io_managed_by="Helm", app_kubernetes_io_name="scp-cache", app_kubernetes_io_part_of="ocscp", app_kubernetes_io_version="<SCP release version>", coherenceCluster="scp-coherence-cluster", helm_sh_chart="scp-cache-24.2.5-beta.22", instance="192.168.140.194:8091", job="kubernetes-pods", microservice="scp-cache", namespace="scpsvc", pod="ocscp-scp-cache-dc9b4775b-vhmtx", pod_template_hash="dc9b4775b", ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local", scp_local_coh_cluster_name="scp-coherence-cluster", scp_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e", scp_remote_coh_cluster_fqdn="ocats-scp-ocats-scpglbratelimitstub.scpsvc.svc.cluster.local", scp_remote_coh_cluster_name="scpstub-coherence-cluster", scp_remote_coh_cluster_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e", scp_remote_coh_cluster_port="30001", vendor="oracle"}

Table 5-168 ocscp_worker_health_check_rx_req_total

Field	Details
Description	Number of health check requests received by SCP
Type	Counter
Dimension	- pod_name - ocscp_fqdn - ocscp_nf_instance_id - ocscp_consumer_host - ocscp_consumer_nf_instance_id - ocscp_consumer_nf_type - ocscp_consumer_nf_unique_id
Example	- ocscp_worker_health_check_rx_req_total{pod_name="ocscp-scp-worker-6dbf7ff8b6-g284z"} - ocscp_worker_health_check_rx_req_total{ocscp_fqdn="ocscp-scp-worker.ny-ocscp-01.svc.thrust2"} - ocscp_worker_health_check_rx_req_total{ocscp_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_worker_health_check_rx_req_total{ocscp_consumer_host="smf1.scpsvc.svc.cluster.local"} - ocscp_worker_health_check_rx_req_total{ocscp_consumer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_worker_health_check_rx_req_total{ocscp_consumer_nf_type="SMF"} - ocscp_worker_health_check_rx_req_total{ocscp_consumer_nf_unique_id="SE1SMSF1"}

Table 5-169 ocscp_worker_health_check_success_res_total

Field	Details
Description	Number of success responses sent by SCP in response to the health check status query
Type	Counter
Dimension	- pod_name - ocscp_fqdn - ocscp_nf_instance_id - ocscp_consumer_host - ocscp_consumer_nf_instance_id - ocscp_consumer_nf_type - ocscp_consumer_nf_unique_id - successRspType

Table 5-169 (Cont.) ocscp_worker_health_check_success_res_total

Field	Details
Example	- ocscp_worker_health_check_success_res_total{pod_name="ocscp-scp-worker-6dbf7ff8b6-g284z"} - ocscp_worker_health_check_success_res_total{ocscp_fqdn="ocscp-scp-worker.ny-ocscp-01.svc.thrust2"} - ocscp_worker_health_check_success_res_total{ocscp_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_worker_health_check_success_res_total{ocscp_consumer_host="smf1.scpsvc.svc.cluster.local"} - ocscp_worker_health_check_success_res_total{ocscp_consumer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_worker_health_check_success_res_total{ocscp_consumer_nf_type="SMF"} - ocscp_worker_health_check_success_res_total{ocscp_consumer_nf_unique_id="SE1SMSF1"} - ocscp_worker_health_check_success_res_total{successRspType="200"}

Table 5-170 ocscp_worker_health_check_error_res_total

Field	Details
Description	Number of error responses sent by SCP in response to the health check status query
Type	Counter
Dimension	- pod_name - ocscp_fqdn - ocscp_nf_instance_id - ocscp_consumer_host - ocscp_consumer_nf_instance_id - ocscp_consumer_nf_type - ocscp_consumer_nf_unique_id - status - cause - detail - title

Table 5-170 (Cont.) ocscp_worker_health_check_error_res_total

Field	Details
Example	- ocscp_worker_health_check_error_res_total{pod_name="ocscp-scp-worker-6dbf7ff8b6-g284z"} - ocscp_worker_health_check_error_res_total{ocscp_fqdn="ocscp-scp-worker.ny-ocscp-01.svc.thrust2"} - ocscp_worker_health_check_error_res_total{ocscp_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_worker_health_check_error_res_total{ocscp_consumer_host="smf1.scpsvc.svc.cluster.local"} - ocscp_worker_health_check_error_res_total{ocscp_consumer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_worker_health_check_error_res_total{ocscp_consumer_nf_type="SMF"} - ocscp_worker_health_check_error_res_total{cause="NF_CONGESTION_RISK"} - ocscp_worker_health_check_error_res_total{detail="NF service is overloaded/congested"} - ocscp_worker_health_check_error_res_total{status="503"} - ocscp_worker_health_check_error_res_total{title="NF service is overloaded/congested"}

Table 5-171 ocscp_interscp_health_check_status_success_total

Field	Details
Description	Number of health check success status of peer SCP based on Outlier Detection and Health Check features
Type	Counter
Dimension	- interScpHost - pod_name - ocscp_nf_instance_id - ocscp_fqdn
Example	- ocscp_interscp_health_check_status_success_total{interScpHost="scp51svc.scpsvc.svc.cluster.local"} - ocscp_interscp_health_check_status_success_total{ocscp_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_interscp_health_check_status_success_total{pod_name="ocscp-scp-worker-6dbf7ff8b6-g284z"} - ocscp_interscp_health_check_status_success_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-172 ocscp_interscp_health_check_status_failure_total

Field	Details
Description	Number of health check failure status of peer SCP based on Outlier Detection and Health check features
Type	Counter

Table 5-172 (Cont.) ocscp_interscp_health_check_status_failure_total

Field	Details
Dimension	• interScpHost • pod_name • ocscp_nf_instance_id • ocscp_fqdn
Example	• ocscp_interscp_health_check_status_failure_total{interScpHost="scp51svc.scpsvc.svc.cluster.local"} • ocscp_interscp_health_check_status_failure_total{ocscp_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} • ocscp_interscp_health_check_status_failure_total{pod_name="ocscp-scp-worker-6dbf7ff8b6-g284z"} • ocscp_interscp_health_check_status_failure_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-173 ocscp_health_check_status_failed

Field	Details
Description	SCP health check status. The value can be either 0 or 1. 0 indicates SCP health check status is successful. 1 indicates SCP health check status has failed.
Type	Gauge
Dimension	• pod_name • ocscp_fqdn • ocscp_nf_instance_id • ocscp_consumer_host • ocscp_consumer_nf_instance_id • ocscp_consumer_nf_type • ocscp_consumer_nf_unique_id • status • cause • detail • title

Table 5-173 (Cont.) ocscp_health_check_status_failed

Field	Details
Example	- ocscp_health_check_status_failed{pod_name="ocscp-scp-worker-6dbf7ff8b6-g284z"} - ocscp_health_check_status_failed{ocscp_fqdn="ocscp-scp-worker.ny-ocscp-01.svc.thrust2"} - ocscp_health_check_status_failed{ocscp_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_health_check_status_failed{ocscp_consumer_host="smf1.scpsvc.svc.cluster.local"} - ocscp_health_check_status_failed{ocscp_consumer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_health_check_status_failed{ocscp_consumer_nf_type="SMF"} - ocscp_health_check_status_failed{cause="NF_CONGESTION_RISK"} - ocscp_health_check_status_failed{detail="NF service is overloaded/congested"} - ocscp_health_check_status_failed{status="503"} - ocscp_health_check_status_failed{title="NF service is overloaded/congested"}

Table 5-174 ocscp_interscp_health_check_status_failed

Field	Details
Description	Inter-SCP health check status based on Outlier Detection and Health Check features The value can be either 0 or 1. 0 indicates inter-SCP health check status is successful. 1 indicates inter-SCP health check status has failed.
Type	Gauge
Dimension	- interScpHost - pod_name - ocscp_nf_instance_id - ocscp_fqdn
Example	- ocscp_interscp_health_check_status_failure_total{interScpHost="scp51svc.scpsvc.svc.cluster.local"} - ocscp_interscp_health_check_status_failed{ocscp_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_interscp_health_check_status_failed{pod_name="ocscp-scp-worker-6dbf7ff8b6-g284z"} - ocscp_interscp_health_check_status_failed{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-175 ocscp_oauth_access_token_cache_match_total

Field	Details
Description	Number of times requests fetched by access token from cache in SCP-nrfProxy-oauth
Type	Counter

Table 5-175 (Cont.) ocscp_oauth_access_token_cache_match_total

Field	Details
Dimension	ocscp_fqdn
Example	-

Table 5-176 ocscp_worker_access_token_forwarded_in_tx_res_total

Field	Details
Description	Number of responses in the 3gpp-Sbi-Access-Token header
Type	Counter
Dimension	ocscp_fqdn
Example	-

Table 5-177 ocscp_worker_access_token_forwarded_in_tx_req_total

Field	Details
Description	Number of requests in the 3gpp-Sbi-Access-Token header
Type	Counter
Dimension	ocscp_fqdn
Example	-

Table 5-178 ocscp_worker_access_token_cache_match_total

Field	Details
Description	Number of times requests fetched access token from cache in SCP-Worker
Type	Counter
Dimension	ocscp_fqdn
Example	-

Table 5-179 ocscp_nrf_oauth_error_counts_total

Field	Details
Description	Error responses received from NRF
Type	Counter
Dimension	- ocscp_sub - ocscp_aud
Example	-

Table 5-180 ocscp_oauth_access_token_tx_rsp_total

Field	Details
Description	Responses from SCP-nrfProxy-oauth to SCP-Worker.
Type	Counter

Table 5-180 (Cont.) ocscp_oauth_access_token_tx_rsp_total

Field	Details
Dimension	- ocscp_fqdn - ocscp_response_code - requester_nfType - target_nfType
Example	-

Table 5-181 ocscp_oauth_access_token_rx_res_total

Field	Details
Description	Responses from NRF to SCP-nrfProxy-oauth through SCP-Worker.
Type	Counter
Dimension	- ocscp_fqdn - ocscp_response_code - requester_nfType - target_nfType
Example	-

Table 5-182 ocscp_oauth_access_token_tx_req_total

Field	Details
Description	Requests generated by SCP-nrfProxy-oauth toward NRF through SCP-Worker.
Type	Counter
Dimension	- ocscp_fqdn - requester_nfType - target_nfType
Example	-

Table 5-183 ocscp_oauth_access_token_rx_req_total

Field	Details
Description	Requests received in the SCP-nrfProxy-oauth from SCP-Worker.
Type	Counter
Dimension	ocscp_fqdn
Example	-

Table 5-184 ocscp_worker_access_token_rx_res_total

Field	Details
Description	Access token response from SCP-nrfProxy-oauth to SCP-Worker.
Type	Counter

Table 5-184 (Cont.) ocscp_worker_access_token_rx_res_total

Field	Details
Dimension	- ocscp_fqdn - ocscp_nf_type - ocscp_nf_service_type - ocscp_consumer_nf_type - ocscp_response_code - ocscp_response_code_class
Example	ocscp_worker_access_token_rx_res_total{ocscp_consumer_nf_type="AMF",ocscp_nf_service_type="nsmf-pdusession",ocscp_nf_type="SMF",ocscp_response_code="200",ocscp_response_code_class="2",ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-185 ocscp_worker_access_token_tx_req_total

Field	Details
Description	Access token request from SCP-Worker to SCP-nrfProxy-oauth.
Type	Counter
Dimension	- ocscp_fqdn - ocscp_nf_type - ocscp_nf_service_type - ocscp_consumer_nf_type
Example	ocscp_worker_access_token_tx_req_total{ocscp_consumer_nf_type="AMF",ocscp_nf_service_type="nsmf-pdusession",ocscp_nf_type="SMF",ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-186 ocscp_audit_interscp_health_check_tx_req_total

Field	Details
Description	Number of inter-SCP health check requests triggered by SCPC-Audit to SCP-Worker
Type	Counter
Dimension	- interScpHost - ocscp_fqdn
Example	- ocscp_audit_interscp_health_check_tx_req_total{interScpHost="scp51svc.scpsvc.svc.cluster.local"} - ocscp_audit_interscp_health_check_tx_req_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-187 ocscp_audit_interscp_health_check_rx_res_total

Field	Details
Description	Number of inter-SCP health check responses received by SCPC-Audit from SCP-Worker
Type	Counter

Table 5-187 (Cont.) ocscp_audit_interscp_health_check_rx_res_total

Field	Details
Dimension	- interScpHost - ocscp_fqdn - responseCode
Example	- ocscp_audit_interscp_health_check_rx_res_total{interScpHost="scp51svc.scpsvc.svc.cluster.local"} - ocscp_audit_interscp_health_check_rx_res_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_audit_interscp_health_check_rx_res_total{responseCode="200"}

Table 5-188 ocscp_notification_lock_operation_time

Field	Details
Description	Indicates the time taken to acquire and release the lock
Type	Counter
Dimension	- nfInstanceId: NfInstanceId of the notification for which the lock operation is performed - lockEvent: lock event (Released or Acquire) - timeInMs: time taken by the lock event
Example	- ocscp_notification_lock_operation_time{namespace="ocscp"} - ocscp_notification_lock_operation_time{lockEvent="ACQUIRED_LOCK"} - ocscp_notification_lock_operation_time{lockEvent="RELEASED_LOCK"}

Table 5-189 ocscp_ignored_processed_nf_notification_total

Field	Details
Description	Indicates the notifications that are not processed (ignored) because another notification with the same nfInstanceId is processed by another thread.
Type	Counter
Dimension	- nfType: NfType of the notification - nfInstanceId: NfInstanceId of the notification
Example	- ocscp_ignored_processed_nf_notification_total{namespace="ocscp" } - ocscp_ignored_processed_nf_notification_total{nfType="AUSF" }

Table 5-190 ocscp_notification_queue_waiting_time

Field	Details
Description	Indicates the time taken by the notification in the waiting queue.
Type	Counter

Table 5-190 (Cont.) ocscp_notification_queue_waiting_time

Field	Details
Dimension	• nfType: NfType of the notification • nfInstanceId: NfInstanceId of the notification • timeInMs: time taken by the notification in queue.
Example	ocscp_notification_queue_waiting_time {namespace="ocscp"}

Table 5-191 ocscp_notification_profile_processing_time

Field	Details
Description	Indicates the total time taken to process notifications.
Type	Counter
Dimension	• nfType: NfType of the notification • nfInstanceId: NfInstanceId of the notification • nfProfileProcessingStatus: Status of the notification processing • processingTimeInMs: Total Time taken for notification processing
Example	ocscp_notification_profile_processing_time {nfType = "AUSF"}

Table 5-192 ocscp_metric_5gsbi_upstream_destination_rejection_total

Field	Details
Description	Tracks the total number of upstream destinations rejected by SCP due to certain unhealthy conditions of the host, such as CircuitBreaker, Producer Congestion Control, etc.
Type	Counter
Dimension	• ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) • ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) • ocscp_nf_end_point (Default value = NA) • ocscp_producer_host (default value = NA, Contains fqdn of producer) • ocscp_authority • ocscp_cause
Example	• ocscp_metric_5gsbi_upstream_destination_rejection_total{ocscp_nf_type="PCF"} • ocscp_metric_5gsbi_upstream_destination_rejection_total{ocscp_nf_service_type="npcf-smpolicycontrol"} • ocscp_metric_5gsbi_upstream_destination_rejection_total{ocscp_cause="CircuitBreaking"} • ocscp_metric_5gsbi_upstream_destination_rejection_total{ocscp_nf_end_point="NA"} • ocscp_metric_5gsbi_upstream_destination_rejection_total{ocscp_authority="ocscp-scp-worker.scpsvc.svc.cluster.local:9443"}

Table 5-193 ocscp_metric_5gsbi_rx_res_failure_total

Field	Details
Description	Tracks the total number of upstream response failures at SCP with a specific cause, such as ConnectionTimeout and UnknownHostException so on.
Type	Counter
Dimension	- ocscp_nf_type (e.g., UDM, PCF, AMF, etc.) - ocscp_nf_service_type (e.g., nudm-uecm, nudm-sdm, etc.) - ocscp_nf_end_point (Default value = NA) - ocscp_response_code(e.g., 201, 503, 404, etc.) - ocscp_response_code_class (e.g., 2, 5, etc.) - ocscp_authority - scheme(e.g., https) - ocscp_cause
Example	- ocscp_metric_5gsbi_rx_res_failure_total{ocscp_nf_type="PCF"} - ocscp_metric_5gsbi_rx_res_failure_total{ocscp_nf_service_type="npcf-smpolicycontrol"} - ocscp_metric_5gsbi_rx_res_failure_total{scheme="https"} - ocscp_metric_5gsbi_rx_res_failure_total{ocscp_nf_end_point="NA"} - ocscp_metric_5gsbi_rx_res_failure_total{ocscp_authority="ocscp-scp-worker.scpsvc.svc.cluster.local:9443"} - ocscp_metric_5gsbi_rx_res_failure_total{ocscp_cause="UnknownHostException"}

Table 5-194 ocscp_configuration_dnssrv_nrf_non_migration_task_failure_count_total

Field	Details
Description	Indicates the number of failed nonmigrated tasks of the DNS SRV-based NRF Info feature. The nonmigrated tasks are as follows: - Capacity or priority update of target FQDN - New NRF SRV FQDN addition or deletion - NF Set update
Type	Counter
Dimensions	- nrfSrvFqdn - failureReason
Examples	- ocscp_configuration_dnssrv_nrf_non_migration_task_failure_count={nrfSrvFqdn="nrf1svc.scpsvc.svc.cluster.local"} - ocscp_configuration_dnssrv_nrf_non_migration_task_failure_count={failureReason="START_NRF_RULE_PROCESSING"}

Table 5-195 ocscp_configuration_dnssrv_nrf_non_migration_task_success_count_total

Field	Details
Description	Indicates the number of successful nonmigrated tasks. The nonmigrated tasks are as follows: - Capacity or priority update of the target FQDN - New NRF SRV FQDN addition or deletion - NF Set update
Type	Counter
Dimensions	nrfSrvFqdn
Examples	ocscp_configuration_dnssrv_nrf_non_migration_task_success_count_total={nrfSrvFqdn="nrf1svc.scpsvc.svc.cluster.local"}

Table 5-196 ocscp_configuration_dnssrv_nrf_non_migration_task_failure

Field	Details
Description	Indicates the status of the nonmigrated task of the DNS SRV based NRF Info feature. The status of nonmigrated tasks will be either 0 or 1. The nonmigrated tasks are as follows: - Capacity or priority update of target FQDN - New NRF SRV FQDN addition or deletion - NF Set update
Type	Gauge
Dimensions	- nrfSrvFqdn - failureReason - taskId - taskName - taskState - eventName - msName
Examples	- ocscp_configuration_dnssrv_nrf_non_migration_task_failure={nrfSrvFqdn="nrf1svc.scpsvc.svc.cluster.local"} - ocscp_configuration_dnssrv_nrf_non_migration_task_failure={failureReason="RETRY_EVENT_COUNT_EXPIRED"} - ocscp_configuration_dnssrv_nrf_non_migration_task_failure={taskId=1234} - ocscp_configuration_dnssrv_nrf_non_migration_task_failure={taskName="NRF_SET_DELETED_TASK"} - ocscp_configuration_dnssrv_nrf_non_migration_task_failure={eventName="START_NRF_RULE_PROCESSING"} - ocscp_configuration_dnssrv_nrf_non_migration_task_failure={msName="scpc-notification"} - ocscp_configuration_dnssrv_nrf_non_migration_task_failure={taskState="UNSUB_DEREG_COMPLETED"}

Table 5-197 ocscp_configuration_dnssrv_nrf_migration_task_failure

Field	Details
Description	Indicates the status of the migrated task of the DNS SRV based NRF Info feature. The status of migrated tasks will be either 0 or 1. The migrated tasks are as follows: - Static(custom.values.yaml file) to DNS SRV based NRF migration - DNS SRV based NRF to static(custom.values.yaml) migration If there is any failure in the following migration task, then the value will become 1, otherwise the value will be 0.
Type	Gauge
Dimensions	- failureReason - taskId - taskName - taskState - eventName - msName
Examples	- ocscp_configuration_dnssrv_nrf_migration_task_failure={failureReason=WAIT_TIME_FOR_COMPLETION_RESP_ELAPSED, RETRY_EVENT_COUNT_EXPIRED} - ocscp_configuration_dnssrv_nrf_migration_task_failure={taskId=1234} - ocscp_configuration_dnssrv_nrf_migration_task_failure={taskName="MIGRATION_OF_NRF_INFO_FROM_STATIC_TO_DNS"} - ocscp_configuration_dnssrv_nrf_migration_task_failure={eventName="FEATURE_ENABLED"} - ocscp_configuration_dnssrv_nrf_migration_task_failure={msName="scpc-configuration"} - ocscp_configuration_dnssrv_nrf_migration_task_failure={taskState="WAITING_FOR_DNS_SRV_DATA"}

Table 5-198 ocscp_configuration_dnssrv_nrf_duplicate_target_detected

Field	Details
Description	Indicates when duplicate target FQDNs are detected across different NRF SRV FQDNs.
Type	Gauge
Dimensions	- nrfSrvFqdn - ocscp_fqdn
Examples	- ocscp_configuration_dnssrv_nrf_duplicate_target_detected={nrfSrvFqdn="nrf1svc.scpsvc.svc.cluster.local"} - ocscp_configuration_dnssrv_nrf_duplicate_target_detected={ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-199 ocscp_subscription_nrf_deregistration_success_total

Field	Details
Description	Indicates the number of successful deregistrations with NRF.
Type	Counter
Dimensions	nrfRegionOrSetId
Examples	ocscp_subscription_nrf_deregistration_success_total={nrfRegionOrSetId="setnrf11.nrfset.5gc.mnc012.mcc345"}

Table 5-200 ocscp_subscription_nrf_deregistration_failure_total

Field	Details
Description	Indicates the number of failed deregistrations with NRF.
Type	Counter
Dimensions	nrfRegionOrSetId
Examples	ocscp_subscription_nrf_deregistration_failure_total={nrfRegionOrSetId="setnrf11.nrfset.5gc.mnc012.mcc345"}

Table 5-201 ocscp_worker_peer_oci_conveyance_total

Field	Details
Description	Indicates the number of messages routed by SCP with peer NF's Overload Control Information (OCI) header.
Type	Counter
Dimensions	- ocscp_peer_nf_instance_id - ocscp_peer_nf_service_type - ocscp_fqdn
Examples	- ocscp_worker_peer_oci_conveyance_total{ocscp_peer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_worker_peer_oci_conveyance_total{ocscp_peer_nf_service_type="npcf-smpolicycontrol"} - ocscp_worker_peer_oci_conveyance_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-202 ocscp_worker_oci_conveyance_total

Field	Details
Description	Indicates the number of messages routed by SCP with SCP's OCI header.
Type	Counter
Dimensions	- ocscp_peer_nf_instance_id - ocscp_fqdn
Examples	- ocscp_worker_oci_conveyance_total{ocscp_peer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_worker_oci_conveyance_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-203 ocscp_worker_oci_enforced_total

Field	Details
Description	Indicates the number of messages on which OCI is enforced causing those messages to be responded with error or alternate routed.
Type	Counter
Dimensions	- ocscp_producer_nf_instance_id - ocscp_nf_service_type - action - error_sent - rerouted - ocscp_smp_priority - ocscp_fqdn
Examples	- ocscp_worker_oci_enforced_total{ocscp_producer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_worker_oci_enforced_total{ocscp_nf_service_type="npcf-smpolicycontrol"} - ocscp_worker_oci_enforced_total{action="rerouted"} - ocscp_worker_oci_enforced_total{ocscp_smp_priority="10"} - ocscp_worker_oci_enforced_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-204 ocscp_worker_rx_unsupported_oci_scope_total

Field	Details
Description	Indicates the number of messages received with unsupported OCI scope from peer.
Type	Counter
Dimensions	- scope - ocscp_fqdn
Examples	- ocscp_worker_rx_unsupported_oci_scope_total{scope="S-NSSAI: {"sst": 1, "sd": "A08923"}; DNN: internet.mnc012.mcc345.gprs"} - ocscp_worker_rx_unsupported_oci_scope_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-205 ocscp_metric_nf_oci_rx_total

Field	Details
Description	Indicates the number of messages received with OCI.
Type	Counter
Dimensions	- ocscp_peer_nf_instance_id - ocscp_fqdn
Examples	- ocscp_metric_nf_oci_rx_total{ocscp_peer_nf_instance_id="6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e"} - ocscp_metric_nf_oci_rx_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"}

Table 5-206 ocscp_metric_trafficfeed_partition_unhealthy

Field	Details
Description	No leader is available for Kafka partition.
Type	Gauge
Dimensions	- kafka_partition_id - topic - cause - no_leader - unreachable
Examples	ocscp_metric_trafficfeed_partition_unhealthy{ocscp_cause="no_leader",ocscp_nf_service_type="nudm-uecm",ocscp_trafficfeed_triggerpoint="requestEgress",kafka_partition_id="NA",kafka_broker_id="NA"}

Table 5-207 ocscp_metric_trafficfeed_success_total

Field	Details
Description	Number of messages sent to traffic feed. This metric is unavailable when "acks" is set to "0". To make it available, change "acks" to "1" or "all".
Type	Counter
Dimensions	- nfServiceType (e.g. nudm-uecm, nudm-sdm, etc.) - triggerpoint (e.g. requestIngress, requestEgress, responseIngress, responseEgress) - kafka_partition_id - kafka_broker_id
Examples	ocscp_metric_trafficfeed_success_total{ocscp_nf_service_type="nausf-auth",ocscp_trafficfeed_triggerpoint="requestEgress",kafka_partition_id="NA",kafka_broker_id="NA"}

Table 5-208 ocscp_metric_trafficfeed_cluster_unhealthy

Field	Details
Description	The whole traffic feed Kafka cluster is down.
Type	Gauge
Dimensions	cause (unreachable)
Examples	ocscp_metric_trafficfeed_cluster_unhealthy{ocscp_cause=unreachable}

Table 5-209 ocscp_ignore_unknown_service_total

Field	Details
Description	The total number of unknown services ignored in the NF profile.
Type	Counter
Dimensions	- nfInstanceId - nfType - serviceName

Table 5-209 (Cont.) ocscp_ignore_unknown_service_total

Field	Details
Examples	- ocscp_ignore_unknown_service_total{nflInstanceId=6faf1bbc-6e4a-4454-a507-a14ef8e1bc5e} - ocscp_ignore_unknown_service_total{nfType=UDM} - ocscp_ignore_unknown_service_total{serviceName=nudm_unknown}

Table 5-210 ocscp_worker_ondemand_discovery_response_cache_skipped_total

Field	Details
Description	The total number of discovery responses from NRF to SCP-Worker that are skipped from being cached due to the configured rule (the configured rule is specified by the RuleName dimension).
Type	Counter
Dimensions	- rule_name - ocscp_nf_type - ocscp_fqdn
Examples	- ocscp_worker_ondemand_discovery_response_cache_skipped_total{applicationName="ocscp"} - ocscp_worker_ondemand_discovery_response_cache_skipped_total{microservice="scp-worker"} - ocscp_worker_ondemand_discovery_response_cache_skipped_total{ocscp_nf_type="PCF"} - ocscp_worker_ondemand_discovery_response_cache_skipped_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_worker_ondemand_discovery_response_cache_skipped_total{rule_name="rule2"}

Table 5-211 ocscp_worker_ondemand_discovery_response_cache_added_total

Field	Details
Description	The total number of discovery responses from NRF to SCP-Worker that are cached.
Type	Counter
Dimensions	- ocscp_nf_type - ocscp_fqdn
Examples	ocscp_worker_ondemand_discovery_response_cache_added_total{applicationName="ocscp",microservice="scp-worker",ocscp_nf_type="PCF",ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local",vendor="Oracle",}

Table 5-212 ocscp_worker_ondemand_discovery_response_cache_removed_total

Field	Details
Description	The total number of discovery responses from NRF to SCP-Worker that were removed from the cache either due to expiry time or queue overflow.
Type	Counter

Table 5-212 (Cont.) ocscp_worker_ondemand_discovery_response_cache_removed_total

Field	Details
Dimensions	- ocscp_nf_type - ocscp_fqdn
Examples	ocscp_worker_ondemand_discovery_response_cache_removed_total{applicationName="ocscp",microservice="scp-worker",ocscp_nf_type="PCF",ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local",vendor="Oracle",}

Table 5-213 ocscp_nrfproxy_ondemand_discovery_response_cache_skipped_total

Field	Details
Description	The total number of discovery responses from NRF to nrf-proxy that are skipped from being cached due to the configured rule (the configured rule is specified by the RuleName dimension).
Type	Counter
Dimensions	- rule_name - ocscp_producer_nf_type
Examples	ocscp_nrfproxy_ondemand_discovery_response_cache_skipped_total{microservice="scp-nrfproxy",ocscp_producer_nf_type="PCF",rule_name="rule1",}

Table 5-214 ocscp_nrfproxy_ondemand_discovery_response_cache_added_total

Field	Details
Description	The total number of discovery responses from NRF to nrf-proxy that are cached.
Type	Counter
Dimensions	ocscp_producer_nf_type
Examples	ocscp_nrfproxy_ondemand_discovery_response_cache_added_total{microservice="scp-nrfproxy",ocscp_producer_nf_type="PCF"}

Table 5-215 ocscp_nrfproxy_ondemand_discovery_response_cache_removed_total

Field	Details
Description	The total number of discovery responses from NRF to nrf-proxy that were removed from the cache either due to expiry time or queue overflow.
Type	Counter
Dimensions	ocscp_producer_nf_type
Examples	ocscp_nrfproxy_ondemand_discovery_response_cache_removed_total{microservice="scp-nrfproxy",ocscp_producer_nf_type="PCF"}

Table 5-216 ocscp_worker_problemdetails_details_maxlengthexceeded_total

Field	Details
Description	Number of messages for which the ProblemDetails.detail parameter in the error response exceeded the configured value for errorDetailMaxSize in SCP Features REST API for enhanced_error_rsp configuration.
Type	Counter

Table 5-216 (Cont.) ocscp_worker_problemdetails_details_maxlengthexceeded_total

Field	Details
Dimensions	ocscp_attempted_producers
Examples	ocscp_worker_problemdetails_details_maxlengthexceeded_total{applicationName="ocscp",microservice="scp-worker",ocscp_attempted_producers="[udm1svc.scpsvc.svc.cluster.local,udm2svc.scpsvc.svc.cluster.local]",ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local",vendor="Oracle"}

Table 5-217 ocscp_worker_object_pool_in_use_objects

Field	Details
Description	The total number of pool objects currently in use.
Type	Gauge
Dimensions	ocscp_object_pool_name
Examples	ocscp_worker_object_pool_in_use_objects{ocscp_object_pool_name=STREAM_INFO_OBJECT_POOL}

Table 5-218 ocscp_worker_object_pool_ready_to_use_objects

Field	Details
Description	The total number of pool objects present in the pool is ready to be used.
Type	Gauge
Dimensions	ocscp_object_pool_name
Examples	ocscp_worker_object_pool_ready_to_use_objects{ocscp_object_pool_name=STREAM_INFO_OBJECT_POOL}

Table 5-219 ocscp_worker_object_pool_created_objects

Field	Details
Description	The total number of pool objects created.
Type	Gauge
Dimensions	ocscp_object_pool_name
Examples	ocscp_worker_object_pool_created_objects{ocscp_object_pool_name=STREAM_INFO_OBJECT_POOL}

Table 5-220 ocscp_metric_http_egress_res_total

Field	Details
Description	The total number of egress 5G SBI responses from SCP, including 5G SBI service responses, notification responses, and SCP health check response messages. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimensions	NA

Table 5-220 (Cont.) ocscp_metric_http_egress_res_total

Field	Details
Examples	ocscp_metric_http_egress_res_total{applicationName="ocscp",microservice="scp-worker",ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local",vendor="Oracle"}

Table 5-221 ocscp_metric_http_ingress_req_total

Field	Details
Description	The total number of ingress 5G SBI requests at SCP includes 5G SBI service requests, notification requests, and SCP health check request messages. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimensions	NA
Examples	ocscp_metric_http_ingress_req_total{applicationName="ocscp",microservice="scp-worker",ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local",vendor="Oracle"}

Table 5-222 ocscp_worker_object_pool_denied_total

Field	Details
Description	The total count of pool objects denied once MAX_POOL_SIZE is reached.
Type	Counter
Dimensions	NA
Examples	ocscp_worker_object_pool_denied_total{ocscp_object_pool_name=STREAM_INFO_OBJECT_POOL}

Table 5-223 ocscp_worker_object_pool_tran_timeout_objects_total

Field	Details
Description	The total counts of pool objects getting total transaction timeouts during different async transaction states in the callflow.
Type	Counter
Dimensions	- ocscp_object_pool_name - ocscp_worker_client_name - ocscp_object_trans_state
Examples	- ocscp_worker_object_pool_tran_timeout_objects_total{ocscp_object_pool_name=STREAM_INFO_OBJECT_POOL} - ocscp_worker_object_pool_tran_timeout_objects_total{ocscp_worker_client_name=NRF_PROXY_CLIENT} - ocscp_worker_object_pool_tran_timeout_objects_total{ocscp_object_trans_state=ON_PROGRESS}

Table 5-224 ocscp_worker_object_pool_clash_in_callback_total

Field	Details
Description	The total counts of pool objects getting clashed due to an unexpected async flow sequence conflict or usage time ambiguity.

Table 5-224 (Cont.) ocscp_worker_object_pool_clash_in_callback_total

Field	Details
Type	Counter
Dimensions	- ocscp_object_pool_name - ocscp_worker_client_name - clash_type
Examples	- ocscp_worker_object_pool_clash_in_callback_total{ocscp_object_pool_name=STREAM_INFO_OBJECT_POOL} - ocscp_worker_object_pool_clash_in_callback_total{ocscp_worker_client_name=NRF_PROXY_CLIENT} - ocscp_worker_object_pool_clash_in_callback_total{clash_type = BY_SEQ}

Table 5-225 ocscp_worker_object_pool_borrow_conflict_total

Field	Details
Description	The total counts of pool objects that are attempted to be reused when they are still under previous transactions, such as duplicate usage.
Type	Counter
Dimensions	ocscp_object_pool_name
Examples	ocscp_worker_object_pool_borrow_conflict_total{ocscp_object_pool_name=STREAM_INFO_OBJECT_POOL}

Table 5-226 ocscp_metric_message_current_request_rate_total

Field	Details
Description	The total number of current request rate per rate limit bucket key.
Type	Counter
Dimensions	- ocscp_fqdn - bucketKey
Examples	- ocscp_metric_message_current_request_rate_total{ocscp_fqdn="ocscp-scp-worker.scpsvc.svc.cluster.local"} - ocscp_metric_message_current_request_rate_total{bucketKey="egress_pcf1svc.ny-ocscp-01.svc.thrust2"}

Table 5-227 coherence_cache_hits

Field	Details
Description	The total number of cache hits since the last statistics reset. A cache hit refers to a read operation invocation for which an entry exists in this map.
Type	Counter
Dimensions	- coherenceCluster - name (cache name)
Examples	coherence_cache_hits{coherenceCluster="scp-coherence-cluster"}

Table 5-228 coherence_cache_size

Field	Details
Description	The total number of entries in cache.
Type	Counter
Dimensions	- coherenceCluster - name (cache name)
Examples	coherence_cache_size{coherenceCluster="scp-coherence-cluster"}

Table 5-229 ocscp_worker_https_ingress_connection_success_total

Field	Details
Description	This metric will be pegged upon the successful establishment of an Ingress HTTPS connection. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimensions	- tls_version - source - destination
Examples	- ocscp_worker_https_ingress_connection_success_total{tls_version="TLSv1.3"} - ocscp_worker_https_ingress_connection_success_total{source="192.168.219.122"} - ocscp_worker_https_ingress_connection_success_total{destination="192.168.219.88:9443"}

Table 5-230 ocscp_worker_https_ingress_connection_failure_total

Field	Details
Description	This metric will be pegged by Ingress connection failures, where failure refers to the inability to establish a connection due to handshake failure. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimensions	- failureReason - tls_version - source - destination
Examples	- ocscp_worker_https_ingress_connection_failure_total{destination="192.168.219.88:9443"} - ocscp_worker_https_ingress_connection_failure_total{tls_version="TLSv1.3"} - ocscp_worker_https_ingress_connection_failure_total{source="192.168.219.122"} - ocscp_worker_https_ingress_connection_failure_total{ failureReason="TLS Handshake failed. Reason: Client requested protocol TLSv1.2 is not enabled or supported in server context"}

Table 5-231 ocscp_worker_https_egress_connection_success_total

Field	Details
Description	This metric will be pegged upon the successful establishment of an egress HTTPS connection. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimensions	- tls_version - destination
Examples	- ocscp_worker_https_egress_connection_success_total{tls_version="TLSv1.3"} - ocscp_worker_https_egress_connection_success_total{destination="10.98.212.210:8443"}

Table 5-232 ocscp_worker_https_egress_connection_failure_total

Field	Details
Description	This metric will be pegged in the case of egress HTTPS connection failures. Here, failure refers to the inability to establish a connection due to handshake failure. Note: Peg this metric for both HTTP and HTTPS traffic because Hypertext Transfer Protocol Secure (HTTPS) is the secured version of HTTP.
Type	Counter
Dimensions	- failureReason - tls_version - destination
Examples	- ocscp_worker_https_egress_connection_failure_total{tls_version="TLSv1.3"} - ocscp_worker_https_egress_connection_failure_total{destination="10.98.212.210:8443"} - ocscp_worker_https_egress_connection_failure_total{failureReason="certificate expired"}

Table 5-233 ocscp_worker_5gsbi_upstream_connect_failure_total

Field	Details
Description	This metric is pegged when an upstream connection with a producer NF fails.
Type	Counter
Dimensions	- ocscp_producer_host (default value = unkown, Contains fqdn of producer) - ocscp_nf_end_point (default value =unkown, Contains IPAddress and Port of producer)
Examples	- ocscp_worker_5gsbi_upstream_connect_failure_total{ocscp_nf_end_point="10.233.19.102:8080",ocscp_producer_host="udm1svc.scpsvc.svc.cluster.local"} - ocscp_worker_5gsbi_upstream_connect_failure_total{ocscp_nf_end_point="[fd00:0:0:2:0:0:0:6978]:8080",ocscp_producer_host="udm1svc.scpsvc.svc.cluster.local"}

Table 5-234 ocscp_nrfproxy_oauth_queue_alert

Field	Details
Description	This metric is pegged when the metric value is either 0 or 1. If the severity reaches Minor, Major, or Critical, the value is 1, else 0.
Type	Gauge
Dimensions	- queueName: name of the queue (accessTokenCohEvExecutorQueue, accessTokenAppEvExecutorQueue, scpUpstreamExecutorQueue, scpDownstreamExecutorQueue) - queueCapacity: The maximum size of queue is 10000 - severity: CRITICAL, MAJOR, MINOR
Examples	- ocscp_nrfproxy_oauth_queue_alert{severity="CRITICAL"} - ocscp_nrfproxy_oauth_queue_alert{severity="MAJOR"} - ocscp_nrfproxy_oauth_queue_alert{severity="MINOR"}

Table 5-235 ocscp_worker_pod_overload_control_cpu_utilization_total

Field	Details
Description	This metric is pegged when the CPU threshold exceeds the configured overload levels.
Type	Counter
Dimensions	ocscp_threshold_level
Examples	ocscp_worker_pod_overload_control_cpu_utilization_total{ocscp_threshold_level="MINOR"}

Table 5-236 ocscp_worker_self_oci_above_oci_conveyance_threshold_total

Field	Details
Description	This metric is pegged when SCP's load exceeds the configured load for self OCI conveyance.
Type	Counter
Dimensions	ocscp_oci_threshold_level
Examples	ocscp_worker_self_oci_above_oci_conveyance_threshold_total{ocscp_oci_threshold_level="CRITICAL"}

5.2.2 Mediation Metrics

The following table describes metric supported by the Mediation feature:

Table 5-237 ocscp_med_total_rule_count

Field	Details
Description	Total number of Mediation rules configured.
Type	Counter
Dimension	- app (nf-mediation,nf-mediation-test) - nfInstanceId - vendor

Table 5-237 (Cont.) ocscp_med_total_rule_count

Field	Details
Example usage to filter metric by dimension on the Grafana GUI	ocscp_med_total_rule_count{app="nf-mediation",nfInstanceId="IWF1",vendor="oracle"}

Table 5-238 ocscp_med_active_rule_count

Field	Details
Description	Total number of rules executed when a rule match is found for a particular message.
Type	Counter
Dimension	- app (nf-mediation,nf-mediation-test) - nfInstanceId - vendor
Example usage to filter metric by dimension on the Grafana GUI	ocscp_med_active_rule_count_total{app="nf-mediation",nfInstanceId="IWF1",vendor="oracle"}

Table 5-239 ocscp_med_individual_rule_exec_count_total

Field	Details
Description	Total number of times a particular rule gets invoked.
Type	Counter
Dimension	- app (nf-mediation,nf-mediation-test) - nfInstanceId - vendor - ruleName - ruleGroupName
Example usage to filter metric by dimension on the Grafana GUI	sum(ocscp_med_individual_rule_exec_count_total{app="nf-mediation"}) by (ruleName,ruleGroupName)

Table 5-240 ocscp_med_http_req_total

Field	Details
Description	Total number of ingress messages to NF.
Type	Counter
Dimension	- app (nf-mediation) - nfInstanceId - vendor - ruleGroupName - msgType (request, response)
Example usage to filter metric by dimension on the Grafana GUI	ocscp_med_req_total{msgType="request",app="nf-mediation",ruleGroupName="scp-agenda"}

Table 5-241 ocscp_med_http_rsp_total

Field	Details
Description	Total number of egress messages.
Type	Counter
Dimension	- app (nf-mediation) - nfInstanceId - vendor - ruleGroupName - statusCode (supports specific codes:: 200, 500, 503) - msgType (request, response)
Example usage to filter metric by dimension on the Grafana GUI	ocscp_med_rsp_total{msgType="request",ruleGroupName="scp-agenda",statusCode~="2.*"}

Table 5-242 ocscp_med_test_req_total

Field	Details
Description	Total number of incoming messages to the NF Test mode.
Type	Counter
Dimension	- app (nf-mediation-test) - nfInstanceId - vendor - ruleGroupName - msgType (request, response)
Example usage to filter metric by dimension on the Grafana GUI	ocscp_med_test_req_total{msgType="request",app="nf-mediation-test",ruleGroupName="scp-agenda"}

Table 5-243 ocscp_med_test_rsp_total

Field	Details
Description	Total number of response by the Test mode.
Type	Counter
Dimension	- app (nf-mediation-test) - nfInstanceId - vendor - ruleGroupName - statusCode (supports specific codes:: 200, 500, 503) - msgType (request, response)
Example usage to filter metric by dimension on the Grafana GUI	ocscp_med_test_rsp_total{msgType="request"}

Table 5-244 ocscp_med_msg_forwarded_to_test_mode_total

Field	Details
Description	Total Number of Requests forwarded to Test mode
Type	Counter
Dimension	app(nf-mediation)

Table 5-244 (Cont.) ocscp_med_msg_forwarded_to_test_mode_total

Field	Details
Example usage to filter metric by dimension on the Grafana GUI	sum(ocscp_med_msg_forwarded_to_test_mode_total)

Table 5-245 ocscp_med_rule_update_status

Field	Details
Description	If rules reloading failed or successful. Value = 0 {failed} Value = 1 {successful}
Type	Gauge
Dimension	- app (nf-mediation, nf-mediation-test) - vendor - nfInstanceId
Example usage to filter metric by dimension on the Grafana GUI	ocscp_med_rule_update_status{app="nf-mediation"}

Table 5-246 ocscp_med_individual_rule_processing_time

Field	Details
Description	Processing time of the every rule invoked.
Type	Histogram
Dimension	- app (nf-mediation, nf-mediation-test) - vendor - nfInstanceId - ruleName - ruleGroupName
Example usage to filter metric by dimension on the Grafana GUI	-

Table 5-247 ocscp_med_msg_processing_time_seconds

Field	Details
Description	Processing time of messages received by Mediation.
Type	Histogram
Dimension	- app (nf-mediation, nf-mediation-test) - vendor - nfInstanceId
Example usage to filter metric by dimension on the Grafana GUI	-

Table 5-248 ocscp_med_forward_to_ext_total

Field	Details
Description	Total number of messages forwarded to external.

Table 5-248 (Cont.) ocscp_med_forward_to_ext_total

Field	Details
Type	Counter
Dimension	• app (nf-mediation,nf-mediation-test) • vendor • nfInstanceId
Example usage to filter metric by dimension on the Grafana GUI	-

Table 5-249 ocscp_med_incoming_rsp_from_ext_total

Field	Details
Description	Total number of responses from external.
Type	Counter
Dimension	• app (nf-mediation,nf-mediation-test) • vendor • nfInstanceId
Example usage to filter metric by dimension on the Grafana GUI	-

Table 5-250 ocscp_med_incoming_d2h_req_total

Field	Details
Description	Total number of requests from D2H service as a part of the protocol translation mode.
Type	Counter
Dimension	• app (nf-mediation,nf-mediation-test) • vendor • nfInstanceId
Example usage to filter metric by dimension on the Grafana GUI	-

Table 5-251 ocscp_med_outgoing_rsp_to_d2h_total

Field	Details
Description	Total number of responses to D2H service as a part of protocol translation mode.
Type	Counter
Dimension	• app (nf-mediation,nf-mediation-test) • vendor • nfInstanceId
Example usage to filter metric by dimension on the Grafana GUI	-

Table 5-252 ocscp_med_forward_to_h2d_total

Field	Details
Description	Total number of messages forwarded to H2D as a part of protocol translation mode.
Type	Counter
Dimension	• app (nf-mediation,nf-mediation-test) • vendor • nflInstanceId
Example usage to filter metric by dimension on the Grafana GUI	-

Table 5-253 ocscp_med_incoming_rsp_from_h2d_total

Field	Details
Description	Total number of responses received from H2D as a part of protocol translation as a service.
Type	Counter
Dimension	• app (nf-mediation,nf-mediation-test) • vendor • nflInstanceId
Example usage to filter metric by dimension on the Grafana GUI	-

Note

In the above table, <metricPrefix> indicates metric prefix that can be customized. The default metric prefix is ocmed.

6

Alerts

This section provides information about the supported alerts and how to configure the alerts.

Note

The performance and capacity of the SCP system may vary based on the call model, feature or interface configuration, network conditions, and underlying CNE and hardware environment.

You can configure alerts in Prometheus and ScpAlertrules.yaml file.

The following table provides information about Service Communication Proxy (SCP) alerts.

Caution

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

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 Environment (OCCNE) version of kube-api server.
- The alert file can be customized as required by the deployment environment. For example, namespace can be added as a filtered criteria to the alert expression to filter alerts only for a specific namespace.

6.1 System level alerts

This section lists the system level alerts.

6.1.1 SCPNotificationPodMemoryUsage

Table 6-1 SCPNotificationPodMemoryUsage

Field	Description
Severity	Major

Table 6-1 (Cont.) SCPNotificationPodMemoryUsage

Field	Description
Conditions	sum(container_memory_usage_bytes{image!="",pod=~".*scpc-notification.+"}) by (pod,namespace, instance) > 3006477107
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3001
Description	Notify Notification service Pod memory usage if it is above threshold Threshold value is 85% of allocated (4GB) memory: 3.4 GB
Recommended Actions	Cause: When high notification rate or very large NF profile size is present in notifications. Diagnostic Information: Monitor the notification metric: <code>ocscp_nrf_notifications_requests_nf_total</code>. Notification usage reduces after some time when it crosses 2.5 GB or 3 GB. Recovery: This alert is cleared automatically when the scpc-notification pod memory usage reduces below the defined threshold. Reduce the notification rate. These notifications are generated by NRF and can be controlled through NRF. For any assistance, contact My Oracle Support.

6.1.2 SCPWorkerPodMemoryUsage

Table 6-2 SCPWorkerPodMemoryUsage

Field	Description
Severity	major
Conditions	sum(container_memory_usage_bytes{image!="",pod=~".*scp-worker.+"}) by (pod,namespace, instance) > 6012954214
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7004
Description	Notify Worker per Pod memory usage is above threshold Threshold value is 85% of allocated (8GB) memory: 7.3 GB
Recommended Actions	Cause: When there is high traffic rate, alternate routing, more number of routing rules and rules size, and due to network or producer NF latency. Diagnostic Information: Monitor traffic rate, alerts, and latency on the KPI Dashboard. Check the traffic rates of the following metrics if they are too high: • <code>ocscp_metric_http_rx_req_total</code> • <code>ocscp_metric_http_tx_req_total</code> • <code>ocscp_metric_http_rx_res_total</code> • <code>ocscp_metric_http_tx_res_total</code> Check the upstream response time by using the following command and ensure whether upstream is taking too long to respond: <code>ocscp_metric_upstream_service_time_total</code>. Check the following platform metric for current memory usage by the scp-worker pod: <code>container_memory_usage_bytes</code>. Recovery: This alert is cleared automatically when the scp-worker pod memory usage reduces below the defined threshold. Reduce the traffic rate and improve the latency. For any assistance, contact My Oracle Support.

6.1.3 SCPInstanceDown

Table 6-3 SCPInstanceDown

Field	Description
Severity	Critical
Conditions	kube_pod_status_ready{pod =~ '.*scp-worker.*'.*scpc-notification.*'.*scpc-subscription.*'.*scpc-configuration.*'.*scpc-audit.*'.*scpc-alternate-resolution.*',condition =~ 'true'} !=1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7006
Description	Notify that if any pod in ocsdp release is down. Provides information like pod name, instance id and app name.
Recommended Actions	Cause: When the following issues occur: - The Control plane pods, such as configuration, subscription, notification, audit, and alternate-resolution are down due to connection failure with DB. - Pod restarts due to kubernetes liveness or readiness probe failures. - Application restarts or starts failure. Diagnostic Information: - Check if DB services are active by running the following command: <pre>kubectl describe pod <podname> -n <namespace></pre> - Check kubernetes events for probe failures in the platform logs. - Check if any exception is reported in the SCP application logs. Recovery: This alert is cleared automatically when the inactive pod is active. Recover DB services if down. Collect the application logs and contact My Oracle Support for any assistance.

6.2 Application level alerts

This section lists the application level alerts.

6.2.1 SCPCcaFeatureEnabledWithoutHttps

Table 6-4 SCPCcaFeatureEnabledWithoutHttps

Field	Description
Severity	Info
Condition	ocscp_worker_cca_validation_feature_enabled_without_https > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.9022
Description	An alert is raised when the CCA validation feature is enabled without enabling HTTPS.

Table 6-4 (Cont.) SCPcaFeatureEnabledWithoutHttps

Field	Description
Recommended Actions	Cause: CCA validation feature is enabled without enabling HTTPS. Diagnostic Information: Deploy HTTPS SCP deployment. Recovery: The alert is cleared automatically if either the CCA feature is disabled or deployment is changed to HTTPS. For any assistance, contact My Oracle Support.

6.2.2 SCPIngressTrafficRateAboveMinorThreshold

Table 6-5 SCPIngressTrafficRateAboveMinorThreshold

Field	Description
Severity	minor
Condition	sum(rate(ocscp_metric_http_rx_req_total{app_kubernetes_io_name="scp-worker"}[2m]))by (kubernetes_namespace,ocscp_locality,kubernetes_pod_name)>= 1200 to 1400
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7001
Description	Notify that Traffic rate is from 1200 to 1400 mps (user configure minor threshold value) with Locality and current value of traffic rate.
Recommended Actions	Cause: When the Consumer NF sends more traffic than expected. Diagnostic Information: Monitor the ingress traffic to pod using the KPI Dashboard. Refer to the rate of ocscp_metric_http_rx_req_total metric on the Grafana GUI. Recovery: This alert is cleared automatically when the ingress traffic reduces below the minor threshold or exceeds the major threshold. If this alert is not cleared, then check the Consumer NF for an uneven distribution of traffic per connection or for any other issue. For any assistance, contact My Oracle Support.

6.2.3 SCPIngressTrafficRateAboveMajorThreshold

Table 6-6 SCPIngressTrafficRateAboveMajorThreshold

Field	Description
Severity	major
Conditions	sum(rate(ocscp_metric_http_rx_req_total{app_kubernetes_io_name="scp-worker"}[2m]))by (kubernetes_namespace,ocscp_locality,kubernetes_pod_name) >= 1400 to 1600
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7001
Description	Notify that Traffic rate is from 1400 to 1600 mps (user configure major threshold value) with Locality and current value of traffic rate.

Table 6-6 (Cont.) SCPIngressTrafficRateAboveMajorThreshold

Field	Description
Recommended Actions	Cause: When the Consumer NF sends more traffic than expected. Diagnostic Information: Monitor the ingress traffic to pod using the KPI Dashboard. Refer to the rate of <code>ocscp_metric_http_rx_req_total</code> metric on the Grafana GUI. Recovery: This alert is cleared automatically when the ingress traffic reduces below the major threshold or exceeds the critical threshold. If this alert is not cleared, then check the Consumer NF for an uneven distribution of traffic per connection or for any other issue. If this alert continues for a long duration, then reduce the ingress traffic from consumer to pod. For any assistance, contact My Oracle Support.

6.2.4 SCPIngressTrafficRateAboveCriticalThreshold

Table 6-7 SCPIngressTrafficRateAboveCriticalThreshold

Field	Description
Severity	critical
Conditions	<code>sum(rate(ocscp_metric_http_rx_req_total{app_kubernetes_io_name="scp-worker"}[2m]))by(kubernetes_namespace,ocscp_locality,kubernetes_pod_name) >= 1600</code>
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7001
Description	Notify that Traffic rate is above 1600mps (user configure critical threshold value) with Locality and current value of traffic rate.
Recommended Actions	Cause: When the Consumer NF sends more traffic than expected. Diagnostic Information: Monitor the ingress traffic to pod using the KPI Dashboard. Refer to the rate of the <code>ocscp_metric_http_rx_req_total</code> metric on the Grafana GUI. Recovery: This alert is cleared automatically when the ingress traffic reduces below the critical threshold. If this alert is not cleared, then check the Consumer NF for an uneven distribution of traffic per connection or for any other issue. If this alert continues for a long duration, then reduce the ingress traffic from consumer to pod. For any assistance, contact My Oracle Support.

6.2.5 SCPRoutingFailedForProducer

Table 6-8 SCPRoutingFailedForProducer

Field	Description
Severity	Info

Table 6-8 (Cont.) SCPRoutingFailedForProducer

Field	Description
Conditions	increase(ocscp_metric_routing_attempt_fail_total{app_kubernetes_io_name="scp-worker"}[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7005
Description	Notify that Routing failed for producer. Provides detail such as NFService Type, NFType, Locality, producer FQDN and value.
Recommended Actions	Cause: When routing fails to select a producer NF due to unavailability of routing rules for an NF service or producer. Diagnostic Information: - Check whether the routing rules are configured for the NF for which routing failed. - Check the notification logs for any error while processing the notification of the NF for which routing failed. Then, run the following command to get the notification logs: <code>kubectl logs <podname> -n <namespace></code> - Check whether the NF is reachable or not by using one of the following steps: - Run the ping command from primary/secondary nodes using IP of service. Example of a ping command: <code>ping <IPAddress></code>. - Run the ping command from inside the pod, if FQDN of service is used. - Instead of using the ping command, you can collect tcpdump for ensuring the connectivity. tcpdump must be run on the debug container for scp-worker microservice. Example of a tcpdump: <code>tcpdump -w capture.pcap -i <pod interface></code>. Recovery: This alert is cleared automatically when the routing is complete for a producer NF or no more traffic is received in the next Prometheus scrape interval. Check if the NF is deregistered. Register the NF to create routing rules if rules do not exist. For any assistance, contact My Oracle Support.

6.2.6 SCPAuditErrorResponse

Table 6-9 SCPAuditErrorResponse

Field	Description
Severity	Info
Conditions	ocscp_audit_error_response > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.4001
Description	Alert is raised when Audit module receives a 3xx, 4xx, or 5xx error from NRF. This alert is labeled with specific nftype, nrfRegionOrSetId, and auditmethod. Note: Alert is cleared on the next audit cycle.

Table 6-9 (Cont.) SCPAuditErrorResponse

Field	Description
Recommended Actions	Cause: When the configured NRF sends error responses, down, or not reachable. Diagnostic Information: - Check if NRF is up and reachable. To check the NRF status, see <i>Oracle Communications Cloud Native Core, Network Repository Function User Guide</i>. - Check if the NF is reachable or not using one of the following steps: - Run the ping command from primary/secondary nodes using IP of NRF. Example of a ping command: <code>ping <IpAddress></code>. - Run the ping command from inside the pod if FQDN of NRF is used. - Instead of using ping, you can collect tcpdump for ensuring connectivity. Example of a tcpdump: <code>tcpdump -w capture.pcap -i <pod interface></code>. - Monitor audit and worker service logs: <code>kubectl logs <pod name> -n <namespace></code> - Check jaeger traces for scp-worker. Recovery: The alert is cleared automatically during the next audit cycle and when no more errors are received. Collect audit and worker service logs and contact My Oracle Support for any assistance.

6.2.7 SCPAuditEmptyNFArrayResponse

Table 6-10 SCPAuditEmptyNFArrayResponse

Field	Description
Severity	Critical
Conditions	<code>ocscp_audit_2xx_empty_nf_array_rx_total > 0</code>
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.4002
Description	Alert is generated when Audit module receives a 2xx response with empty NFInstance array from NRF. Alert is labeled with specific nftype, nrfRegionOrSetId, and auditmethod. Alert is cleared if Audit receives a success response with non-empty NFInstance array or on next audit cycle if topology source is changed to LOCAL.
Recommended Actions	Cause: When NRF does not have any NF registered or due to any error condition on NRF. Diagnostic Information: Check if NRF contains any registered NF and validate as required. For more information, refer to NRF documents. Recovery: This alert is cleared automatically if Audit receives a success response with non-empty NFInstance array or during the next audit cycle when the topology source is changed to LOCAL. Register a NF with NRF or change the topology source to LOCAL. For any assistance, contact My Oracle Support.

6.2.8 DuplicateLocalityFoundInForeignNF

Table 6-11 DuplicateLocalityFoundInForeignNF

Field	Description
Severity	Major
Conditions	ocscp_notification_duplicate_foreign_location > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3002
Description	Alert is raised when an unknown NF or SCP is registered with duplicate locality from the present region.
Recommended Actions	Cause: When SCP discovers a duplicate locality of an NF from an unknown region. Diagnostic Information: Check logs for NF notification received by running the following command: <code>kubectl -n <namespace> logs <pod name></code>. Check the following metric to get the NFInstanceId information for which this alert is raised: <code>ocscp_notification_duplicate_foreign_location (nfInstanceId)</code>. From the metric, get the NF Instance ID, Locality, and serving_scope. Check the NF Profile of the corresponding NF in the unknown region as identified by the serving_scope. Check and correct the locality in the NF Profile to ensure it aligns with localities of that unknown region that should be different from locality of SCP which reported this alert. Recovery: This alert is cleared automatically if an unknown NF or SCP is deregistered or registers update with the correct locality. Re-register NF with correct locality information. Collect logs for notification and audit service. For any assistance, contact My Oracle Support.

6.2.9 ForeignNFLocalityNotServed

Table 6-12 ForeignNFLocalityNotServed

Field	Description
Severity	Critical
Conditions	ocscp_notification_foreign_nf_locality_unserved > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3003
Description	Alert is raised when a Foreign Producer NF's locality is not served by any SCP.

Table 6-12 (Cont.) ForeignNFLocalityNotServed

Field	Description
Recommended Actions	Cause: When SCP discovers an unknown Producer NF's without any locality served by an SCP. Diagnostic Information: Check logs for received NF notification by running the following command: <code>kubectl get pods -n <namespace></code>. Note: Use the complete name of notification pod in the following command: <code>kubectl logs <pod> -n <namespace></code>. Check the following metric to get the NFInstanceId information for which this alert is raised: <code>ocscp_notification_foreign_nf_locality_unserved(nfInstanceId)</code>. Recovery: This alert is cleared automatically if the unknown NF is deregistered or registers update received with locality served by SCP. Re-register NF with correct locality information. For any assistance, contact My Oracle Support.

6.2.10 UnknownLocalityFoundInForeignNF

Table 6-13 UnknownLocalityFoundInForeignNF

Field	Description
Severity	critical
Conditions	<code>ocscp_notification_foreign_nf_locality_absent > 0</code>
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3004
Description	Alert will be raised when a Foreign Producer NF's locality is unknown.
Recommended Actions	Cause: When SCP discovers an unknown Producer NF's without locality information. Diagnostic Information: Check logs for the received NF notification by running the following command: <code>kubectl get pods -n <namespace></code>. Use the complete name of notification pod in the following command: <code>kubectl logs <pod> -n <namespace> -f --tail=0</code> Check the following metric to get the NFInstanceId information for which this alert is raised: <code>ocscp_notification_foreign_nf_locality_absent(nfInstanceId)</code>. Recovery: This alert is cleared automatically if unknown NF is deregistered or registers update received with locality known to SCP. Re-register NF with correct locality information. For any assistance, contact My Oracle Support.

6.2.11 SCPUpsstreamResponseTimeout

Table 6-14 SCPUpsstreamResponseTimeout

Field	Description
Severity	info

Table 6-14 (Cont.) SCPUstreamResponseTimeout

Field	Description
Conditions	<code>idelta(ocscp_metric_upstream_response_timeout_total[2m]) > 0</code>
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7011
Description	Alert is raised when Upstream connection to a producer NF fails
Recommended Actions	Cause: When a producer NF is down, not reachable, or latency is high. Diagnostic Information: Check whether the producer NF is up and network connectivity to the producer NF is established by using one of the following steps: - Run the ping command from primary/secondary nodes by using IP of producer NF. Example of a ping command: <code>ping <IPAddress></code>. - Run the ping command from inside the pod, if FQDN of producer NF is used. - Instead of using the ping command, you can collect tcpdump for ensuring the connectivity. tcpdump must be run on the debug container for scp-worker microservice. Example of a tcpdump: <code>tcpdump -w capture.pcap -i <pod interface></code>. Check the upstream response time by using the following metric and determine if upstream is taking too long to respond: ocscp_metric_upstream_service_time (producer FQDN) Recovery: This alert is cleared automatically in the next scrape interval if the system does not observe any error. For any assistance, contact My Oracle Support.

6.2.12 SCPSingleNfInstanceAvailableForNFType

Table 6-15 SCPSingleNfInstanceAvailableForNFType

Field	Description
Severity	Major
Conditions	<code>ocscp_no_nf_instance == 0</code>
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3005
Description	Alert is raised when there is a single NFInstance available with SCP for an NFType.
Recommended Actions	Cause: When the <code>preventiveAuditOnLastNFInstanceDeletion</code> attribute is set to true, SCP has single NFInstance available for an NFType. Diagnostic Information: Check all SCP NRFs for specific NFType in the alert if only one NFInstance is available. For information about registered NFs, see <i>Oracle Communications Cloud Native Core, Network Repository Function User Guide</i>. Check the number of NFs of a particular type by using API or CNC console of SCP. For information about procedures to check the NFs available with SCP, see "Configuring Service Communication Proxy using the CNC Console" in <i>Oracle Communications Cloud Native Core, Service Communication Proxy User Guide</i>. Recovery: This alert is cleared automatically in the next scrape interval if more than one NFInstance is available for a specified NFType in the alert. For any assistance, contact My Oracle Support.

6.2.13 SCPNoNfInstanceForNFType

Table 6-16 SCPNoNfInstanceForNFType

Field	Description
Severity	Critical
Conditions	ocscp_no_nf_instance == 1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3006
Description	Alert is raised when there is a no NFInstance available with SCP for a NFType
Recommended Actions	Cause: When the preventiveAuditOnLastNFInstanceDeletion flag is set to true, SCP has no NFInstance available for a NFType. Diagnostic Information: Check all SCP NRFs for specific NFType in the alert if no NFInstance is available. For information about registered NFs, see <i>Oracle Communications Cloud Native Core, Network Repository Function User Guide</i>. Check the number of NFs of a particular type by using API or CNC console of SCP. For information about procedures to check the NFs available with SCP, see "Configuring Service Communication Proxy using the CNC Console" in <i>Oracle Communications Cloud Native Core, Service Communication Proxy User Guide</i>. Recovery: This alert is cleared automatically in the next scrape interval if at least one NFInstance is available for a specified NFType in the alert. For any assistance, contact My Oracle Support.

6.2.14 SCPIngressTrafficRateExceededConfiguredLimit

Table 6-17 SCPIngressTrafficRateExceededConfiguredLimit

Alert Parameters	Value
Description	Ingress traffic rate exceeds configured rate limit for consumer fqdn: {{ \$labels.ocscp_consumer_fqdn }}
Summary	'Ingress traffic rate exceeds configured rate limit for consumer fqdn: ocscpconsumerfqdn = {{ \$labels.ocscp_consumer_host }}, consumernfinstanceid = {{ \$labels.ocscp_consumer_nf_instance_id }}, consumernftype = {{ \$labels.ocscp_consumer_nf_type }}, configuredingressrate = {{ \$labels.ocscp_configured_ingress_rate }}, namespace: {{ \$labels.kubernetes_namespace }}, podname: {{ \$labels.kubernetes_pod_name }}, scp_fqdn: ' {{ \$labels.scp_fqdn }} ', timestamp: {{ with query "time()" }} {{ . first value humanizeTimestamp }} {{ end }} and value = {{ \$value }} '
Severity	Critical
Condition	This alert is raised when the ingress traffic rate exceeds the configured rate for consumer FQDN. increase(ocscp_metric_ingress_rate_limiting_throttle_req_total[2m]) > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7012
Metric Used	ocscp_metric_ingress_rate_limiting_throttle_req_total

Table 6-17 (Cont.) SCPIngressTrafficRateExceededConfiguredLimit

Alert Parameters	Value
Recommended Actions	Cause: When the ingress traffic rate exceeds the configured rate limit for the consumer FQDN. Diagnostic Information: - Check the ingress traffic rate from the consumer FQDN. - To check the ingress rate, refer to the following metrics: cscp_metric_http_rx_req_total - Check the ingress rate limit configuration as described in <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>. Recovery: This alert is cleared when no more requests get suppressed due to ingress rate limiting in the next scrape interval. For any assistance, contact My Oracle Support.

6.2.15 SCPIngressTrafficRoutedWithoutRateLimitTreatment

Table 6-18 SCPIngressTrafficRoutedWithoutRateLimitTreatment

Alert Parameters	Value
Description	Ingress traffic routed without rate limit treatment
Summary	'Ingress traffic routed without rate limit treatment: consumernftype = {{ \$labels.ocscp_consumer_nf_type }}, consumernfinstanceid = {{ \$labels.ocscp_consumer_nf_instance_id }}, consumerfqdn = {{ \$labels.ocscp_consumer_host }}, cause = {{ \$labels.ocscp_cause }}, namespace: {{ \$labels.kubernetes_namespace }}, podname: {{ \$labels.kubernetes_pod_name }}, scp_fqdn: ' {{ \$labels.scp_fqdn }}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }} and value = {{ \$value }}'
Severity	Major
Condition	This alert is raised when the ingress traffic routes without rate limiting treatment. increase(ocscp_metric_ingress_rate_limiting_not_applied_req_total[2m]) > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7013
Metric Used	ocscp_metric_ingress_rate_limiting_not_applied_req_total
Recommended Actions	Cause: When the ingress traffic routes without rate limiting treatment. Diagnostic Information: - Check the ingress rate limiting configurations for the untreated FQDNs that can be obtained from the following metric: ocscp_metric_ingress_rate_limiting_not_applied_req_total(ocscp_consumer_fqdn) - Check the ingress rate limit configuration as described in <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>. Recovery: This alert is cleared when no more requests get routed without ingress rate limiting treatment in the next scrape interval. For any assistance, contact My Oracle Support.

6.2.16 SCPEgressTrafficRateExceededConfiguredLimit

Table 6-19 SCPEgressTrafficRateExceededConfiguredLimit

Field	Description
Summary	'Egress traffic rate exceeds configured rate limit: producerntype = {{labels.ocscp_nf_type}}, producerntfservicetype = {{labels.ocscp_nf_service_type}}, producerninstanceid = {{labels.ocscp_nf_instance_id}}, producerfqdn = {{labels.ocscp_producer_host}}, consumerntype = {{labels.ocscp_consumer_nf_type}}, consumerntinstanceid = {{labels.ocscp_consumer_nf_instance_id}}, consumerfqdn = {{labels.ocscp_consumer_host}}, configuredegressrate = {{labels.ocscp_configured_egress_rate}}, namespace: {{labels.kubernetes_namespace}}, podname: {{labels.kubernetes_pod_name}}, scp_fqdn: ' {{labels.scp_fqdn}} ', timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }} and value = {{ \$value }} '
Severity	Critical
Conditions	idelta(ocscp_metric_egress_rate_limiting_throttle_req_total{app_kubernetes_io_name="scp-worker"}[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7014
Description	Alert is raised when the egress traffic rate exceed the configured rate.
Recommended Actions	Cause: When the egress traffic rate exceeds the configured rate. Diagnostic Information: Check the egress traffic rate by using the following metric: ocscp_metric_http_tx_req_total. Check the egress rate limit configuration as described in <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>. Recovery: This alert is cleared when no more requests get suppressed due to egress rate limiting in the next scrape interval. For any assistance, contact My Oracle Support.

6.2.17 SCPEgressTrafficRoutedWithoutRateLimitTreatment

Table 6-20 SCPEgressTrafficRoutedWithoutRateLimitTreatment

Field	Description
Severity	Major
Conditions	idelta(ocscp_metric_egress_rate_limiting_not_applied_req_total{app_kubernetes_io_name="scp-worker"}[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7015
Description	Alert is raised when egress traffic routes without rate limiting.

Table 6-20 (Cont.) SCPEgressTrafficRoutedWithoutRateLimitTreatment

Field	Description
Recommended Actions	Cause: When the egress traffic routes without rate limiting treatment. Diagnostic Information: Check the egress rate limiting configurations for the untreated producer FQDN. Obtain the producer FQDN by using the following metric: ocscp_metric_egress_rate_limiting_not_applied_req_total(ocscp_producer_fqdn) Check the egress rate limit configuration as described in <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>. Recovery: This alert is cleared when no more requests get routed without egress rate limiting treatment in the next scrape interval. For any assistance, contact My Oracle Support.

6.2.18 SCPNotificatoinRejectTopologySourceLocal

Table 6-21 SCPNotificatoinRejectTopologySourceLocal

Field	Description
Severity	Info
Conditions	increase(ocscp_notifications_rejected_topologysource_local_total[15m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3007
Description	Alert is raised when SCP rejects a notification from NRF due to topology source set to LOCAL for NF Type.
Recommended Actions	Cause: When NF Topology Source Info is set to LOCAL. Diagnostic Information: Check the topology source information of an NF Type. For information about the topology source APIs, see <i>Oracle Communications Cloud Native Core, Service Communication Proxy User Guide</i>. Recovery: This alert is cleared automatically after 15 minutes when NF Topology Source Info is set to NRF from LOCAL. For any assistance, contact My Oracle Support.

6.2.19 SCPNotificationProcessingFailureForNF

Table 6-22 SCPNotificationProcessingFailureForNF

Field	Description
Severity	Major
Conditions	increase(ocscp_failure_processed_nf_notification_total[15m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3008
Description	Alerts is raised when Notification processing has failed on SCP.

Table 6-22 (Cont.) SCPNotificationProcessingFailureForNF

Field	Description
Recommended Actions	Cause: When Notification processing has failed on SCP. Diagnostic Information: Check notification pod logs for any errors by running the following command: <pre>kubectl logs <notification pod name> -n <scp namespace></pre> . To get the list of pods, run the following command: <pre>kubectl get pod -n <scp namespace></pre> Sample logs: <pre>{ "instant": { "epochSecond":1620272241,"nanoOfSecond":609935406}, "thread":"runQueueThreadPool1","level":"ERROR","loggerName":"com.oracle.cgbu.cne.scp.soothsayer.Process","message":"{logMsg=Notified profile IP endpoints already present in stored profile, logMsgCode=DUPLICATE_IPENDPOINT_OR_FQDN_FOUND_IN_STORED_PROFILE, rootCause=Notified Profile contains an ipEndPoint {\\"ipv4Address\\":\\"10.75.203.74\\",\\"transport\\":\\"TCP\\",\\"port\\":32673} which is already present in stored Profile with nfInstanceId 93ED74AA-A29C-4450-9D7A-9278CAF6266D for serviceInstanceId audmcl08nv08-udmueauthn-589d6d5bcc-4dslh}", "endOfBatch":false, "loggerFqcn":"org.apache.logging.log4j.spi.AbstractLogger", "threadId":34, "threadPriority":5, "messageTimestamp":"21-05-06 03:37:21.609+0000", "application":"ocscp-soothsayer", "microservice":"ocscp-scp-notification", "engVersion":"24.2.5", "mktgVersion":"24.2.5.0.0", "vendor":"oracle", "namespace":"scpsvc", "node":"slave1", "pod":"ocscp-scp-notification-547f699c96-m7nc8", "subsystem":"notification", "instanceType":"prod", "processId":"1"} { "instant": { "epochSecond":1620272241,"nanoOfSecond":734873890}, "thread":"runQueueThreadPool1","level":"WARN","loggerName":"com.oracle.cgbu.cne.scp.soothsayer.scheduler.RunQueueConsumer","message":"{logMsg=Profile Processing failed, nfInstanceId=93ED74AA-A29C-4450-9D7A-9278CAF6266D}", "endOfBatch":false, "loggerFqcn":"org.apache.logging.log4j.spi.AbstractLogger", "threadId":34, "threadPriority":5, "messageTimestamp":"21-05-06 03:37:21.734+0000", "application":"ocscp-soothsayer", "microservice":"ocscp-scp-</pre>

Table 6-22 (Cont.) SCPNotificationProcessingFailureForNF

Field	Description
	notification", "engVersion": "24.2.5", "mktgVersion": "24.2.5.0.0", "vendor": "oracle", "namespace": "scpsvc", "node": "slave1", "pod": "ocscp-scpc-notification-547f699c96-m7nc8", "subsystem": "notification", "instanceType": "prod", "processId": "1"} For information about the topology source APIs, see <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>. Recovery: This alert is cleared automatically after 15 minutes. For any assistance, contact My Oracle Support.

6.2.20 SCPSubscriptionFailureForNFType

Table 6-23 SCPSubscriptionFailureForNFType

Field	Description
Severity	Critical
Conditions	increase(ocscp_subscription_nf_failure_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.2001
Description	Alerts is raised when SCP subscription to NRF has failed. This alert is labeled with specific nfType, nrfRegionOrSetId, and auditmethod.

Table 6-23 (Cont.) SCPSubscriptionFailureForNFType

Field	Description
Recommended Actions	Cause: When the subscription fails for an NF Type with NRF. Diagnostic Information: Check whether NRF is up. To check the NRF status, see the <i>Oracle Communications Cloud Native Core, Network Repository Function User Guide</i>. Check whether the NRF is reachable or not by using one of the following steps: - Run the ping command from primary or secondary nodes using IP of NRF. Example of a ping command: ping <IPAddress>. - Run the ping command from inside the pod if FQDN of NRF is used. - Instead of using ping, you can collect tcpdump for ensuring the connectivity. Example of a tcpdump: tcpdump -w capture.pcap -i <pod interface>. If NRF is up, check scp-worker logs to find any error response from NRF. If there are error responses, monitor NRF logs: kubectl logs <pod name> -n <namespace>. Sample logs: <pre>{ "instant": {"epochSecond":1620275506,"nanoOfSecond":134773910}, "thread":"pool-8- thread-1","level":"ERROR","loggerName":"com.oracle.cgbu.cne.scp.soothsayer.subscription.processor.SubscriptionDataConsumer","message":"{logMsg=Exception occurred while handling action for subscriber data, action=TO_BE_RENEWED, stackTrace=com.oracle.cgbu.cne.scp.soothsayer.subscription.operations.NrfSubscriptionClient.triggerPatchRequest(NrfSubscriptionClient.java:177)\ncom.oracle.cgbu.cne.scp.soothsayer.subscription.processor.SubscriptionDataConsumer.handleAction(SubscriptionsDataConsumer.java:322)\ncom.oracle.cgbu.cne.scp.soothsayer.subscription.processor.SubscriptionDataConsumer.consumeSQ(SubscriptionsDataConsumer.java:140)\ncom.oracle.cgbu.cne.scp.soothsayer.subscription.processor.SubscriptionDataConsumer.run(SubscriptionsDataConsumer.java:84)\njava.base/ java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1130)\njava.base/ java.util.concurrent.ThreadPoolExecutor\$Worker.run(ThreadPoolExecutor.java:630)\njava.base/ java.lang.Thread.run(Thread.java:832)}", "endOfBatch":false, "loggerFqn":"org.apache.logging.log4j.spi.AbstractLogger", "threadId":33, "threadPriority":5, "messageTimestamp":"21-05-06 04:31:46.134+0000", "application":"ocscp-soothsayer", "microservice":"ocscp-scp-subscription", "engVersion":"1.15.0", "mktgVersion":"1.15.0.0.0", "vendor":"oracle", "namespace":"scpsvc", "node":"master", "pod":"ocscp-scp-</pre>

Table 6-23 (Cont.) SCPSubscriptionFailureForNFType

Field	Description
	subscription-55cfb57cc6-2qp2g", "subsystem": "subscription", "instanceType": "prod", "processId": "1"} Recovery: This alert is cleared automatically when NRF is up and running or errors are corrected for received error responses. For any assistance, contact My Oracle Support.

6.2.21 SCPReSubscriptionFailureForNFType

Table 6-24 SCPReSubscriptionFailureForNFType

Field	Description
Severity	Critical
Conditions	increase(ocscp_patch_subscription_nf_failure_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.2002
Description	Alerts is raised when SCP re-subscription to NRF has failed. This alert is labeled with specific nfType, nrfRegionOrSetId, and auditmethod.
Recommended Actions	Cause: When the re-subscription fails for an NF Type with NRF. Diagnostic Information: Check whether NRF is up. To check the NRF status, see the <i>Oracle Communications Cloud Native Core, Network Repository Function User Guide</i>. Check whether the NRF is reachable or not by using one of the following steps: - Run the ping command from primary/secondary nodes by using IP of NRF. Example of a ping command: ping <IPAddress>. - Run the ping command from inside the pod, if FQDN of NRF is used. - Instead of using ping, you can collect tcpdump for ensuring the connectivity. Example of a tcpdump: tcpdump -w capture.pcap -i <pod interface>. Check scp-worker logs to find any error response from NRF. If there are error responses, monitor NRF logs: kubectl logs <pod name> -n <namespace>. Recovery: This alert is cleared automatically when NRF is up and running or errors are corrected for received error responses. For any assistance, contact My Oracle Support.

6.2.22 SCPNrfRegistrationFailureForRegionOrSetId

Table 6-25 SCPNrfRegistrationFailureForRegionOrSetId

Field	Description
Severity	Major
Conditions	increase(ocscp_nrf_registration_failure_total[2m]) > 0

Table 6-25 (Cont.) SCPNrfRegistrationFailureForRegionOrSetId

Field	Description
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.2003
Description	Alerts is raised when SCP registration fails. This alert is labeled with specific ntype, nrfRegionOrSetId, and auditmethod.
Recommended Actions	Cause: When the registration fails for an NF Type with NRF. Diagnostic Information: Check whether NRF is up. To check the NRF status, see the <i>Oracle Communications Cloud Native Core, Network Repository Function User Guide</i>. Check whether the NRF is reachable or not by using one of the following steps: - Run the ping command from primary or secondary nodes by using IP of NRF. Example of a ping command: ping <IPAddress>. - Run the ping command from inside the pod if FQDN of NRF is used. - Instead of using ping, you can collect tcpdump for ensuring the connectivity. Example of a tcpdump: tcpdump -w capture.pcap -i <pod interface>. Check scp-worker logs to find any error response from NRF. If there are error responses, monitor NRF logs: kubectl logs <pod name> -n <namespace>. Sample logs: <pre>{ "instant": { "epochSecond":1620638888,"nanoOfSecond":78728229}, "thread":"registration-0", "level":"ERROR", "loggerName":"com.oracle.cgbu.cne.scp.soothsayer.subscription.processor.NrfRegistrationProcessor", "message":"{logMsg=Registration will be retried after configured interval, configuredIntervalInSec=6}", "endOfBatch":false, "loggerFqn":"org.apache.logging.log4j.spi.AbstractLogger", "threadId":33, "threadPriority":5, "messageTimestamp":"21-05-10 09:28:08.078+0000", "application":"ocscp-soothsayer", "microservice":"ocscp-scpc-subscription", "engVersion":"24.2.5", "mktgVersion":"24.2.5.0.0", "vendor":"oracle", "namespace":"scpsvc", "node":"slave1", "pod":"ocscp-scpc-subscription-66c68b9db6-6g582", "subsystem":"subscription", "instanceType":"prod", "processId":"1"} }</pre> Recovery: This alert is cleared automatically when NRF is up and running or errors are corrected for received error responses. For any assistance, contact My Oracle Support.

6.2.23 SCPNrfHeartbeatFailureForRegionOrSetId

Table 6-26 SCPNrfHeartbeatFailureForRegionOrSetId

Field	Description
Severity	Major
Conditions	increase(ocscp_nrf_heartbeat_failures_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.2004
Description	Alerts is raised when SCP Heartbeat fails.
Recommended Actions	Cause: When the Heartbeat fails for an NF Type with NRF. Diagnostic Information: Check whether NRF is up. To check the NRF status, see the <i>Oracle Communications Cloud Native Core, Network Repository Function User Guide</i>. Check whether the NRF is reachable or not by using one of the following steps: - Run the ping command from primary or secondary nodes by using IP of NRF. Example of a ping command: <code>ping <IPAddress></code>. - Run the ping command from inside the pod if FQDN of NRF is used. - Instead of using ping, you can collect tcpdump for ensuring the connectivity. Example of a tcpdump: <code>tcpdump -w capture.pcap -i <pod interface></code>. Check scp-worker logs to find any error response from NRF. If there are error responses, monitor NRF logs: <code>kubectl logs <pod name> -n <namespace></code>. Recovery: This alert is cleared automatically when NRF is up and running or errors are corrected for received error responses. For any assistance, contact My Oracle Support.

6.2.24 SCPDBOperationFailure

Table 6-27 SCPDBOperationFailure

Field	Description
Severity	Warning
Conditions	increase(ocscp_db_operation_failure_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.2005
Description	Alert is raised for any DB operation failures.
Recommended Actions	Cause: When the SCP DB operation fails. Diagnostic Information: Check whether the DB service is up. Check the status/age of the mysql pod by using the following command: <code>kubectl get pods -n <namespace></code>. Where, <namespace> is the namespace in which mysql pod is deployed. Recovery: This alert is cleared automatically when the DB service is up and running. For any assistance, contact My Oracle Support.

6.2.25 SCPGeneratedErrorsResponseForNFService

Table 6-28 SCPGeneratedErrorsResponseForNFService

Field	Description
Severity	Info
Conditions	increase(ocscp_metric_scp_generated_response_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7016
Description	Alert is raised for NF type for which SCP generated response is triggered.
Recommended Actions	Cause: When the error response is generated for NF Service Type by SCP. Diagnostic Information: Monitor scp-worker logs to determine the reason for error responses generated by SCP. Check for error reason in the logs: kubectl logs <pod name> -n <namespace>. Recovery: This alert is cleared automatically when the cause for error response at SCP worker is corrected and configured. For any assistance, contact My Oracle Support.

6.2.26 SCPCircuitBreakingAppliedForNF

Table 6-29 SCPCircuitBreakingAppliedForNF

Field	Description
Severity	Info
Conditions	ocscp_circuit_breaking_applied > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7017
Description	Alert is raised for NF when circuit breaking is applied.
Recommended Actions	Cause: When Circuit Breaking applies for producer NF FQDN based on the configured http2MaxRequests value. Diagnostic Information: Monitor scp-worker logs for number of error responses when outstanding requests exceed the configured http2MaxRequests value: kubectl logs <pod name> -n <namespace>. Check the latency to upstream producer from SCP. Use the following metric to check the same: ocscp_metric_upstream_service_time_total(ocscp_producer_host or ocscp_nf_end_point). Recovery: This alert is cleared automatically when the configuration for http2MaxRequests for circuit breaking is configured beyond the traffic at worker or lower the traffic than the value configured for circuit breaking. For any assistance, contact My Oracle Support.

6.2.27 SCPUpgradeStarted

Table 6-30 SCPUpgradeStarted

Field	Description
Severity	Info
Conditions	When the SCP upgrade process for an SCP microservice starts.
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.6001
Description	Alert is raised when the SCP upgrade process for an SCP microservice starts.
Recommended Actions	Cause: When SCP upgrade is performed for a particular microservice. Diagnostic Information: Not applicable. Recovery: This alert is cleared automatically in 5 minutes when the <code>customAlertExpiryEnabled</code> parameter is set to <code>false</code> in the <code>ocscp_values.yaml</code> file. Otherwise, it is cleared after a specific duration as specified in the <code>customAlertExpiryDuration</code> parameter when the <code>customAlertExpiryEnabled</code> value is <code>true</code>. For any assistance, contact My Oracle Support.

6.2.28 SCPUpgradeFailed

Table 6-31 SCPUpgradeFailed

Field	Description
Severity	Critical
Conditions	When any SCP microservice upgrade fails during the upgrade process.
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.6002
Description	Alert is raised when any SCP microservice upgrade fails.
Recommended Actions	Cause: When any SCP microservice upgrade fails during the upgrade process. Diagnostic Information: Monitor new hook-jobs that might have failed after multiple attempts. Also, monitor any failed log. Run the following command to check the pod of hook-job: <code>kubectl get pods -n <namespace></code>. Run the following command to check the logs: <code>kubectl logs <pod name> -n <namespace></code>. Recovery: This alert is cleared automatically in 5 minutes when the <code>customAlertExpiryEnabled</code> parameter is set to <code>false</code> in the <code>ocscp_values.yaml</code> file. Otherwise, it is cleared after a specific duration as specified in the <code>customAlertExpiryDuration</code> parameter when the <code>customAlertExpiryEnabled</code> value is <code>true</code>. For any assistance, contact My Oracle Support.

6.2.29 SCPUpgradeSuccessful

Table 6-32 SCPUpgradeSuccessful

Field	Description
Severity	Info
Conditions	When any SCP microservice upgrade is completed.
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.6003
Description	Alert is raised when any SCP microservice upgrade is completed.
Recommended Actions	Cause: When any SCP microservice upgrade is completed. Diagnostic Information: Not applicable. Run the following command to check the pod of hook-job: <code>kubectl get pods -n <namespace></code>. Run the following command to check the logs: <code>kubectl logs <pod name> -n <namespace></code>. Recovery: This alert is cleared automatically in 5 minutes when the <code>customAlertExpiryEnabled</code> parameter is set to <code>false</code> in the <code>ocscp_values.yaml</code> file. Otherwise, it is cleared after a specific duration as specified in the <code>customAlertExpiryDuration</code> parameter when the <code>customAlertExpiryEnabled</code> value is <code>true</code>. For any assistance, contact My Oracle Support.

6.2.30 SCPRollbackStarted

Table 6-33 SCPRollbackStarted

Field	Description
Severity	Info
Conditions	When the rollback process for an SCP microservice starts.
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.6004
Description	Alert is raised when the rollback process for an SCP microservice starts.
Recommended Actions	Cause: When the rollback process for an SCP microservice starts. Diagnostic Information: Not applicable. Recovery: This alert is cleared automatically in 5 minutes when the <code>customAlertExpiryEnabled</code> parameter is set to <code>false</code> in the <code>ocscp_values.yaml</code> file. Otherwise, it is cleared after a specific duration as specified in the <code>customAlertExpiryDuration</code> parameter when the <code>customAlertExpiryEnabled</code> value is <code>true</code>. For any assistance, contact My Oracle Support.

6.2.31 SCPRollbackFailed

Table 6-34 SCPRollbackFailed

Field	Description
Severity	Critical

Table 6-34 (Cont.) SCPRollbackFailed

Field	Description
Conditions	When any SCP microservice rollback fails during the rollback process.
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.6005
Description	Alert is raised when any SCP microservice rollback fails.
Recommended Actions	Cause: When any SCP microservice rollback fails during the rollback process. Diagnostic Information: Monitor new hook-jobs that might have failed after multiple attempts. Also, monitor any failed log. Run the following command to check the pod of hook-job: <pre>kubectl get pods -n <namespace></pre> Run the following command to check the logs: <pre>kubectl logs <pod name> -n <namespace></pre> Recovery: This alert is cleared automatically in 5 minutes when the customAlertExpiryEnabled parameter is set to false in the ocscp_values.yaml file. Otherwise, it is cleared after a specific duration as specified in the customAlertExpiryDuration parameter when the customAlertExpiryEnabled value is true. For any assistance, contact My Oracle Support.

6.2.32 SCPRollbackSuccessful

Table 6-35 SCPRollbackSuccessful

Field	Description
Severity	Info
Conditions	When any SCP microservice rollback is completed.
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.6006
Description	Alert is raised when any SCP microservice rollback is completed.
Recommended Actions	Cause: When any SCP microservice rollback is completed. Diagnostic Information: Not applicable. Recovery: This alert is cleared automatically in 5 minutes when the customAlertExpiryEnabled parameter is set to false in the ocscp_values.yaml file. Otherwise, it is cleared after a specific duration as specified in the customAlertExpiryDuration parameter when the customAlertExpiryEnabled value is true. For any assistance, contact My Oracle Support.

6.2.33 ScpWorkerPodCpuUtilizationAboveWarnThreshold

Table 6-36 ScpWorkerPodCpuUtilizationAboveWarnThreshold

Field	Details
Description	CPU utilization of SCP worker at warn level
Summary	CPU utilization of SCP worker at warn level. namespace: <code>{{labels.kubernetes_namespace}}</code> , podname: <code>{{labels.kubernetes_pod_name}}</code> , timestamp: <code>{{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}</code> and value = <code>{{ \$value }}</code>
Severity	Warning
Condition	This alert is raised when CPU utilization of SCP-Worker reaches the WARN level. <code>ocscp_worker_pod_overload_control_cpu_utilization_warn > 0</code>
OID	1.3.6.1.4.1.323.5.3.35.1.2.7018
Metric Used	<code>ocscp_worker_pod_overload_control_cpu_utilization_warn</code>
Recommended Action	Cause: When CPU utilization of scp-worker reaches the WARN level. Diagnostic Information: 1. Get the configured threshold level values using Pod Overload Control Policy REST APIs. For more information about Pod Overload Control Policy, see <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>. 2. Check the CPU threshold level status from the scp-worker pod logs under the warn level. Recovery: • Reduce the incoming service request rate. • This alert is automatically cleared when CPU utilization is reduced to below WARN threshold level. For any assistance, contact My Oracle Support.

6.2.34 ScpWorkerPodCpuUtilizationAboveMinorThreshold

Table 6-37 ScpWorkerPodCpuUtilizationAboveMinorThreshold

Field	Details
Description	CPU utilization of SCP worker at minor level
Summary	CPU utilization of SCP worker at minor level. namespace: <code>{{labels.kubernetes_namespace}}</code> , podname: <code>{{labels.kubernetes_pod_name}}</code> , timestamp: <code>{{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}</code> and value = <code>{{ \$value }}</code>
Severity	Minor
Condition	This alert is raised when CPU utilization of scp-worker reaches the MINOR level. <code>ocscp_worker_pod_overload_control_cpu_utilization_minor > 0</code>
OID	1.3.6.1.4.1.323.5.3.35.1.2.7019
Metric Used	<code>ocscp_worker_pod_overload_control_cpu_utilization_minor</code>

Table 6-37 (Cont.) ScpWorkerPodCpuUtilizationAboveMinorThreshold

Field	Details
Recommended Action	Cause: When CPU utilization of SCP-Worker reaches the MINOR level. Diagnostic Information: 1. Get the configured threshold level values using Pod Overload Control Policy REST APIs. For more information about Pod Overload Control Policy, see <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>. 2. Check the CPU threshold level status from the scp-worker pod logs under the warn level. Recovery: • Reduce the incoming service request rate. • This alert is automatically cleared when CPU utilization is reduced to below MINOR threshold level. For any assistance, contact My Oracle Support.

6.2.35 ScpWorkerPodCpuUtilizationAboveMajorThreshold

Table 6-38 ScpWorkerPodCpuUtilizationAboveMajorThreshold

Field	Details
Description	CPU utilization of SCP worker at major level
Summary	CPU utilization of SCP worker at major level. namespace: <code>{{ \$labels.kubernetes_namespace }}</code> , podname: <code>{{ \$labels.kubernetes_pod_name }}</code> , timestamp: <code>{{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}</code> and value = <code>{{ \$value }}</code>
Severity	Major
Condition	This alert is raised when CPU utilization of scp-worker reaches the MAJOR level. <code>ocscp_worker_pod_overload_control_cpu_utilization_major > 0</code>
OID	1.3.6.1.4.1.323.5.3.35.1.2.7020
Metric Used	<code>ocscp_worker_pod_overload_control_cpu_utilization_major</code>
Recommended Action	Cause: When CPU utilization of SCP-Worker reaches the MAJOR level. Diagnostic Information: 1. Get the configured threshold level values using Pod Overload Control Policy REST APIs. For more information about Pod Overload Control Policy, see <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>. 2. Check the CPU threshold level status from the scp-worker pod logs under the warn level. Recovery: • Reduce the incoming service request rate. • This alert is automatically cleared when CPU utilization is reduced to below MAJOR threshold level. For any assistance, contact My Oracle Support.

6.2.36 ScpWorkerPodCpuUtilizationAboveCriticalThreshold

Table 6-39 ScpWorkerPodCpuUtilizationAboveCriticalThreshold

Field	Details
Description	CPU utilization of SCP worker at critical level
Summary	CPU utilization of SCP worker at critical level. namespace: {{labels.kubernetes_namespace}}, podname: {{labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }} and value = {{ \$value }}
Severity	Critical
Condition	This alert is raised when CPU utilization of scp-worker reaches the CRITICAL level. ocscp_worker_pod_overload_control_cpu_utilization_critical > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7021
Metric Used	ocscp_worker_pod_overload_control_cpu_utilization_critical
Recommended Action	Cause: When CPU utilization of scp-worker reaches the CRITICAL level. Diagnostic Information: 1. Get the configured threshold level values using Pod Overload Control Policy REST APIs. For more information about Pod Overload Control Policy, see <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>. 2. Check the CPU threshold level status from the scp-worker pod logs under the warn level. Recovery: • Reduce the incoming service request rate. • This alert is automatically cleared when CPU utilization is reduced to below CRITICAL threshold level. For any assistance, contact My Oracle Support.

6.2.37 SCPUnhealthyPeerSCPDetected

Table 6-40 SCPUnhealthyPeerSCPDetected

Field	Details
Description	Next hop SCP is marked unhealthy
Summary	'Next hop SCP is marked unhealthy. peerScpFqdn: {{labels.peerScpName}}, scpFqdn: {{labels.scpFqdn}}, namespace: {{labels.kubernetes_namespace}}, podname: {{labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }} and value = {{ \$value }}'
Severity	Info
Condition	This alert is raised when the peer SCP is marked as unhealthy. ocscp_peer_scp_unhealthy > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7022
Metric Used	ocscp_peer_scp_unhealthy

Table 6-40 (Cont.) SCPUnhealthyPeerSCPDetected

Field	Details
Recommended Action	Cause: The peer SCP is marked as unhealthy because of consecutive failure responses. Diagnostic Information: 1. Check transport failures and routing errors on the peer SCP. 2. Run the ping command from master or slave nodes using IP of Service. Sample command: ping <IPAddress>. 3. If FQDN of service is used, run the ping command from inside the pod. 4. If the pod does not support the ping command, get the debug container of SCP pod. 5. If you do not want to use the ping command, collect tcpdump to establish the connection. Sample command: tcpdump -w capture.pcap -i <pod interface> Recovery: This alert is automatically cleared after the degradation time is over. Degradation time = Number of consecutive degradations multiplied by configured base ejection. For any assistance, contact My Oracle Support.

6.2.38 SCPDnsSrvQueryFailure

Table 6-41 SCPDnsSrvQueryFailure

Field	Details
Description	DNS SRV Query failed with cause {{\$labels.cause}}
Summary	'DNS SRV Query failed with cause {{\$labels.cause}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when the DNS server lookup for SRV fails due to network, servfail, or timed-out errors. ocscp_alternate_resolution_dnssrv_rx_error_res == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.8001
Metric Used	ocscp_alternate_resolution_dnssrv_rx_error_res
Recommended Action	Cause: When the dnsSRVAlternateRouting flag is set to true, if the DNS SRV lookup fails due to network, servfail, or timed-out errors. Diagnostic Information: Check the DNS SRV server status and re-establish the status to normal. Recovery: This alert is automatically cleared when SCP performs a successful DNS SRV query. For any assistance, contact My Oracle Support.

6.2.39 SCPPProducerOverloadThrottled

Table 6-42 SCPPProducerOverloadThrottled

Field	Details
Description	Producer is in Throttled Overload state
Summary	'Producer is in Throttled Overload state. producerFqdn: {{labels.producerFqdn}}, scpfqdn: {{labels.scp_fqdn}}, namespace: {{labels.kubernetes_namespace}}, podname: {{labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}
Severity	Info
Condition	This alert is raised when the producer NF is in the throttled congestion state. ocscp_load_manager_peer_load_throttled_threshold == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.7023
Metric Used	ocscp_load_manager_peer_load_throttled_threshold
Recommended Action	Cause: When the load of producer NF is higher than the throttled threshold configured for the service. Diagnostic Information: 1. Check and configure the throttled threshold for each service using Routing Options REST APIs configurations as described in "Configuring Routing Options" in <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>. 2. Check the load for each service using the NF Profile REST APIs as described in <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>. Recovery: This alert clears automatically when the NF profile is deregistered or changed with load less than the throttled abatement threshold. For any assistance, contact My Oracle Support.

6.2.40 SCPPProducerOverloadAlternateRouted

Table 6-43 SCPPProducerOverloadAlternateRouted

Field	Details
Description	Producer is in Alternate Route Overload state
Summary	'Producer is in Alternate Route Overload state. producerFqdn: {{labels.producerFqdn}}, scpfqdn: {{labels.scp_fqdn}}, namespace: {{labels.kubernetes_namespace}}, podname: {{labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}
Severity	Info
Condition	This alert is raised when the producer NF is in the alternate routing congestion state. ocscp_load_manager_peer_load_alternateRoute_threshold == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.7024

Table 6-43 (Cont.) SCPProducerOverloadAlternateRouted

Field	Details
Metric Used	ocscp_load_manager_peer_load_alternateRoute_threshold
Recommended Action	Cause: When the load of producer NF is higher than alternate routing threshold configured for the service. Diagnostic Information: 1. Check and configure the alternate routing threshold for each service using Routing Options REST APIs configurations as described in "Configuring Routing Options" in <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>. 2. Check the load for each service using the NF Profile REST APIs as described in <i>Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide</i>. Recovery: This alert clears automatically when the NF profile is deregistered or changed with load less than the alternate routing abatement threshold. For any assistance, contact My Oracle Support.

6.2.41 SCPSeppNotConfigured

Table 6-44 SCPSeppNotConfigured

Field	Details
Description	SEPP is not configured for PLMN
Summary	'SEPP is not configured for PLMN'Summary: 'SEPP is not configured for PLMN. plmnid: {{\$labels.plmn_id}}, scpfqdn: {{\$labels.scp_fqdn}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when Security Edge Protection Proxy (SEPP) is not configured. ocscp_metric_sepp_not_configured_current == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.7025
Metric Used	cscp_metric_sepp_not_configured_current
Recommended Action	Cause: When SEPP routing related rules are not configured at SCP for selected PLMN in the inter-PLMN routing. Diagnostic Information: 1. Check whether SEPP profile is registered or SEPP related configuration is made at SCP. 2. Verify the routing rules created for selected inter-PLMN in the PLMN_SEPP_MAPPING table. Recovery: This alert clears automatically when the SEPP related routing rules are created at SCP for selected PLMN in the inter-PLMN routing. For any assistance, contact My Oracle Support.

6.2.42 SCPSeppRoutingFailed

Table 6-45 SCPSeppRoutingFailed

Field	Details
Description	Routing towards SEPP failed
Summary	Routing towards SEPP failed. sepp_fqdn: {{labels.ocscp_sepp_fqdn}}, scpfqdn: {{labels.scp_fqdn}}, namespace: {{labels.kubernetes_namespace}}, podname: {{labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}
Severity	Minor
Condition	This alert is raised when routing towards SEPP fails. ocscp_metric_sepp_routing_attempt_fail_current == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.7026
Metric Used	ocscp_metric_sepp_routing_attempt_fail_current
Recommended Action	Cause: Inter-PLMN routing failed for the selected SEPP instances. Diagnostic Information: 1. Check whether selected SEPP is up and healthy. 2. Check if selected SEPP is reachable. 3. Use the ping command from primary or secondary nodes using IP of SEPP. Sample ping command is ping <IpAddress> 4. If FQDN of SEPP is used, try ping command from inside the pod. 5. Alternative to ping, collect tcpdump for ensuring proper connectivity. Sample command: tcpdump -w capture.pcap -i <pod interface> 6. Check scp-worker logs for any error response from selected SEPP. If there are error responses, monitor selected SEPP logs by running this command: kubectl logs <pod name> -n <namespace> Recovery: This alert clears automatically when routing is successful for selected SEPP. For any assistance, contact My Oracle Support.

6.2.43 SCPGlobalEgressRLRemoteParticipantConnectivityFailure

Table 6-46 SCPGlobalEgressRLRemoteParticipantConnectivityFailure

Field	Details
Description	'SCP Global Egress RL Remote Participant Connectivity Failure for participant
Summary	'SCP Global Egress RL Remote Participant Connectivity Failure for participant: {{labels.scp_remote_coh_cluster_name}}, scp_fqdn: {{labels.scp_fqdn}}, scp_local_coh_cluster_name: {{labels.scp_local_coh_cluster_name}}, scp_remote_coh_cluster_fqdn: {{labels.scp_remote_coh_cluster_fqdn}}, scp_remote_coh_cluster_port: {{labels.scp_remote_coh_cluster_port}}, namespace: {{labels.kubernetes_namespace}}, podname: {{labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}
Severity	Major

Table 6-46 (Cont.) SCPGlobalEgressRLRemoteParticipantConnectivityFailure

Field	Details
Condition	This alert is raised when the remote participant SCP connection is not established or goes down. ocscp_global_egress_rl_remote_participant_connectivity_failure == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9001
Metric Used	ocscp_global_egress_rl_bucketkey_not_rate_controlled_total
Recommended Action	Cause: When the remote participant SCP connection is not established or down. Diagnostic Information: 1. Check whether the FQDN or IP port are properly configured with remote SCP's scp-cache microservice. 2. Check whether clusterName and NFInstanceId are configured for the remote SCP. 3. Check whether communication path is active between the two SCP's scp-cache microservice. 4. Run the following command to monitor scp-cache logs: <code>kubectl logs <pod name> -n <namespace></code> Recovery: This alert clears automatically if the connection is established with the remote participant SCP. For any assistance, contact My Oracle Support.

6.2.44 SCPGlobalEgressRLRemoteParticipantWithDuplicateNFInstanceId

Table 6-47 SCPGlobalEgressRLRemoteParticipantWithDuplicateNFInstanceId

Field	Details
Description	SCP global egress RL remote participant configured with duplicate NF InstanceId for participant.
Summary	'SCP Global Egress RL Remote Participant Configured With Duplicate NFInstanceId for participant: {{ \$labels.scp_remote_coh_cluster_name }}, scp_fqdn: {{ \$labels.scp_fqdn }}, scp_nf_instance_id: {{ \$labels.scp_nf_instance_id }}, scp_local_coh_cluster_name: {{ \$labels.scp_local_coh_cluster_name }}, scp_remote_coh_cluster_fqdn: {{ \$labels.scp_remote_coh_cluster_fqdn }}, scp_remote_coh_cluster_port: {{ \$labels.scp_remote_coh_cluster_port }}, namespace: {{ \$labels.namespace }}, podname: {{ \$labels.pod }}, timestamp: {{ with query "time()" }} {{ . first value humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when a duplicate remote coherence participant is found. ocscp_global_egress_rl_remote_participant_is_duplicate == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9002
Metric Used	ocscp_global_egress_rl_remote_participant_is_duplicate

Table 6-47 (Cont.) SCPGlobalEgressRLRemoteParticipantWithDuplicateNFInstanceId

Field	Details
Recommended Action	Cause: Duplicate configuration of remote coherence participants with local SCP. Diagnostic Information: 1. Ensure Global Rate Limit feature is enabled. 2. Check whether the clusterName and NFInstanceId of local SCPs and remote SCPs are not duplicates. 3. Check whether the clusterName and NFInstanceId of the first remote SCP and the second remote SCP are not duplicates. 4. Configure unique values for clusterName and NFInstanceId for local SCPs as well as remote SCPs. Recovery: - This alert is cleared automatically if no duplicate configurations between local and remote SCPs are found. For any assistance, contact My Oracle Support.

6.2.45 SCPMediationConnectivityFailure

Table 6-48 SCPMediationConnectivityFailure

Field	Details
Description	'SCP Mediation Connectivity Failed, scp_fqdn
Summary	'SCP Mediation Connectivity Failed, scp_fqdn: {{\$labels.scp_fqdn}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{ end }'
Severity	Major
Condition	This alert is raised when Mediation connection is not established or request to Mediation is not successful. ocscp_mediation_http_not_reachable == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9002
Metric Used	ocscp_mediation_http_not_reachable

Table 6-48 (Cont.) SCPMediationConnectivityFailure

Field	Details
Recommended Action	Cause: The remote Mediation connection is not established or request to Mediation is not successful. Diagnostic Information: 1. Check the errors of the ocscp_mediation_http_not_reachable metric on the Grafana dashboard. 2. Run the following command to check the status of the mediation pod: <pre>kubectl get pods -n <namespace></pre> Recovery: 1. If the mediation pod is not in the ready state, run the following command to check the scp-mediation logs: <pre>kubectl logs <pod name> -n <namespace></pre> 2. If the mediation pod is absent, then redeploy or upgrade SCP with mediationService set to true. This alert clears automatically when the connection is established with Mediation when Mediation is invoked from any of the trigger points. For any assistance, contact My Oracle Support.

6.2.46 SCPNotificationQueuesUtilizationAboveMinorThreshold

Table 6-49 SCPNotificationQueuesUtilizationAboveMinorThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Notification Queues Utilization Above Minor Threshold'
Summary	'SCP Notification Queues Utilization Above Minor Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when the queues in the notification service are utilized above 65% of the maximum size (user configure minor threshold value). ocscp_notification_queue_alert{severity="MINOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.3009
Metric Used	ocscp_notification_queue_alert

Table 6-49 (Cont.) SCPNotificationQueuesUtilizationAboveMinorThreshold

Field	Details
Recommended Action	Cause: The Notification module is getting more traffic than expected. Diagnostic Information: 1. Monitor Notification traffic to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_notification_queue_utilization Recovery: This alert clears automatically when notification traffic goes below minor threshold or exceeds major threshold. For any assistance, contact My Oracle Support.

6.2.47 SCPNotificationQueuesUtilizationAboveMajorThreshold

Table 6-50 SCPNotificationQueuesUtilizationAboveMajorThreshold

Field	Details
Description	'instancetype: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Notification Queues Utilization Above Major Threshold'
Summary	'SCP Notification Queues Utilization Above Major Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when the queues in the notification service is utilized above 75% of the maximum size (user configure major threshold value). ocscp_notification_queue_alert{severity="MAJOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.3010
Metric Used	ocscp_notification_queue_alert
Recommended Action	Cause: The Notification module is getting more traffic than expected. Diagnostic Information: 1. Monitor Notification traffic to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_notification_queue_utilization Recovery: This alert clears automatically when notification traffic goes below major threshold or above critical major threshold. For any assistance, contact My Oracle Support.

6.2.48 SCPNotificationQueuesUtilizationAboveCriticalThreshold

Table 6-51 SCPNotificationQueuesUtilizationAboveCriticalThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Notification Queues Utilization Above Critical Threshold'
Summary	'SCP Notification Queues Utilization Above Critical Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when the queues in notification service are utilized above 85% of the maximum size (user configure critical threshold value). ocscp_notification_queue_alert{severity="CRITICAL"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.3011
Metric Used	ocscp_notification_queue_alert
Recommended Action	Cause: The Notification module is getting more traffic than expected. Diagnostic Information: 1. Monitor Notification traffic to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_notification_queue_utilization Recovery: This alert clears automatically when notification traffic goes below critical threshold. For any assistance, contact My Oracle Support.

6.2.49 SCPNrfProxyQueuesUtilizationAboveMinorThreshold

Table 6-52 SCPNrfProxyQueuesUtilizationAboveMinorThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Nrfproxy Queues Utilization Above Minor Threshold'
Summary	'SCP Nrfproxy Queues Utilization Above Minor Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when the task queues in scp-nrfproxy service are utilized above 65% of the maximum size (user configure minor threshold value). ocscp_nrfproxy_queue_alert{severity="MINOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9010
Metric Used	ocscp_nrfproxy_queue_alert

Table 6-52 (Cont.) SCPNrfProxyQueuesUtilizationAboveMinorThreshold

Field	Details
Recommended Action	Cause: NrfProxy task queues are getting filled and the traffic is more than expected. Diagnostic Information: 1. Monitor traffic towards nrfProxy to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_nrfproxy_queue_utilization Recovery: This alert clears automatically when the traffic goes below minor threshold or above major threshold. For any assistance, contact My Oracle Support.

6.2.50 SCPNrfProxyQueuesUtilizationAboveMajorThreshold

Table 6-53 SCPNrfProxyQueuesUtilizationAboveMajorThreshold

Field	Details
Description	'instancetype: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Nrfproxy Queues Utilization Above Major Threshold'
Summary	'SCP Nrfproxy Queues Utilization Above Major Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when the task queues in scp-nrfproxy service are utilized above 75% of the maximum size (user configure major threshold value). ocscp_nrfproxy_queue_alert{severity="MAJOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9011
Metric Used	ocscp_nrfproxy_queue_alert
Recommended Action	Cause: NrfProxy task queues are getting filled and the traffic is more than expected. Diagnostic Information: 1. Monitor traffic towards nrfProxy to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_nrfproxy_queue_utilization Recovery: This alert clears automatically when the traffic goes below major threshold or above critical threshold. For any assistance, contact My Oracle Support.

6.2.51 SCPNrfProxyQueuesUtilizationAboveCriticalThreshold

Table 6-54 SCPNrfProxyQueuesUtilizationAboveCriticalThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Nrfproxy Queues Utilization Above Critical Threshold'
Summary	'SCP Nrfproxy Queues Utilization Above Critical Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when the task queues in scp-nrfproxy service are utilized above 85% of the maximum size (user configure critical threshold value). ocscp_nrfproxy_queue_alert{severity="CRITICAL"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9012
Metric Used	ocscp_nrfproxy_queue_alert
Recommended Action	Cause: NrfProxy task queues are getting filled and the traffic is more than expected. Diagnostic Information: 1. Monitor traffic towards nrfProxy to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_nrfproxy_queue_utilization Recovery: This alert clears automatically when the traffic goes below critical threshold. For any assistance, contact My Oracle Support.

6.2.52 SCPWorkerQueuesUtilizationAboveMinorThreshold

Table 6-55 SCPWorkerQueuesUtilizationAboveMinorThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Worker Queues Utilization Above Minor Threshold'
Summary	'SCP Worker Queues Utilization Above Minor Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when task queues in scp-worker service are utilized above 65% of the maximum size (user configure minor threshold value). ocscp_worker_queue_alert{severity="MINOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9007
Metric Used	ocscp_worker_queue_alert

Table 6-55 (Cont.) SCPWorkerQueuesUtilizationAboveMinorThreshold

Field	Details
Recommended Action	Cause: Worker task queues are getting filled and the traffic is more than expected. Diagnostic Information: 1. Monitor traffic towards nrfProxy to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_worker_queue_utilization Recovery: This alert clears automatically when the traffic goes below minor threshold or above major threshold. For any assistance, contact My Oracle Support.

6.2.53 SCPWorkerQueuesUtilizationAboveMajorThreshold

Table 6-56 SCPWorkerQueuesUtilizationAboveMajorThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Worker Queues Utilization Above Major Threshold'
Summary	'SCP Worker Queues Utilization Above Major Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{ end }{'
Severity	Major
Condition	This alert is raised when task queues in scp-worker service are utilized above 75% of the maximum size (user configure major threshold value). ocscp_worker_queue_alert{severity="MAJOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9008
Metric Used	ocscp_worker_queue_alert
Recommended Action	Cause: Worker task queues are getting filled and the traffic is more than expected. Diagnostic Information: 1. Monitor traffic towards nrfProxy to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_worker_queue_utilization Recovery: This alert clears automatically when the traffic goes below major threshold or goes above critical threshold. For any assistance, contact My Oracle Support.

6.2.54 SCPWorkerQueuesUtilizationAboveCriticalThreshold

Table 6-57 SCPWorkerQueuesUtilizationAboveCriticalThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Worker Queues Utilization Above Critical Threshold'
Summary	'SCP Worker Queues Utilization Above Critical Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when task queues in scp-worker service are utilized above 85% of the maximum size (user configure critical threshold value). ocscp_worker_queue_alert{severity="CRITICAL"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9009
Metric Used	ocscp_worker_queue_alert
Recommended Action	Cause: Worker task queues are getting filled and the traffic is more than expected. Diagnostic Information: 1. Monitor traffic towards nrfProxy to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_worker_queue_utilization Recovery: This alert clears automatically when the traffic goes below critical threshold. For any assistance, contact My Oracle Support.

6.2.55 SCPCacheQueuesUtilizationAboveMinorThreshold

Table 6-58 SCPCacheQueuesUtilizationAboveMinorThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Cache Queues Utilization Above Minor Threshold'
Summary	'SCP Cache Queues Utilization Above Minor Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when the task queues in the scp-cache service are utilized above 65% of their maximum size (the user-configured minor threshold value). ocscp_cache_queue_alert{severity="MINOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.13002
Metric Used	ocscp_cache_queue_utilization

Table 6-58 (Cont.) SCPCacheQueuesUtilizationAboveMinorThreshold

Field	Details
Recommended Action	Cause: When the cache task queues are getting filled, and traffic is higher than expected. Diagnostic Information: 1. Monitor traffic towards the cache pod using the KPI Dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocsdp_cache_queue_utilization. Recovery: The alert is cleared automatically when minor threshold or goes above major threshold. For any assistance, contact My Oracle Support.

6.2.56 SCPCacheQueuesUtilizationAboveMajorThreshold

Table 6-59 SCPCacheQueuesUtilizationAboveMajorThreshold

Field	Details
Description	'instancetype: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Cache Queues Utilization Above Major Threshold'
Summary	SCP Cache Queues Utilization Above Major Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{{ end }}
Severity	Major
Condition	This alert is raised when the task queues in the scp-cache service are utilized above 75% of their maximum size (the user-configured major threshold value). ocsdp_cache_queue_alert{severity="MAJOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.13001
Metric Used	ocsdp_cache_queue_utilization
Recommended Action	Cause: When the cache task queues are getting filled, and traffic is higher than expected. Diagnostic Information: 1. Monitor traffic towards the cache pod using the KPI Dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocsdp_cache_queue_utilization. Recovery: The alert is cleared automatically when traffic falls below a major threshold or goes above a critical threshold. For any assistance, contact My Oracle Support.

6.2.57 SCPCacheQueuesUtilizationAboveCriticalThreshold

Table 6-60 SCPCacheQueuesUtilizationAboveCriticalThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Cache Queues Utilization Above Critical Threshold'
Summary	'SCP Cache Queues Utilization Above Critical Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when the task queues in the scp-cache service are utilized above 85% of their maximum size (the user-configured critical threshold value). ocscp_cache_queue_alert{severity="CRITICAL"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.13000
Metric Used	ocscp_cache_queue_utilization
Recommended Action	Cause: When the cache task queues are getting filled, and traffic is higher than expected. Diagnostic Information: 1. Monitor traffic towards the cache pod using the KPI Dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_cache_queue_utilization. Recovery: The alert is cleared automatically when traffic falls below a critical threshold. For any assistance, contact My Oracle Support.

6.2.58 SCPLoadManagerQueuesUtilizationAboveMinorThreshold

Table 6-61 SCPLoadManagerQueuesUtilizationAboveMinorThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Load Manage Queues Utilization Above Minor Threshold'
Summary	'SCP Load Manager Queues Utilization Above Minor Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when the task queues in the scp-load-manager service are utilized above 65% of their maximum size (the user-configured minor threshold value). ocscp_load_manager_queue_alert{severity="MINOR"}
OID	1.3.6.1.4.1.323.5.3.35.1.2.11002
Metric Used	ocscp_load_manager_queue_utilization

Table 6-61 (Cont.) SCPLoadManagerQueuesUtilizationAboveMinorThreshold

Field	Details
Recommended Action	Cause: When the cache task queues are getting filled, and traffic is higher than expected. Diagnostic Information: 1. Monitor traffic towards the cache pod using the KPI Dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: <code>ocscp_load_manager_queue_utilization</code>. Recovery: The alert is cleared automatically when traffic falls below a minor threshold. For any assistance, contact My Oracle Support.

6.2.59 SCPLoadManagerQueuesUtilizationAboveMajorThreshold

Table 6-62 SCPLoadManagerQueuesUtilizationAboveMajorThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Load Manage Queues Utilization Above Major Threshold'
Summary	'SCP Load Manager Queues Utilization Above Major Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when the task queues in the scp-load-manager service are utilized above 75% of their maximum size (the user-configured major threshold value). <code>ocscp_load_manager_queue_alert{severity="MAJOR"} == 1</code>
OID	1.3.6.1.4.1.323.5.3.35.1.2.11001
Metric Used	<code>ocscp_load_manager_queue_utilization</code>
Recommended Action	Cause: When the cache task queues are getting filled, and traffic is higher than expected. Diagnostic Information: 1. Monitor traffic towards the cache pod using the KPI Dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: <code>ocscp_load_manager_queue_utilization</code>. Recovery: The alert is cleared automatically when traffic falls below a major threshold. For any assistance, contact My Oracle Support.

6.2.60 SCPLoadManagerQueuesUtilizationAboveCriticalThreshold

Table 6-63 SCPLoadManagerQueuesUtilizationAboveCriticalThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}: SCP Load Manager Queues Utilization Above Critical Threshold'
Summary	'SCP Load Manager Queues Utilization Above Critical Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when the task queues in the scp-load-manager service are utilized above 85% of their maximum size (the user-configured critical threshold value). ocscp_load_manager_queue_alert{severity="CRITICAL"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.11000
Metric Used	ocscp_load_manager_queue_utilization
Recommended Action	Cause: When the cache task queues are getting filled, and traffic is higher than expected. Diagnostic Information: 1. Monitor traffic towards the cache pod using the KPI Dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_load_manager_queue_utilization. Recovery: The alert is cleared automatically when traffic falls below a critical threshold. For any assistance, contact My Oracle Support.

6.2.61 SCPProducerNfSetUnhealthy

Table 6-64 SCPProducerNfSetUnhealthy

Field	Details
Description	All producer NFs in NF set are marked unhealthy
Summary	'All producer NFs in NF set are marked unhealthy. nfSet: {{\$labels.ocscp_nf_setid}}, scpFqdn: {{\$labels.scp_fqdn}}, namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Info
Condition	This alert is raised when all producer NFs in an NF Set are marked unhealthy. ocscp_metric_nf_set_unhealthy > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7027
Metric Used	ocscp_metric_nf_set_unhealthy

Table 6-64 (Cont.) SCPProducerNfSetUnhealthy

Field	Details
Recommended Action	Cause: All the producer NFs are marked unhealthy because of consecutive failure responses. Diagnostic Information: 1. Check transport failures and routing errors on producer NFs. 2. Run the ping command from primary or secondary nodes using IP of Service. Sample command: ping <IPAddress>. 3. If FQDN of service is used, run the ping command from inside the pod. 4. If the pod does not support the ping command, get the debug container of SCP pod. 5. If you do not want to use the ping command, collect tcpdump to establish the connection. Sample command: tcpdump -w capture.pcap -i <pod interface> Recovery: This alert is automatically cleared after the degradation time is over. Degradation time = Number of consecutive degradations multiplied by configured base ejection. For any assistance, contact My Oracle Support.

6.2.62 SCPPeerSeppUnhealthy

Table 6-65 SCPPeerSeppUnhealthy

Field	Details
Description	Peer Sepp is marked unhealthy
Summary	'Peer Sepp is marked unhealthy. seppFqdn: {{\$labels.ocscp_sepp_fqdn}}, namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Info
Condition	This alert is raised when peer SEPP is marked unhealthy. ocscp_sepp_unhealthy > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7028
Metric Used	ocscp_sepp_unhealthy

Table 6-65 (Cont.) SCPPeerSeppUnhealthy

Field	Details
Recommended Action	Cause: The peer SEPP is marked unhealthy because of consecutive failure responses. Diagnostic Information: 1. Check transport failures and routing errors on peer SCP. 2. Run the ping command from primary or secondary nodes using IP of Service. Sample command: ping <IPAddress>. 3. If FQDN of service is used, run the ping command from inside the pod. 4. If the pod does not support the ping command, get the debug container of SCP pod. 5. If you do not want to use the ping command, collect tcpdump to establish the connection. Sample command: tcpdump -w capture.pcap -i <pod interface> Recovery: This alert is automatically cleared after the degradation time is over. Degradation time = Number of consecutive degradations multiplied by configured base ejection. For any assistance, contact My Oracle Support.

6.2.63 SCPMicroServiceUnreachable

Table 6-66 SCPMicroServiceUnreachable

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}: SCP communication between the micro-services indicated by source and destination has failed'
Summary	Summary: 'SCP communication between the micro-services indicated by source and destination has failed: {{\$labels.instance}}, namespace: {{\$labels.namespace}}, source:{{\$labels.source}}, destination: {{\$labels.destination}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{ end }{'
Severity	Critical
Condition	This alert is raised when the communication between SCP microservices indicated by source and destination has failed. ocscp_metric_svc_unreachable==1
OID	1.3.6.1.4.1.323.5.3.35.1.2.7029
Metric Used	ocscp_metric_svc_unreachable
Recommended Action	Cause: Communication between SCP microservices has failed. Diagnostic Information: Verify whether endpoints of all the services are in Running and Ready state. If not, restart the services. Recovery: This alert clears automatically when the required services are in Running and Ready state. For any assistance, contact My Oracle Support.

6.2.64 SCPTrafficFeedSendFailed

Table 6-67 SCPTrafficFeedSendFailed

Field	Details
Description	'Sending messages to Traffic Feed failed. Cause : {{\$labels.ocscp_cause}}'
Summary	'Sending messages to Traffic Feed failed, cause: {{\$labels.ocscp_cause}}, scp_fqdn: {{\$labels.scp_fqdn}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when sending messages to traffic feed fails. increase(ocscp_metric_trafficfeed_attempted_total{app_kubernetes_io_name="scp-worker"}[1h]) > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9003
Metric Used	ocscp_metric_trafficfeed_attempted_total
Recommended Action	Cause: Sending of message to traffic feed failed. Diagnostic Information: • Check failure reason. • Check if the traffic feed OCNADD configuration is correct. • Check if the OCNADD server is reachable and available. Recovery: This alert clears automatically after 24 hrs if sending messages to traffic feed stops failing. For any assistance, contact My Oracle Support.

6.2.65 SCPTrafficFeedKafkaClusterUnhealthy

Table 6-68 SCPTrafficFeedKafkaClusterUnhealthy

Field	Details
Description	'Kafka cluster is marked unhealthy, Cause : {{\$labels.ocscp_cause}}'
Summary	'Kafka cluster is marked unhealthy, cause: {{\$labels.ocscp_cause}}, scp_fqdn: {{\$labels.ocscp_fqdn}}, namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when the Kafka cluster is unhealthy. ocscp_metric_trafficfeed_cluster_unhealthy == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9026
Metric Used	ocscp_metric_trafficfeed_cluster_unhealthy
Recommended Action	Cause: The Kafka cluster is unhealthy. Diagnostic Information: • Check and diagnose OCNADD Kafka cluster. Recovery: This alert clears when the Kafka cluster recovers from the failure condition. For any assistance, contact My Oracle Support.

6.2.66 SCPTrafficFeedPartitionUnhealthy

Table 6-69 SCPTrafficFeedPartitionUnhealthy

Field	Details
Description	'Kafka partition {{ \$labels.kafka_partition_id }} is marked unhealthy, Cause : {{ \$labels.ocscp_cause }}'
Summary	'Kafka cluster is marked unhealthy, cause: {{ \$labels.ocscp_cause }}, partition_id: {{ \$labels.kafka_partition_id }}, topic: {{ \$labels.topic }}, scp_fqdn: {{ \$labels.ocscp_fqdn }}, namespace: {{ \$labels.namespace }}, podname: {{ \$labels.kubernetes_pod_name }}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{ end }'
Severity	Major
Condition	This alert is raised when the Kafka partition is unhealthy. ocscp_metric_trafficfeed_partition_unhealthy == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9025
Metric Used	ocscp_metric_trafficfeed_partition_unhealthy
Recommended Action	Cause: The Kafka partition is unhealthy. Diagnostic Information: Check and diagnose OCNADD Kafka cluster. Recovery: This alert clears when the Kafka partition recovers from the failure condition. For any assistance, contact My Oracle Support.

6.2.67 SCPServiceMeshFailure

Table 6-70 SCPServiceMeshFailure

Field	Details
Description	'SCP servicemesh failure encountered'
Summary	'SCP servicemesh failure encountered for nfservicetype: {{ \$labels.ocscp_nf_service_type }}, nftype: {{ \$labels.ocscp_nf_type }}, nfinstanceid: {{ \$labels.ocscp_nf_instance_id }}, serviceinstanceid: {{ \$labels.ocscp_service_instance_id }}, producerfqdn: {{ \$labels.ocscp_producer_host }}, responsecode: {{ \$labels.ocscp_response_code }}, serverheader: {{ \$labels.ocscp_server_header }}, namespace: {{ \$labels.namespace }}, podname: {{ \$labels.pod }}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{ end }'SeverityInfo
Severity	Info
Condition	This alert is raised when service mesh failure occurs. increase(ocscp_metric_sidecarproxy_failures_total[2m]) > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7030
Metric Used	ocscp_metric_sidecarproxy_failures_total

Table 6-70 (Cont.) SCPServiceMeshFailure

Field	Details
Recommended Action	Cause: Service mesh failure observed at SCP. Diagnostic Information: - Service mesh is unable to reach peer NF due to connection failures or host is not known to service mesh or some other errors at service mesh. - Verify the service mesh logs to detect error details. - Check sidecar status of peer NF. Recovery: This alert clears automatically after 2 minutes if there is no service mesh failure observed by SCP with the same dimensions. For any assistance, contact My Oracle Support.

6.2.68 SCPHealthCheckFailedForPeerSCP

Table 6-71 SCPHealthCheckFailedForPeerSCP

Field	Details
Description	'SCP HealthCheck failed for peer SCP'
Summary	'SCP HealthCheck failed for peer SCP. namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{{ end }}'
Severity	Info
Condition	This alert is raised when peer SCP or inter-SCP becomes unhealthy due to health check status and outlier detection. ocscp_interscp_health_check_status_failed == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9023
Metric Used	ocscp_interscp_health_check_status_failed
Recommended Action	Cause: When peer SCP is unhealthy to receive any SBI message requests due to health check and outlier detection. Diagnostic Information: - Monitor if the overall average load of peer SCP is greater than the configured threshold value. - Check outlier detection. Recovery: This alert clears automatically if SCP-C decides SCP-P is healthy or available based on the current and previous status of outlier detection and health check. For any assistance, contact My Oracle Support.

6.2.69 SCPHealthCheckFailed

Table 6-72 SCPHealthCheckFailed

Field	Details
Description	'SCP HealthCheck failed'
Summary	'SCP HealthCheck failed. namespace: {{\$labels.kubernetes_namespace}}, podname: {{\$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{{ end }}'

Table 6-72 (Cont.) SCPHealthCheckFailed

Field	Details
Severity	Info
Condition	This alert is raised when SCP is unhealthy because the overall average load of SCP is greater than the configured threshold. ocscp_health_check_status_failed == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9024
Metric Used	ocscp_health_check_status_failed
Recommended Action	Cause: When SCP is unhealthy to receive any SBI message requests due to the overall average load. Diagnostic Information: Monitor if the overall average load of SCP is greater than the configured threshold value. Recovery: This alert clears automatically when the overall average load of SCP is less than the configured threshold value. For any assistance, contact My Oracle Support .

6.2.70 ScpWorkerPodPendingTransUtilizationAboveMinorThreshold

Table 6-73 ScpWorkerPodPendingTransUtilizationAboveMinorThreshold

Field	Details
Description	Worker Pending Transaction lead to minor level
Summary	'Worker Pending Transaction lead to minor level.namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when pending transaction utilization of SCP-Worker reaches MINOR level. ocscp_worker_pod_overload_control_pendingTrans_utilization_minor > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9014
Metric Used	ocscp_worker_pod_overload_control_pendingTrans_utilization_minor
Recommended Action	Cause: When pending transactions utilization of SCP-Worker reaches MINOR level. Diagnostic Information: - Monitor pending transactions usage while processing traffic. The threshold value of minor pending transaction utilization can be checked from database. - Check MINOR level logs of SCP-Worker for pending transaction status. Recovery: This alert clears automatically when pending transaction utilization is below MINOR threshold level. For any assistance, contact My Oracle Support .

6.2.71 ScpWorkerPodPendingTransUtilizationAboveMajorThreshold

Table 6-74 ScpWorkerPodPendingTransUtilizationAboveMajorThreshold

Field	Details
Description	Worker Pending Transaction lead to major level

Table 6-74 (Cont.) ScpWorkerPodPendingTransUtilizationAboveMajorThreshold

Field	Details
Summary	Worker Pending Transaction lead to major level. namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}
Severity	Major
Condition	This alert is raised when pending transaction utilization of SCP-Worker reaches MAJOR level. ocscp_worker_pod_overload_control_pendingTrans_utilization_major > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9015
Metric Used	ocscp_worker_pod_overload_control_pendingTrans_utilization_major
Recommended Action	Cause: When pending transactions utilization of SCP-Worker reaches MAJOR level. Diagnostic Information: - Monitor pending transactions usage while processing traffic. The threshold value of major pending transaction utilization can be checked from database. - Check MAJOR level logs of SCP-Worker for pending transaction status. Recovery: This alert clears automatically when pending transaction utilization is below MAJOR threshold level. For any assistance, contact My Oracle Support.

6.2.72 ScpWorkerPodPendingTransUtilizationAboveCriticalThreshold

Table 6-75 ScpWorkerPodPendingTransUtilizationAboveCriticalThreshold

Field	Details
Description	Worker Pending Transaction lead to critical level
Summary	'Worker Pending Transaction lead to critical level. namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}
Severity	Critical
Condition	This alert is raised when pending transaction utilization of SCP-Worker reaches CRITICAL level. ocscp_worker_pod_overload_control_pendingTrans_utilization_critical > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9016
Metric Used	ocscp_worker_pod_overload_control_pendingTrans_utilization_critical
Recommended Action	Cause: When pending transactions utilization of SCP-Worker reaches CRITICAL level. Diagnostic Information: - Monitor pending transactions usage while processing traffic. The threshold value of critical pending transaction utilization can be checked from database. - Check CRITICAL level logs of SCP-Worker for pending transaction status. Recovery: This alert clears automatically when pending transaction utilization is below CRITICAL threshold level. For any assistance, contact My Oracle Support.

6.2.73 ScpWorkerPodPendingTransUtilizationAboveWarnThreshold

Table 6-76 ScpWorkerPodPendingTransUtilizationAboveWarnThreshold

Field	Details
Description	Worker Pending Transaction lead to Warn level
Summary	'Worker Pending Transaction lead to warn level. namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Warn
Condition	This alert is raised when pending transaction utilization of SCP-Worker reaches WARN level. ocscp_worker_pod_overload_control_pendingTrans_utilization_warn > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9017
Metric Used	ocscp_worker_pod_overload_control_pendingTrans_utilization_warn
Recommended Action	Cause: When pending transactions utilization of SCP-Worker reaches WARN level. Diagnostic Information: - Monitor pending transactions usage while processing traffic. The threshold value of warn pending transaction utilization can be checked from database. - Check WARN level logs of SCP-Worker for pending transaction status. Recovery: This alert clears automatically when pending transaction utilization is below WARN threshold level. For any assistance, contact My Oracle Support.

6.2.74 ScpWorkerPodResourceUtilizationAboveMinorThreshold

Table 6-77 ScpWorkerPodResourceUtilizationAboveMinorThreshold

Field	Details
Description	Worker overload control lead to minor level
Summary	'Worker overload control lead to minor level.namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when overload control resource utilization of SCP-Worker reaches MINOR level. ocscp_worker_pod_overload_control_resource_utilization_minor > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9018
Metric Used	ocscp_worker_pod_overload_control_resource_utilization_minor
Recommended Action	Cause: When overload control resource utilization of SCP-Worker reaches MINOR level. Diagnostic Information: - Monitor pending transactions usage while processing traffic. The threshold value of minor overload control resource utilization can be checked from database. - Check MINOR level logs of SCP-Worker for overload control resource utilization status. Recovery: This alert clears automatically when overload control resource utilization is below MINOR threshold level. For any assistance, contact My Oracle Support.

6.2.75 ScpWorkerPodResourceUtilizationAboveMajorThreshold

Table 6-78 ScpWorkerPodResourceUtilizationAboveMajorThreshold

Field	Details
Description	Worker overload control lead to major level
Summary	'Worker overload control lead to major level. namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{ end }{'
Severity	Major
Condition	This alert is raised when overload control resource utilization of SCP-Worker reaches MAJOR level. ocscp_worker_pod_overload_control_resource_utilization_major > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9019
Metric Used	ocscp_worker_pod_overload_control_resource_utilization_major
Recommended Action	Cause: When overload control resource utilization of SCP-Worker reaches MAJOR level. Diagnostic Information: - Monitor pending transactions usage while processing traffic. The threshold value of major overload control resource utilization can be checked from database. - Check MAJOR level logs of SCP-Worker for overload control resource utilization status. Recovery: This alert clears automatically when overload control resource utilization is below MAJOR threshold level. For any assistance, contact My Oracle Support.

6.2.76 ScpWorkerPodResourceUtilizationAboveWarnThreshold

Table 6-79 ScpWorkerPodResourceUtilizationAboveWarnThreshold

Field	Details
Description	'Worker overload control lead to Warn level'
Summary	'Worker overload control lead to warn level. namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{ end }{'
Severity	Warning
Condition	This alert is raised when overload control resource utilization of SCP-Worker reaches WARN level. ocscp_worker_pod_overload_control_resource_utilization_warn > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9021
Metric Used	ocscp_worker_pod_overload_control_resource_utilization_warn

Table 6-79 (Cont.) ScpWorkerPodResourceUtilizationAboveWarnThreshold

Field	Details
Recommended Action	Cause: When overload control resource utilization of SCP-Worker reaches WARN level. Diagnostic Information: - Monitor pending transactions usage while processing traffic. The threshold value of warn overload control resource utilization can be checked from database. - Check WARN level logs of SCP-Worker for overload control resource utilization status. Recovery: This alert clears automatically when overload control resource utilization is below WARN threshold level. For any assistance, contact My Oracle Support.

6.2.77 ScpWorkerPodResourceUtilizationAboveCriticalThreshold

Table 6-80 ScpWorkerPodResourceUtilizationAboveCriticalThreshold

Field	Details
Description	Worker overload control lead to critical level
Summary	'Worker overload control lead to critical level. namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when overload control resource utilization of SCP-Worker reaches CRITICAL level. ocscp_worker_pod_overload_control_resource_utilization_critical > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9020
Metric Used	ocscp_worker_pod_overload_control_resource_utilization_critical
Recommended Action	Cause: When overload control resource utilization of SCP-Worker reaches CRITICAL level. Diagnostic Information: - Monitor pending transactions usage while processing traffic. The threshold value of critical overload control resource utilization can be checked from database. - Check CRITICAL level logs of SCP-Worker for overload control resource utilization status. Recovery: This alert clears automatically when overload control resource utilization is below CRITICAL threshold level. For any assistance, contact My Oracle Support.

6.2.78 SCPDNSSRVNRFMigrationTaskFailure

Table 6-81 SCPDNSSRVNRFMigrationTaskFailure

Field	Description
Severity	critical
Condition	ocscp_configuration_dnssrv_nrf_migration_task_failure == 1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15001

Table 6-81 (Cont.) SCPDNSSRVNRFMigrationTaskFailure

Field	Description
Description	An alert is raised to notify that migration from static to DNS has failed.
Recommended Actions	Cause: - If migration to DNS task was waiting for DNS SRV data and wait time elapsed. - If any migration task fails due to no acknowledgement from other microservices. - Wait time elapsed in task completion response for any migration task. Diagnostic Information: Monitor that all the DNS SRV configurations are proper and that all SCP pods are up and running in the proper state. Recovery: - DNS SRV data, wait time elapsed: Once data is received from DNS, the alert will be cleared. - No acknowledgement: Keep retrying for success acknowledgement and clear or remove the alert on receiving success acknowledgement. - Wait time elapsed for completion response: Resend the event and wait. Repeat this until a completion response is received, and on receiving a completion response, clear the alert. In all the raised conditions, the alert will also be cleared on receiving the new migration task. For any assistance, contact My Oracle Support.

6.2.79 SCPDNSSRVNRFNonMigrationTaskFailure

Table 6-82 SCPDNSSRVNRFNonMigrationTaskFailure

Field	Description
Severity	critical
Condition	ocscp_configuration_dnssrv_nrf_non_migration_task_failure == 1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15003
Description	An alert is raised to notify that the non-migrated task has failed.
Recommended Actions	Cause: - If a non-migrated task fails due to no acknowledgement from other microservices. - Wait time elapsed in task completion response. Diagnostic Information: Monitor that all the DNS SRV configurations are proper and that all SCP pods are up and running in the proper state. Recovery: - No acknowledgement: Keep retrying for success acknowledgement and clear or remove the alert on receiving success acknowledgement. - Wait time elapsed: When a new task is submitted for the same SPN, this alert is immediately cleared; if subsequent task processing fails, again, the alert will be raised. For any assistance, contact My Oracle Support.

6.2.80 SCPDNSSRVNRFDuplicateTargetDetected

Table 6-83 SCPDNSSRVNRFDuplicateTargetDetected

Field	Description
Severity	critical
Condition	ocscp_configuration_dnssrv_nrf_duplicate_target_detected == 1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15002
Description	An alert is raised to notify that a duplicate target NRF has been detected in the DNS SRV records.
Recommended Actions	Cause: This alert is raised when a duplicate target FQDN is received from the DNS SRV for different NRF SRV FQDN(s). In this case, the target FQDNs received against the first NRF SRV FQDN in the scpc-configuration service from the scpc-alternate-resolution service shall be processed, but the target FQDNs received against the subsequent NRF SRV FQDN will be ignored, and this alert shall be raised. Diagnostic Information: Monitor that all the DNS SRV configurations are proper and that all SCP pods are up and running in the proper state. Recovery: If for the same NRF SRV FQDN receives non-overlapping target FQDNs, then the alert will be cleared. For any assistance, contact My Oracle Support.

6.2.81 SCPHighResponseTimeFromProducer

Table 6-84 SCPHighResponseTimeFromProducer

Field	Description
Severity	Info
Condition	(sum(rate(ocscp_metric_upstream_service_time_total{ocscp_upstream_service_time="15000ms"}[2m])) by (kubernetes_namespace) + sum(rate(ocscp_metric_upstream_service_time_total{ocscp_upstream_service_time=">15000ms"}[2m])) by (kubernetes_namespace)) > 200
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15004
Description	It notifies when the traffic exceeds 200 messages per second and the response delay from the producer takes more than 10 seconds.
Recommended Actions	Cause: More than 200 messages per second have an upstream response time above 10 seconds. Diagnostic Information: Monitor metric metricocscp_metric_upstream_service_time_total with ocscp_upstream_service_time="15000ms" and ocscp_upstream_service_time=">15000ms". Recovery: An alert is cleared automatically when the number of responses with a response delay of more than 10 seconds falls below 200 messages per second. If Alert is not getting cleared, then check for any producer NFs or specific service request types that are taking more than 10 seconds to respond and take corrective actions if needed. Note that immediate action may not be needed, as this alert is informational. However, having too many requests with a long response delay may cause performance degradation at SCP. For any assistance, contact My Oracle Support.

6.2.82 SCPCGroupVersionDetectionFailed

Table 6-85 SCPCGroupVersionDetectionFailed

Field	Description
Severity	critical
Condition	ocscp_worker_cgroup_version_detection_failed == 1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15005
Description	Notify that cgroup version detection has failed.
Recommended Actions	Cause: SCP is unable to detect the cgroup version from the underlying kernel with the command "stat -fc %T /sys/fs/cgroup/." The possible expected value is either tmpfs or cgroup2fs. Diagnostic Information: - Option 1: Check worker error level logs for failure of cgroup version detection. - Option 2: - Login to the worker pod and run the command "stat -fc %T /sys/fs/cgroup/." - Verify whether it outputs either tmpfs or cgroup2fs. Recovery: - Make sure the cgroup has either tmpfs or cgroup2fs. - Do same version upgrade on SCP.. For any assistance, contact My Oracle Support.

6.2.83 SCPCPUUsageFileReadFailed

Table 6-86 SCPCPUUsageFileReadFailed

Field	Description
Severity	critical
Condition	ocscp_worker_cpu_usage_file_read_failed == 1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15006
Description	Notify that the CPU usage file read operation failed within the detected cgroup version.

Table 6-86 (Cont.) SCPCPUUsageFileReadFailed

Field	Description
Recommended Actions	Cause: SCP encountered a failure in performing a read operation for the CPU usage file within the detected cgroup version. The file path is determined based on the detected cgroup version. Diagnostic Information: - Option 1: Check worker warn level logs for failure in performing a read operation for the CPU usage file. - Option 2: - Log in to the worker pod and run the command "stat -fc %T /sys/fs/cgroup" to confirm whether it outputs either 'tmpfs' or 'cgroup2fs'. - Depending on the detected cgroup version, inspect the file located at /sys/fs/cgroup/cpu/cpuacct.usage for 'tmpfs' or /sys/fs/cgroup/cpu.stat for 'cgroup2fs'. - Ensure that the file exists at the specified path and possesses the required permissions for read operations. Recovery: - Verify that the files are appropriate according to the cgroup version and possess the necessary permissions for read operations. - Perform the same version upgrade on the SCP. For any assistance, contact My Oracle Support.

6.2.84 SCPIgnoreUnknownService

Table 6-87 SCPIgnoreUnknownService

Field	Description
Severity	Info
Condition	increase(ocscp_ignore_unknown_service_total[24h]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15000
Description	An alert is raised to notify that SCP ignored an unknown service in the NF profile.
Recommended Actions	Cause: SCP has received the NF profile with an unknown service and processed the profile by ignoring this unknown service. Diagnostic Information: Check the received NF profile for the unknown services. Recovery: If the unknown services are not present in the NF profile in the next scrapping interval, then the alert will be cleared. For any assistance, contact My Oracle Support.

6.2.85 SCPWorkerSSLCertificateOnCriticalExpiry

Table 6-88 SCPWorkerSSLCertificateOnCriticalExpiry

Field	Description
Severity	Critical
Condition	ocscp_metric_ssl_certificate_expire_total == 1
OID used for SNMP Traps	11.3.6.1.4.1.323.5.3.35.1.2.15010

Table 6-88 (Cont.) SCPWorkerSSLCertificateOnCriticalExpiry

Field	Description
Description	An alert is raised whenever the SCP SSL certificate is about to expire, based on the configured threshold values.
Recommended Actions	Cause: The SSL certificate expiration is approaching the configured critical expiry time. Diagnostic Information: - Whenever this alert is raised, it indicates that the SSL certificates configured for SCP are about to expire. - Verify the certificate expiry date. Recovery: The SCP SSL secret needs to be updated with renewed SSL certificates. For any assistance, contact My Oracle Support.

6.2.86 SCPWorkerSSLCertificateOnMajorExpiry

Table 6-89 SCPWorkerSSLCertificateOnMajorExpiry

Field	Description
Severity	major
Condition	ocscp_metric_ssl_certificate_expire_total == 2
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15011
Description	An alert is raised whenever the SCP SSL certificate is about to expire, based on the configured threshold values.
Recommended Actions	Cause: The SSL certificate expiration is approaching the configured major expiry time. Diagnostic Information: - Whenever this alert is raised, it indicates that the SSL certificates configured for SCP are about to expire. - Verify the certificate expiry date. Recovery: The SCP SSL secret needs to be updated with renewed SSL certificates. For any assistance, contact My Oracle Support.

6.2.87 SCPWorkerSSLCertificateOnMinorExpiry

Table 6-90 SCPWorkerSSLCertificateOnMinorExpiry

Field	Description
Severity	minor
Condition	ocscp_metric_ssl_certificate_expire_total == 3
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15012
Description	An alert is raised whenever the SCP SSL certificate is about to expire, based on the configured threshold values.

Table 6-90 (Cont.) SCPWorkerSSLCertificateOnMinorExpiry

Field	Description
Recommended Actions	Cause: The SSL certificate expiration is approaching the configured minor expiry time. Diagnostic Information: - Whenever this alert is raised, it indicates that the SSL certificates configured for SCP are about to expire. - Verify the certificate expiry date. Recovery: The SCP SSL secret needs to be updated with renewed SSL certificates. For any assistance, contact My Oracle Support.

6.2.88 SCPIngressConnectionEstablishmentFailure

Table 6-91 SCPIngressConnectionEstablishmentFailure

Field	Description
Severity	info
Condition	increase(ocscp_worker_https_ingress_connection_failure_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15007
Description	An alert is raised whenever any ingress HTTPS connection is failed.
Recommended Actions	Cause: Whenever an Ingress HTTPS connection establishment fails. Diagnostic Information: This alert may be raised if any connection establishment fails or if a handshake fails. Recovery: The alert will be cleared if there are no ingress HTTPS connection failures in the next scrape interval (2 minutes). For any assistance, contact My Oracle Support.

6.2.89 SCPEgressConnectionEstablishmentFailure

Table 6-92 SCPEgressConnectionEstablishmentFailure

Field	Description
Severity	info
Condition	increase(ocscp_worker_https_egress_connection_failure_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15008
Description	An alert is raised whenever any egress HTTPS connection is failed.
Recommended Actions	Cause: Whenever an egress HTTPS connection establishment fails to send the request to producer NFs. Diagnostic Information: This alert may be raised if any connection establishment fails or if a handshake fails. Recovery: The alert will be cleared if there are no engress HTTPS connection failures in the next scrape interval (2 minutes). For any assistance, contact My Oracle Support.

6.2.90 SCPNrfProxyOauthQueuesUtilizationAboveCriticalThreshold

Table 6-93 SCPNrfProxyOauthQueuesUtilizationAboveCriticalThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}: SCP NrfProxy Oauth Queues Utilization Above Critical Threshold'
Summary	'SCP NrfProxy Oauth Queues Utilization Above Critical Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when SCP NrfProxy Oauth Queues Utilization is above the critical threshold. ocscp_nrfproxy_oauth_queue_alert{severity="CRITICAL"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.14000
Metric Used	ocscp_nrfproxy_oauth_queue_alert
Recommended Action	Cause: When NrfProxy Task queues are filled, the traffic exceeds the limit. Diagnostic Information: - Monitor the traffic toward nrfProxy to pod using the Grafana dashboard. - Refer to the rate of ocscp_nrfproxy_oauth_queue_alert metric on the Grafana dashboard. Recovery: This alert clears automatically when the traffic decreases below the critical threshold. For any assistance, contact My Oracle Support.

6.2.91 SCPNrfProxyOauthQueuesUtilizationAboveMajorThreshold

Table 6-94 SCPNrfProxyOauthQueuesUtilizationAboveMajorThreshold

Field	Details
Description	'instancename: {{\$labels.instance}}, namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}: SCP NrfProxyOauth Queues Utilization Above Major Threshold'
Summary	'SCP NrfProxy Oauth Queues Utilization Above Major Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when SCP NrfProxy Oauth Queues Utilization is above the major threshold. ocscp_nrfproxy_oauth_queue_alert{severity="MAJOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.14001
Metric Used	ocscp_nrfproxy_oauth_queue_alert
Recommended Action	Cause: When NrfProxy Task queues are filled, the traffic exceeds the limit. Diagnostic Information: - Monitor the traffic toward nrfProxy to pod using the Grafana dashboard. - Refer to the rate of ocscp_nrfproxy_oauth_queue_alert metric on the Grafana dashboard. Recovery: This alert clears automatically when the traffic decreases below the Major threshold. For any assistance, contact My Oracle Support.

6.2.92 SCPNrfProxyOAuthQueuesUtilizationAboveMinorThreshold

Table 6-95 SCPNrfProxyOAuthQueuesUtilizationAboveMinorThreshold

Field	Details
Description	description: 'instancename: {{\$labels.instance}}, namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}: SCP NrfProxyOAuth Queues Utilization Above Minor Threshold'
Summary	'SCP NrfProxy OAuth Queues Utilization Above Minor Threshold, instancename: {{\$labels.instance}}, namespace: {{\$labels.namespace}}, podname: {{\$labels.pod}}, timestamp: {{ with query "time()" }}{ . first value humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when SCP NrfProxy OAuth Queues Utilization is above the minor threshold. ocscp_nrfproxy_oauth_queue_alert{severity="MINOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.14002
Metric Used	ocscp_nrfproxy_oauth_queue_alert
Recommended Action	Cause: When NrfProxy Task queues are filled, the traffic exceeds the limit. Diagnostic Information: • Monitor the traffic toward nrfProxy to pod using the Grafana dashboard. • Refer to the rate of ocscp_nrfproxy_oauth_queue_alert metric on the Grafana dashboard. Recovery: This alert clears automatically when the traffic decreases below the Major threshold. For any assistance, contact My Oracle Support.

6.3 Configuring Alerts

This section lists the configuring alerts.

6.3.1 Applying Alerts Rule to CNE without Prometheus Operator

SCP Helm Chart Release Name: `_NAME_`

Prometheus NameSpace: `_Namespace_`

Perform the following procedure to configure Service Communication Proxy alerts in Prometheus.

1. Run the following command to check the name of the config map used by Prometheus:

```
$kubectl get configmap -n <_Namespace_>
```

Example:

```
$kubectl get configmap -n prometheus-alert2
NAME                                DATA  AGE
lisa-prometheus-alert2-alertmanager 1      146d
lisa-prometheus-alert2-server        4      146d
```

2. Take a backup of the current config map of Prometheus. This command saves the configmap in the provided file. In the following command, the configmap is stored in the `/tmp/tempConfig.yaml` file:

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

Example:

```
$ kubectl get configmaps lisa-prometheus-alert2-server -o yaml -n prometheus-alert2 > /tmp/tempConfig.yaml
```

3. Check and delete the "alertsscp" rule if it has already configured in the prometheus config map. If configured, this step removes the " alertsscp " rule. This is an optional step if configuring the alerts for the first time.

```
$ sed -i '/etc/config/alertsscp/d' /tmp/tempConfig.yaml
```

4. Add the "alertsscp" rule in the configmap dump file under the ' rule_files ' tag.

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

5. Update the configmap using below command. Ensure to use the same configmap name that was used to take a backup of the prometheus configmap.

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

Example:

```
$ kubectl replace configmap lisa-prometheus-alert2-server -f /tmp/tempConfig.yaml
```

6. Run the following command to patch the configmap with a new "alertsscp" rule:

Note

The patch file provided is the `ocscp_csar_23_2_0_0_0.zip` folder provided with SCP, that is, `SCPAlertrules.yaml`.

```
$ kubectl patch configmap _NAME_-server -n _Namespace_ --type merge --patch "$(cat ~/SCPAlertrules.yaml)"
```

Example:

```
$ kubectl replace configmap lisa-prometheus-alert2-server -f /tmp/tempConfig.yaml
```

Note

Prometheus takes about 20 seconds to apply the updated Config map.

6.3.2 Applying Alerts Rule to CNE with Prometheus Operator

Perform the following procedure to apply alerts rule to Cloud Native Environment (CNE) with Prometheus Operator (CNE 1.9.0 and later).

- Run the following command to apply SCP alerts file to create Prometheus rules Custom Resource Definition (CRD):

```
kubectl apply -f <file_name> -n <scp namespace>
```

Where,

- <file_name> is the SCP alerts file.
- <scp namespace> is the SCP namespace.

Example:

```
kubectl apply -f ocscp_alerting_rules_promha_24.2.5.yaml -n scpsvc
```

Sample file delivered with SCP package:

```
ocscp_alerting_rules_promha_24.2.5.yaml
```

6.3.3 Configuring Service Communication Proxy Alert using the SCPAlertrules.yaml file

Note

Default NameSpace is **scpsvc** for Service Communication Proxy. You can update the NameSpace as per the deployment.

To access the scpAlertsrules_<scp release number>.yaml file from the Scripts folder of ocscp_csar_23_2_0_0_0.zip, download the SCP package from [My Oracle Support](#) as described in "Downloading the SCP Package " in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Alerts Details

Description and summary for alerts are added by the Prometheus alert manager.

Alerts are supported for three different resources/routing crosses threshold.

- SCPIngress Traffic Rate Above Threshold
 - Has three threshold level Minor (above 1400 mps to 2000mps), Major (1600 to 1800 mps), Critical (above 1800 mps). These values are configurable.

- In the description, information is presented similar to: "Ingress Traffic Rate at Locality: <Locality of scp> is above <threshold level (minor/major/critical)> threshold (i.e. <value of threshold>)"
- In Summary: "Namespace: <Namespace of scp deployment that Locality>, Pod: <SCP-worker Pod name>: Current Ingress Traffic Rate is <Current rate of Ingress traffic > mps which is above 70 Percent of Max MPS(<upper limit of ingress traffic rate per pod>)"

Note

Ingress traffic rate is per scp-worker pod in a namespace at particular SCP-Locality. Currently, 2000mps is the upper limit for per scp-worker pod.

- SCP Routing Failed For Service
 - It alerts for which NF Service Type and NF Type at particular locality, Routing failed
 - Description: "Routing failed for service"
 - Summary: "Routing failed for service: NFService Type = <Message NF Service Type>, NFType = <Message NF Type>, Locality = <SCP Locality where Routing Failed> and value = <Accumulated failure till now, of such message for NFType and NFService Type>"

Note

The value field currently does not provide the number of failures in particular time interval, instead it provides the total number of Routing failures.

- SCP Pod Memory Usage: Type of alert is SCPWorkerPodMemoryUsage.
 - Pod memory usage for SCP Pods (Soothsayer and Worker) deployed at a particular node instance is provided.
 - The Soothsayer pod threshold is 8 GB
 - The Worker pod threshold is 4 GB
 - Summary: Instance: "<Node Instance name>, NameSpace: <Namespace of SCP deployment>, Pod: <(Soothsayer/Worker) Pod name>: <Soothsayer/Worker> Pod High Memory usage detected"
 - Summary: "Instance: "<Node Instance name>, Namespace: <Namespace of SCP deployment>, Pod: <(Soothsayer/Worker) Pod name>: Memory usage is above <threshold value>G (current value is: <current value of memory usage>)"

6.3.4 Configuring Alert Manager for SNMP Notifier

Grouping of alerts is based on:

- podname
- alertname
- severity
- namespace
- nfServiceType

- nfServiceInstanceId

User needs to add subroutes for SCP alerts in AlertManager config map as below:

1. Take a backup of the current config map of Alertmanager by running the following command:

```
kubectl get configmaps <NAME-alertmanager> -oyaml -n <Namespace> > /tmp/
bkupAlertManagerConfig.yaml
```

Example:

```
kubectl get configmaps occne-prometheus-alertmanager -oyaml -n occne-infra
> /tmp/bkupAlertManagerConfig.yaml
```

2. Edit Configmap to add subroute for SCP Trap OID:

```
kubectl edit configmaps <NAME-alertmanager> -n <Namespace>
```

Example:

```
kubectl edit configmaps occne-prometheus-alertmanager -n occne-infra
```

3. Add the subroute under 'route' in configmap:

```
routes:
  - receiver: default-receiver
    group_interval: 1m
    group_wait: 10s
    repeat_interval: 9y
    group_by: [podname, alertname, severity, namespace, nfservicetype,
nfserviceinstanceid, servingscope, nftype]
    match_re:
      oid: ^1.3.6.1.4.1.323.5.3.35.(.*)
```

MIB Files for SCP

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.

- ocscp_mib_tc_24.2.5.mib: This is considered as SCP top level mib file, where the Objects and their data types are defined.
- ocscp_mib_24.2.5.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.

Note

MIB files are packaged with ocscp_csar_23_2_0_0_0.zip. You can download the file from [MOS](#) as described in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

6.4 Configuring SCP Alerts for OCI

To configure SCP alerts for OCI, OCI supports metric expressions written in MQL (Metric Query Language) and therefore requires `ocscp_oci_alertrules_24.2.5.zip` file for configuring alerts in OCI observability platform. For more information, see *Oracle Communications Cloud Native Core, OCI Adaptor Deployment Guide*.

7

KPIs and Traces

This chapter provides information about newly introduced and updated Key Performance Indicators (KPIs) and Traces.

7.1 Key Performance Indicators

This section provides information about Key Performance Indicators (KPIs) for Service Communication Proxy (SCP).

7.1.1 Platform KPIs

The following table provides information on Platform metrics.

Table 7-1 SCP CPU Usage

Field	Details
Description	The total SCP CPU utilization per pod.
Expression	<code>sum(rate(container_cpu_usage_seconds_total{image!="",namespace="ocscp"}[2m])) by (pod,container)</code>

Table 7-2 SCP Memory Usage

Field	Details
Description	The total SCP memory utilization per pod.
Expression	<code>(avg_over_time(container_memory_working_set_bytes{namespace="ocscp",container=~"istio-proxy scp.*"}[2m]))</code>

Table 7-3 SCP Memory Usage Percentage

Field	Details
Description	The total percentage of SCP memory utilization per pod.
Expression	<code>round(100 * avg_over_time(container_memory_working_set_bytes{image!="",container_name!="POD",namespace="ocscp", container=~"scp-nrfproxy"}[2m]) / (8*1073741824))</code> Note: Here, scp-nrfproxy is taken as an example; the same holds true for all the other microservices. The hardcoded memory value of 8GB (for nrfproxy) must be replaced in the Grafana dashboard file with the value that is assigned in the deployment environment, and this holds true for all the other microservices.

Table 7-4 SCP CPU Usage Percentage

Field	Details
Description	The total percentage of SCP CPU utilization per pod.

Table 7-4 (Cont.) SCP CPU Usage Percentage

Field	Details
Expression	<code>sum(rate(container_cpu_usage_seconds_total{image!="",namespace="ocscp",container="scpc-audit"}[2m])) by (pod,container)*100/3</code> Note: Here, scp-worker is taken as an example; the same holds true for all the other microservices. The hardcoded CPU value of 8 (for scp-worker) must be replaced in the Grafana dashboard file with the value that is assigned in the deployment environment, and this holds true for all the other microservices.

Table 7-5 CPU Usage

Field	Details
Description	The total percentage of SCP CPU utilization by master and worker pod.
Expression	<code>sum (rate (container_cpu_usage_seconds_total{image!="",node=~"k8s.*",namespace="ocscp"}[2m])) by (node)</code>

Table 7-6 Memory Usage

Field	Details
Description	The total percentage of SCP memory utilization by master and worker pod.
Expression	<code>sum (container_memory_working_set_bytes{id="/",node=~"k8s.*",namespace="ocscp"}) by (node)</code>

Table 7-7 Prometheus CPU Utilization

Field	Details
Description	CPU utilization of Prometheus pods.
Expression	<code>sum(rate(container_cpu_usage_seconds_total{pod="ocne-prometheus-server-5f48d747b8-lx4zn",container="prometheus-server",namespace="ocscp"} [5m]))</code>

Table 7-8 Prometheus Memory Utilization

Field	Details
Description	Memory Utilization of Prometheus Pods.
Expression	<code>container_memory_usage_bytes{container="prometheus-server",namespace="ocscp"}</code>

Table 7-9 Samples_scraped_time

Field	Details
Description	Total number of Metrics samples scraped by Prometheus in give time.
Expression	<code>avg_over_time(scrape_samples_scraped{namespace="ocscp"}[10m])</code>

7.1.2 Application KPIs

The following table provides information on Application KPIs.

Table 7-10 Ingress Request Rate

Field	Details
Description	Total number of ingress requests sent to SCP by consumer NFs.
Expression	sum(irate(ocscp_metric_http_rx_req_total{namespace="ocscp", ocscp_consumer_host!="OCSCP-INITIATED-MSG"}[5m])) by (ocscp_nf_type,ocscp_service_instance_id)

Table 7-11 Egress Request Rate

Field	Details
Description	Total number of egress requests sent to producer NFs by SCP.
Expression	sum(irate(ocscp_metric_http_tx_req_total{namespace="ocscp"}[2m])) by (ocscp_nf_type)

Table 7-12 Ingress Response Rate

Field	Details
Description	Total number of ingress responses sent to SCP by producer NFs.
Expression	sum(irate(ocscp_metric_http_rx_res_total{namespace="ocscp"}[2m])) by (ocscp_response_code_class, ocscp_producer_host)

Table 7-13 Egress Response Rate

Field	Details
Description	Total number of egress responses sent to consumer NFs by SCP.
Expression	sum(irate(ocscp_metric_http_tx_res_total{namespace="ocscp", ocscp_consumer_host!="OCSCP-INITIATED-MSG"}[2m]))

Table 7-14 Response Processing Time

Field	Details
Description	The rate of processing time by SCP for ingress responses into the time buckets.
Expression	sum(irate(ocscp_metric_response_processing_time_total{namespace="ocscp"}[2m]))by(ocscp_processing_time)

Table 7-15 Request Processing Time

Field	Details
Description	The rate of processing time by SCP for ingress requests into the time buckets.
Expression	sum(irate(ocscp_metric_request_processing_time_total{namespace="ocscp"}[2m]))by(ocscp_processing_time)

Table 7-16 Transaction Success Rate

Field	Details
Description	The rate of successful transaction response sent by SCP for a request.

Table 7-16 (Cont.) Transaction Success Rate

Field	Details
Expression	$\frac{((\text{sum}(\text{irate}(\text{ocscp\_metric\_http\_tx\_res\_total}\{\text{ocscp\_response\_code\_class}="2", \text{kubernetes\_namespace}=\sim "\$namespace"\}[2m])))/(\text{sum}(\text{irate}(\text{ocscp\_metric\_http\_tx\_res\_total}\{\text{kubernetes\_namespace}=\sim "\$namespace"\}[2m]))))}{[2m])})*100$

Table 7-17 Response Code Rate

Field	Details
Description	The rate of different response codes class on SCP.
Expression	$\text{sum}(\text{irate}(\text{ocscp\_metric\_http\_rx\_res\_total}\{\text{kubernetes\_namespace}="\$namespace"\}[2m])) \text{ by } (\text{ocscp\_response\_code\_class})$

Table 7-18 scp-worker replicas

Field	Details
Description	The total number of ready replicas for the scp-worker service.
Expression	$\text{kube\_replicaset\_status\_ready\_replicas}\{\text{replicaset}=\sim '.*\text{scp-worker}.*', \text{namespace}="\$namespace"\}$

Table 7-19 NF Notifications Success

Field	Details
Description	The total number of notifications successfully processed by SCP.
Expression	$\text{sum}(\text{ocscp\_notification\_nf\_profile\_processing\_success\_total})$

Table 7-20 NF Notifications Failures

Field	Details
Description	The total number of notifications failed to be processed by SCP.
Expression	$\text{sum}(\text{ocscp\_notification\_nf\_profile\_processing\_failure\_total})$

Table 7-21 Audit Requests Total

Field	Details
Description	The total number of audit requests sent by SCP.
Expression	$\text{sum}(\text{ocscp\_audit\_tx\_req\_total}\{\text{namespace}="ocscp"\})$

Table 7-22 Audit Responses Total

Field	Details
Description	The total number of audit responses received by SCP.
Expression	$\text{sum}(\text{ocscp\_audit\_rx\_res\_total}\{\text{namespace}="ocscp"\})$

Table 7-23 NF Subscription Success

Field	Details
Description	The total number of successful subscriptions on SCP per NF type.
Expression	sum(ocscp_subscription_nf_success_total)

Table 7-24 NF Subscription Failures

Field	Details
Description	The total number of failed subscriptions on SCP per NF type.
Expression	sum(ocscp_subscription_nf_failure_total)

Table 7-25 NRF Registration Success

Field	Details
Description	The total number of successful NRF registrations on SCP.
Expression	sum(ocscp_subscription_nrf_registration_success_total)

Table 7-26 NRF Registration Failure

Field	Details
Description	The total number of failed NRF registrations on SCP.
Expression	sum(ocscp_subscription_nrf_registration_failure_total)

Table 7-27 NRF Heartbeat Success

Field	Details
Description	The total number of successful NRF Heartbeats on SCP.
Expression	sum(ocscp_subscription_nrf_heartbeat_success_total{namespace="ocscp"})

Table 7-28 NRF Heartbeat Failures

Field	Details
Description	The total number of failed NRF Heartbeats on SCP.
Expression	sum(ocscp_subscription_nrf_heartbeat_failures_total{namespace="ocscp"})

7.1.3 Feature KPIs

The following table provides information on Feature KPIs.

Table 7-29 Access Token Count in Response

Field	Details
Description	The rate of responses contains the 3gpp-Sbi-Access-Token header.

Table 7-29 (Cont.) Access Token Count in Response

Field	Details
Expression	sum(irate(ocscp_worker_access_token_forwarded_in_tx_res_total{namespace="ocscp"}[2m]))

Table 7-30 Received Response count for nnrf-oauth2 service

Field	Details
Description	Rate of nnrf-oaut2 service responses from NRF.
Expression	sum(irate(ocscp_metric_http_rx_res_total{namespace="ocscp",ocscp_nf_service_type="nnrf-oauth2",ocscp_response_code="200"}[2m]))by(pod)

Table 7-31 Coherence Cache Size for AccessToken

Field	Details
Description	Number of access tokens in the cache.
Expression	coherence_cache_size{namespace="ocscp",name="accessTokenMap"}

Table 7-32 Access Token Count in Transmitted Request

Field	Details
Description	Rate of SCP sending the 3gpp-Sbi-Access-Token header in the request.
Expression	sum(irate(ocscp_worker_access_token_forwarded_in_tx_req_total{namespace="ocscp"}[2m]))

Table 7-33 SCP Worker - Discovery Tx Requests Rate to NRF Proxy

Field	Details
Description	Rate of worker sends an on-demand discovery request to nrfproxy.
Expression	sum(irate(ocscp_metric_ondemand_discovery_tx_req_total{namespace="ocscp"}[2m]))

Table 7-34 SCP Worker - Discovery Rx Response Rate from NRF Proxy

Field	Details
Description	Rate of worker receives on-demand discovery response from nrfproxy.
Expression	sum(irate(ocscp_metric_ondemand_discovery_rx_res_total{namespace="ocscp"}[2m]))

Table 7-35 NRF Proxy - Discovery Tx Requests Rate to NRF

Field	Details
Description	Rate of discovery requests sent to the NRF by nrfproxy.
Expression	sum(irate(ocscp_nrf_nf_discovery_tx_req_total{namespace="ocscp"}[2m]))by(pod)

Table 7-36 NRF Proxy - Discovery Rx Response Rate from NRF

Field	Details
Description	Rate of discovery responses received on nrfproxy from NRF.
Expression	sum(irate(ocscp_nrf_nf_discovery_rx_res_total{namespace="ocscp"}[2m]))by(pod)

Table 7-37 SCP-Worker - Local Discovery Requests Resolution Rate(Cache Hits)

Field	Details
Description	Rate of worker was able to resolve discovery request using local cache.
Expression	sum(irate(ocscp_metric_ondemand_discovery_cache_match_total{namespace="ocscp"}[2m]))by(pod)

Table 7-38 NRF Proxy Generated Error response Rate

Field	Details
Description	Rate of error responses generated by nrfproxy (excluding nrf-returned errors).
Expression	sum(irate(ocscp_nrfproxy_generated_error_response_total{namespace="ocscp"}[2m]))by(pod)

Table 7-39 SCP LCI Tx Rate

Field	Details
Description	Rate of messages forwarded with SCP's LCI header.
Expression	sum(irate(ocscp_metric_scp_lci_tx_total{namespace=~"ocscp"}[2m]))

Table 7-40 Per NF Load

Field	Details
Description	The current load of peer NF.
Expression	ocscp_nf_load{namespace=~"ocscp"}

Table 7-41 Peer LCI Rx Rate

Field	Details
Description	Rate of messages received with NF's LCI header.
Expression	sum(irate(ocscp_metric_nf_lci_rx_total{namespace=~"ocscp"}[2m]))

Table 7-42 Peer LCI Tx Rate

Field	Details
Description	Rate of messages forwarded with NF's LCI header.
Expression	sum(irate(ocscp_metric_nf_lci_tx_total{namespace=~"ocscp"}[2m]))

Table 7-43 Ingress Rate Limit - Request Throttle Rate

Field	Details
Description	Number of requests throttled due to ingress rate limitations.
Expression	sum(irate(ocscp_metric_ingress_rate_limiting_throttle_req_total{namespace="ocscp"}[2m]))by(ocscp_consumer_fqdn, ocscp_configured_ingress_rate, ocscp_ingress_rl_bucketkey)

Table 7-44 Ingress Rate Limit - Rate Limit Not Applied Rate

Field	Details
Description	Number of requests routed without ingress rate limiting treatment.
Expression	sum(irate(ocscp_metric_ingress_rate_limiting_not_applied_req_total{namespace="ocscp"}[2m]))by(ocscp_consumer_fqdn, ocscp_cause)

Table 7-45 Egress Rate Limit - Request Throttle Rate

Field	Details
Description	Number of requests throttled due to egress rate limiting.
Expression	sum(irate(ocscp_metric_egress_rate_limiting_throttle_req_total{namespace="ocscp"}[2m]))by(ocscp_nf_type, ocscp_producer_host, ocscp_nf_service_type, ocscp_configured_egress_rate, ocscp_egress_rl_bucketkey, ocscp_nf_instance_id)

Table 7-46 Egress Rate Limit - Rate Limit Not Applied Rate

Field	Details
Description	Number of requests routed without egress rate limiting treatment.
Expression	sum(irate(ocscp_metric_egress_rate_limiting_not_applied_req_total{namespace="ocscp"}[2m]))by(ocscp_nf_type, ocscp_producer_host, ocscp_cause)

Table 7-47 SCP Cache Size

Field	Details
Description	Coherence cache size for different caches.
Expression	coherence_cache_size

Table 7-48 SCP Cache Hits

Field	Details
Description	Number of coherence cache hits.
Expression	sum(irate(coherence_cache_hits{namespace="ocscp"}[2m])) by (name)

Table 7-49 Message Request Rate Per Rate Limit Bucket Key

Field	Details
Description	Rate of request received per rate limit key.

Table 7-49 (Cont.) Message Request Rate Per Rate Limit Bucket Key

Field	Details
Expression	sum(irate(ocscp_metric_message_current_request_rate_total{namespace="ocscp"}[2m]))by (bucketKey)

Table 7-50 SCP OCI Conveyance Rate

Field	Details
Description	Number of messages with SCP's OCI conveyed to peers.
Expression	sum(irate(ocscp_worker_oci_conveyance_total{namespace=~"ocscp"}[2m]))

Table 7-51 Peer OCI Conveyance Rate

Field	Details
Description	Number of messages with a peer's OCI conveyed to peers.
Expression	sum(irate(ocscp_worker_peer_oci_conveyance_total{namespace=~"ocscp"}[2m]))

Table 7-52 OCI Enforced Rate

Field	Details
Description	Number of messages on which OCI is enforced that are either rerouted to alternate NFs or an error response is sent.
Expression	sum(irate(ocscp_worker_oci_enforced_total{namespace=~"ocscp"}[2m]))

Table 7-53 OCI Unsupported Rx Rate

Field	Details
Description	Number of messages received with unsupported scope.
Expression	sum(irate(ocscp_worker_rx_unsupported_oci_scope_total{namespace=~"ocscp"}[2m]))

Table 7-54 SCP Worker - Mediation Tx Requests Rate

Field	Details
Description	The total number of messages that forwarded to the mediation service from SCP.
Expression	sum(irate(ocscp_mediation_http_tx_total{namespace="ocscp"}[2m])) by (pod)

Table 7-55 SCP Worker -Mediation Rx Response Rate

Field	Details
Description	The total number of messages that the mediation service sends back as a response to SCP.
Expression	sum(irate(ocscp_mediation_http_rx_total{namespace="ocscp"}[2m])) by (pod)

Table 7-56 SCP Worker - Mediation Upstream Service Time

Field	Details
Description	This indicates the total time taken by the mediation to process the request.
Expression	sum(irate(ocscp_mediation_upstream_service_time_ms_total{namespace="ocscp"}[2m])) by (ocscp_med_upstream_service_time)

7.2 Traces

The following table provides information of traces for SCP.

Table 7-57 SCP Traces

Field Name	Description
component	The software package, framework, library, or module that generated the associated Span. Request or Response Type : common
node_id	Local information. Request or Response Type : common
method	HTTP method of the request for the associated Span. Example: "GET", "POST". Request or Response Type : request,common
scheme	Url scheme is http. Request or Response Type : request
authority	Authority give you details about registered name or server address, along with optional port and user information. Request or Response Type : request
path	A path consists of a sequence of path segments separated by a slash ("/") character. Request or Response Type : request
3gpp-sbi-message-priority	This header shall be included in HTTP/2 messages when a priority for the message needs to be conveyed. Request or Response Type : request
x-forwarded-for	To identify the originating IP address of a client. Request or Response Type : request
x-forwarded-proto	To determine the protocol used between the client and SCP. Request or Response Type : request
x-envoy-internal	service wants to know whether a request is internal origin or not. Request or Response Type : request

Table 7-57 (Cont.) SCP Traces

Field Name	Description
via	It is used for tracking message forwards, avoiding request loops, and identifying the protocol capabilities of senders along the request/response chain. Request or Response Type : request
x-request-id	The x-request-idheader is used to uniquely identify a request as well as perform stable access logging and tracing. Request or Response Type : common
payload	http request body. Request or Response Type : request, common
status	To determine the http request has been succeeded or not. Request or Response Type : response
content-type	The Content-Type entity header is used to indicate the media type of the resource. Request or Response Type : response
content-length	The Content-Length header indicates the size of the entity body in the message, in bytes. Request or Response Type : response
server	The Server response-header field contains information about the software used by the origin server to handle the request. Request or Response Type : response
date	The time and date, when the request is processed. Request or Response Type : response
x-envoy-upstream-service-time	Contains the time in milliseconds spent by the upstream host processing the request. Request or Response Type : response
location	To provide information about the location of a newly created resource. Request or Response Type : response
payload	http response body. Request or Response Type : response
http.url	Specifies the request's URL. Request or Response Type : request, common
downstream_cluster	A downstream host connects to Envoy, sends requests, and receives responses. Request or Response Type : common

Table 7-57 (Cont.) SCP Traces

Field Name	Description
user_agent	When your browser connects to a website, it includes a User-Agent field in its HTTP header. Request or Response Type : request, common
http.protocol	The communication between client and server over HTTP/2. Request or Response Type : common
request_size	HTTP header size. Request or Response Type : common
upstream_cluster	An upstream host receives connections and requests from Envoy and returns responses. Request or Response Type : common
http.status_code	HTTP response status code for the associated Span. Example: 200, 503, 404 Request or Response Type : common
response_size	HTTP header size. Request or Response Type : common
response_flag	Additional details about the response or connection, if any. Request or Response Type : common
span.kind	Specifies the role of the span in a RPC communication. In case of HTTP communication it is searching client and server values for this tag. Request or Response Type : common
error	True, if and only if, the application considers the operation represented by the Span to have failed. Request or Response Type : common
x-request-id	The x-request-id header is used by Envoy to uniquely identify a request as well as perform stable access logging and tracing Request or Response Type : request
x-b3-traceid	The x-b3-traceid HTTP header is used by the Zipkin tracer in Envoy. The TraceId is 64-bit in length and indicates the overall ID of the trace. Every span in a trace shares this ID. Request or Response Type : request
x-b3-spanid	The x-b3-spanid HTTP header is used by the Zipkin tracer in Envoy. The SpanId is 64-bit in length and indicates the position of the current operation in the trace tree. Request or Response Type : request

Table 7-57 (Cont.) SCP Traces

Field Name	Description
x-b3-sampled	The x-b3-sampledHTTP header is used by the Zipkin tracer in Envoy. When the sampled flag is either not specified or set to 1, the span will be reported to the tracing system. Request or Response Type : request
ueidentitytype	NF type. Request or Response Type : request
ueidentityvalue	SUPI range for NF type. Request or Response Type : request
x-envoy-expected-rq-timeout-ms	This is the time in milliseconds the router expects the request to be completed. Request or Response Type : request

8

SCP Interactions with NRF

SCP allows configuration of multiple NRF sets and determines the network topology with all the configured NRF sets.

SCP can register with an NRF instance as per priority and capacity in each configured NRF set. You can configure an SCP profile at the time of deployment. SCP sends NF details for heartbeat with NRF and updates the mate SCP information in the NF profile after the deployment.

SCP subscribes to any NF status change notification for all the 5G NF types with one of the available NRF instances as per priority and capacity in each configured NRF set using NFStatusSubscribe, NFStatusUnsubscribe, and NFStatusNotify service operations of the Nnrf_NFManagement service. NF suspended and deregistered notification removes the detected NF information and updates the routing information as required.

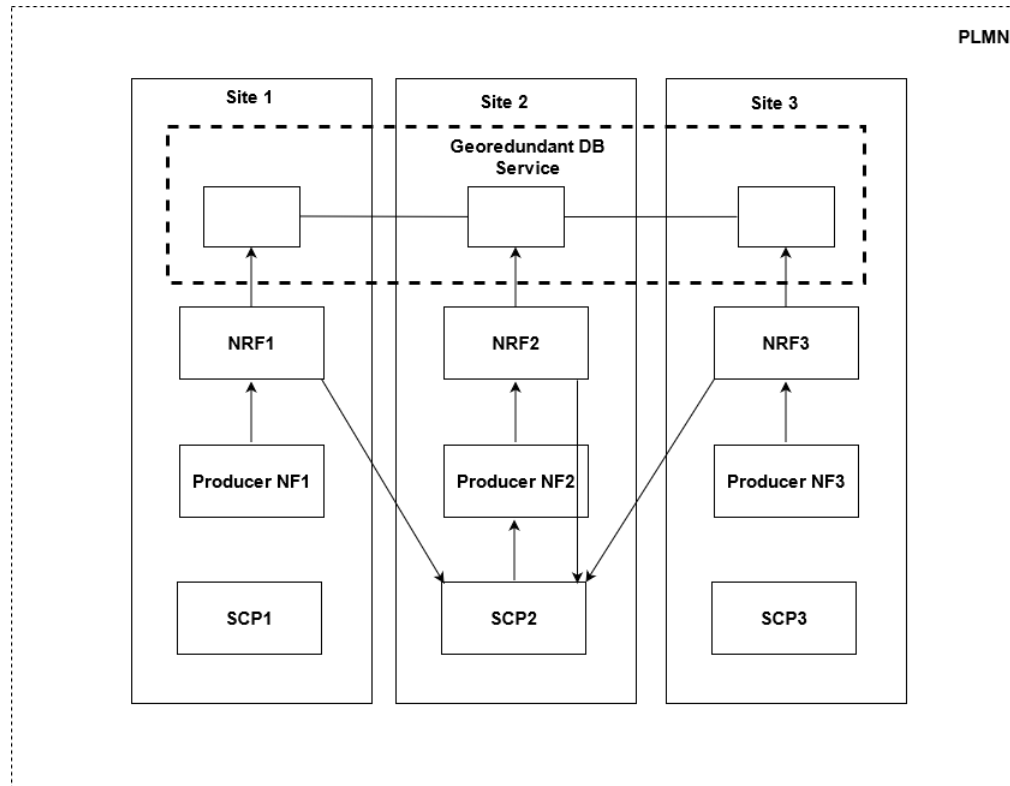
SCP performs periodic 5G NF topology audits with one of the available NRF instances as per priority and capacity in each configured NRF set using the NFListRetrieval and NFProfileRetrieval service operations of the Nnrf_NFManagement service if nnrf-mgmt services is configured for audit, or SCP can learn profiles using the NRF discovery service if nnrf-disc is configured for audit.

SCP maintains servingScope (serving locality) to SCP instance mapping when it gets NF status change notification from NRF for any other SCP instance that can be used for routing or forwarding the service request between SCP instances wherever applicable.

SCP determines all the SCP instances from NF-SetId published in the SCP profile. SCP instances are identified and grouped based on their respective NFSetId published.

Producer NF Registration Notification from NRF to SCP

The following image is an example of a scenario that describes how SCP receives producer NF registration notifications from NRF instances.

Figure 8-1 Producer NF Registration Notification from NRF to SCP

There are three NRF instances, NRF1, NRF2, and NRF3, which are deployed as georedundant NRFs across three sites, in an NRF set. SCP1 and SCP3 do not establish communication with the corresponding NRF instances such as NRF1 and NRF3. However, SCP2 subscribes with NRF2 for all changes for 5G NF types. When producer NF1, NF2, and NF3 register with NRF1, NRF2, and NRF3, respectively, SCP2 receives notification regarding producer NF1, NF2, and NF3 registration from NRF1, NRF2, and NRF3.

SCP uses the `nfStatusNotificationUri` callback URI to receive notifications from NRF. `nfStatusNotificationUri` notifies SCP about registrations, deregistrations, or profile changes of any NF profile. To subscribe to NRF notifications, SCP uses the POST method.

The following format of the callback URI is used by SCP in Subscription requests:

```
http://<scp-worker FQDN or IP>:<scp-worker
Port>/ocscp/scpc-notification/v1/notification/<nrfRegionSetID>/<nftype>/
<UUID>
```

Where,

- The FQDN or IP address is determined by the deployment configuration.
- scp-worker Port indicates the service port of SCP-Worker.

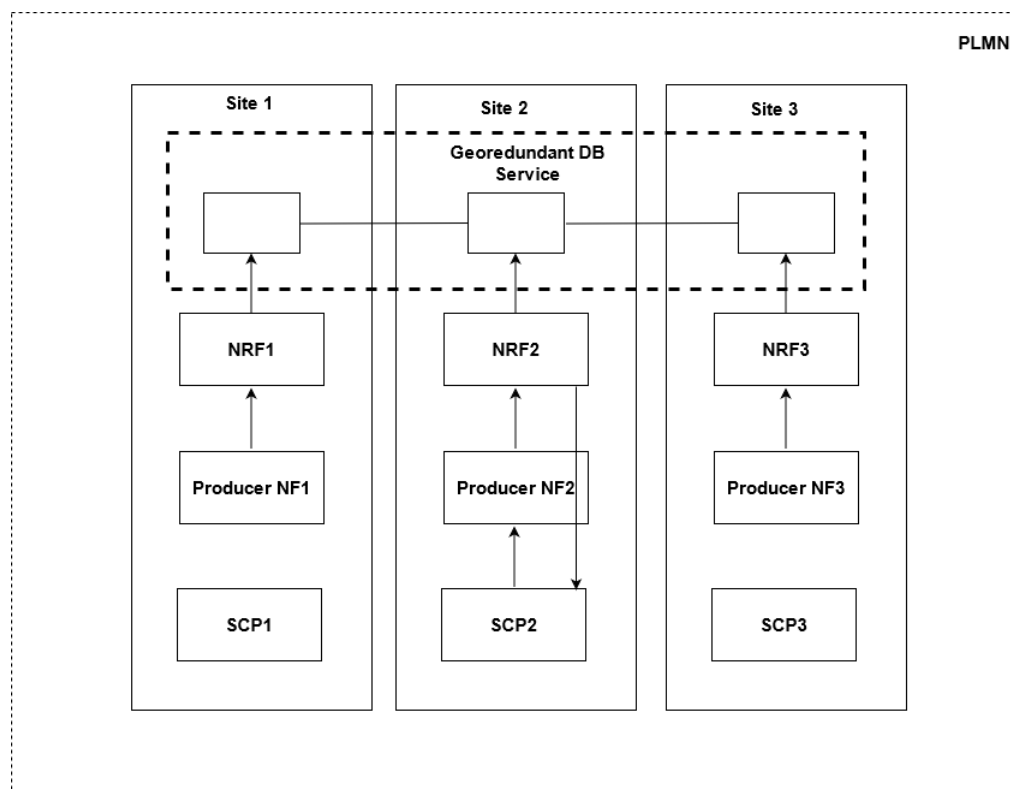
- `nrfRegionSetID` indicates the NRF's region or SetID based on the NRF configuration used during deployment.
- `nfType` indicates the NF type to which the Subscription is sent.
- `UUID` is the Universally Unique ID.

SCP can process notifications containing full NFProfiles that is `nfProfile` or partial profiles that is `profileChanges`. For more information about `nfStatusNotificationUri`, see 3GPP 29.510.

5G NF Topology Audit Response from NRF

The following image is an example of a scenario that describes how SCP receives 5G NF topology audit response from NRF.

Figure 8-2 5G NF Topology Audit Response from NRF



There are three NRF instances, NRF1, NRF2, and NRF3, which are deployed as georedundant NRFs across three sites, in an NRF set. Producer NF1, NF2, and NF3 register with NRF1, NRF2, and NRF3, respectively. SCP1 and SCP3 do not establish communication with the corresponding NRF instances such as NRF1 and NRF3. When SCP2 performs period audit for 5G NF types with NRF2, SCP2 receives audit response from NRF2 with all producer NF1, NF2, and NF3 irrespective of where they are registered.

9

Supported Query Parameters

This section is applicable for the Managing Delegated Discovery Response Caching for Subscriber IDs feature.

Table 9-1 Supported Query Parameter

Name	Data Type	Subscriber Id (Yes or No)	P	Description
target-nf-type	NFType	No	M	This Information Element (IE) contains the NF type of the target NF being discovered.
requester-nf-type	NFType	No	M	This IE contains the NF type of the Requester NF invoking the Nnrf_NFDiscovery service.
requester-nf-instance-id	NfInstanceId	No	O	If present, this IE contains the NF instance ID of the Requester NF.
service-names	array(ServiceName)	No	O	If this IE is included, it will contain an array of service names for which the NRF is requested to provide the list of NF profiles. The NRF will return the NF profiles that have at least one NF service matching the service names specified in this list. The returned NF service names from the NRF will be an interclause of the requested NF service names and the NF service names registered in the NF profile. If this IE is not included, the NRF will return all the NF service names registered in the NF profile. The list contains unique items.
requester-nf-instance-fqdn	Fqdn	No	O	This IE might be present for an NF discovery request within the same PLMN as the NRF. If included, this IE must contain the Fully Qualified Domain Name (FQDN) of the Requester NF that is invoking the Nnrf_NFDiscovery service. The NRF must use this to return only those NF profiles that include at least one NF service containing an entry in the 'allowedNfDomains' list (see clause 6.1.6.2.3) that matches the domain of the requester NF. This IE is ignored by the NRF if received from a requester NF belonging to a different PLMN.
target-plmn-list	array(PlmnId)	No	C	This IE is included when NF services in a different PLMN, or NF services of specific PLMN IDs in the same PLMN comprising multiple PLMN IDs, need to be discovered. When included, this IE will contain the target NF's PLMN ID. If more than one PLMN ID is included, NFs from any PLMN ID present in the list match the query parameter. When inter-PLMN service discovery occurs, the list should contain no more than one PLMN ID. This ID should appear in the service discovery request from the NF in the source PLMN to the NRF in the same PLMN, but it might not appear in the service discovery request from the source NRF to the target NRF. In such a case, if the NRF receives more than 1 PLMN ID, it will only consider the first element of the array and ignore the rest.

Table 9-1 (Cont.) Supported Query Parameter

Name	Data Type	Subscriber Id (Yes or No)	P	Description
requester-plmn-list	array(PlmnId)	No	C	This IE is included when NF services in a different PLMN need to be discovered. When included, this IE shall contain the PLMN ID(s) of the requester NF.
requester-snpn-list	array(PlmnIdNid)	No	C	This IE is included when the requester NF belongs to one or several SNPNs and the NF services of a specific SNPN need to be discovered. When present, this IE shall contain the requester NF's SNPN ID(s). The NRF uses this to return only those NF profiles of NF instances that are allowed to be discovered from the SNPNs identified by this IE, according to the "allowedSnpons" list in the NF Profile and NF Service (see clauses 6.1.6.2.2 and 6.1.6.2.3).
target-nf-instance-id	NfInstanceId	No	O	Identity of the NF instance being discovered.
target-nf-fqdn	Fqdn	No	O	FQDN of the target NF instance being discovered.
hnrf-uri	Uri	No	C	If included, this IE shall contain the API URI of the home NRF's NFDISCOVERY Service (see clause 6.2.1). It shall be included if the Requester NF has previously received such an API URI to be used for service discovery (e.g., from the NSSF in the home PLMN).
snssais	array(Snssai)	No	O	If included, this IE shall contain the list of S-NSSAIs that are served by the NF (service) instances being discovered. The NRF shall return those NF profiles or NF services of NF (service) instances that have at least one of the S-NSSAIs in this list. The S-NSSAIs included in the NF profiles or NF services of NF (service) instances returned by the NRF shall be an interclause of the S-NSSAIs requested and the S-NSSAIs supported by those NF (service) instances. (NOTE 10) When the NF Profile of the NF Instances being discovered has defined the list of supported S-NSSAIs in the "perPlmnSnssaiList", the discovered NF Instances shall be those having any of the S-NSSAIs included in this "snssais" parameter in any of the PLMNs included in the "target-plmn-list" attribute, if present; if the "target-plmn-list" is not included, the NRF shall assume that the discovery request is for any of the PLMNs it supports.
requester-snssais	array(Snssai)	No	O	If included, this IE shall contain the list of S-NSSAI of the requester NF. If this IE is included in a service discovery in a different PLMN, the requester NF shall provide S-NSSAI values of the target PLMN that correspond to the S-NSSAI values of the requester NF. The NRF shall use this to return only those NF profiles of NF instances allowed to be discovered from at least one network slice identified by this IE, according to the "allowedNssais" list in the NF Profile and NF Service (see clauses 6.1.6.2.2 and 6.1.6.2.3). (NOTE 12)

Table 9-1 (Cont.) Supported Query Parameter

Name	Data Type	Subscriber Id (Yes or No)	P	Description
plmn-specific-snssai-list	array(Plmn Snssai)	No	O	If included, this IE shall contain the list of S-NSSAI that are served by the NF service being discovered for the corresponding PLMN provided. The NRF shall use this to identify the NF services that have registered their support for the S-NSSAIs for the corresponding PLMN given. The NRF shall return the NF profiles that have at least one S-NSSAI supported in any of the PLMNs provided in this list. The per PLMN list of S-NSSAIs included in the NF profile returned by the NRF shall be an interclause of the list requested and the list registered in the NF profile. (NOTE 10).
requester-plmn-specific-snssai-list	array(Plmn Snssai)	No	O	If included, this IE shall contain the list of S-NSSAI of the requester NF for each of the PLMNs it supports. The NRF shall use this to return only those NF profiles of NF instances allowed to be discovered from at least one network slice identified by this IE, according to the "allowedNssais" and "allowedPlmns" attributes in the NF Profile and NF Service (see clauses 6.1.6.2.2 and 6.1.6.2.3). (NOTE 12)
nsi-list	array(string)	No	O	If included, this IE shall contain the list of NSI IDs that are served by the services being discovered.
dnn	Dnn	-	O	If included, this IE shall contain the DNN for which NF services serving that DNN were discovered. DNN may be included if the target NF type is, for example, "BSF," "SMF," "PCF," "PCSCF," or "UPF". The DNN shall contain the network identifier, and it may additionally contain an operator identifier. (NOTE 11). If the Snssai(s) are also included, the NF services serving the DNN will be available in the network slice(s) identified by the Snssai(s).
smf-serving-area	string	No	O	If included, this IE shall contain the serving area of the SMF. It may be included if the target NF type is "UPF".
tai	Tai	No	O	Tracking Area Identity.
amf-region-id	AmfRegionId	No	O	AMF Region Identity.
amf-set-id	AmfSetId	No	O	AMF Set Identity.
guami	Guami	No	O	Guami used to search for an appropriate AMF. (NOTE 1)
supi	Supi	Yes	O	If included, this IE shall contain the SUPI of the requester UE to search for an appropriate NF. SUPI may be included if the target NF type is, for example, "PCF," "CHF," "AUSF," "UDM," or "UDR."
ue-ipv4-address	Ipv4Addr	Yes	O	The IPv4 address of the UE for which a BSF or P-CSCF needs to be discovered.
ip-domain	string	Yes	O	The IPv4 address domain of the UE for which a BSF needs to be discovered.
ue-ipv6-prefix	Ipv6Prefix	Yes	O	The IPv6 prefix of the UE for which a BSF or P-CSCF needs to be discovered.

Table 9-1 (Cont.) Supported Query Parameter

Name	Data Type	Subscriber Id (Yes or No)	P	Description
pgw-ind	boolean	No	O	When present, this IE indicates whether a combined SMF/PGW-C or a standalone SMF needs to be discovered. true: A combined SMF/PGW-C is requested to be discovered; false: A standalone SMF is requested to be discovered.
pgw	Fqdn	No	O	If included, this IE shall contain the PGW FQDN, which is received by the AMF from the MME to find the combined SMF or PGW.
gpsi	Gpsi	Yes		If included, this IE shall contain the GPSI of the requester UE to search for an appropriate NF. GPSI may be included if the target NF type is "CHF," "PCF," "UDM," or "UDR."
external-group-identity	ExtGroupId	Yes	O	If included, this IE shall contain the external group identifier of the requester UE to search for an appropriate NF. This may be included if the target NF type is "UDM" or "UDR."
pfd-data	PfdData	No	O	When present, this IE shall contain the application identifiers and/or application function identifiers in PFD management. This may be included if the target NF type is "NEF". The NRF shall return those NEF instances that can provide the PFDs for at least one of the provided application identifiers or for at least one of the provided application function identifiers.
data-set	DataSetId	No	O	Indicates the data set to be supported by the NF to be discovered. May be included if the target NF type is "UDR."
routing-indicator	string	Yes	O	Routing indicator information that allows to route network signaling with SUCI (see 3GPP 23.003 [12]) to an AUSF and UDM instance capable of serving the subscriber. May be included if the target NF type is "AUSF" or "UDM." Pattern: "^[0-9]{1,4}\$"
group-id-list	array(NfGroupId)	No		Identity of the groups of the NFs of the target NF type to be discovered. May be included if the target NF type is "UDR," "UDM," "HSS," "PCF," "AUSF," or "CHF."
dnai-list	array(Dnai)	No		If included, this IE shall contain the data network access identifiers. It may be included if the target NF type is "UPF."
upf-iwk-eps-ind	Boolean	No	O	When present, this IE indicates whether a UPF supporting interworking with EPS needs to be discovered. true: A UPF supporting interworking with EPS is requested to be discovered; false: A UPF not supporting interworking with EPS is requested to be discovered.
chf-supported-plmn	PlmnId	No	O	If included, this IE shall contain the PLMN ID that a CHF supports (for example, in the PlmnRange of ChfInfo attribute in the NFProfile). This IE may be included when the target NF type is "CHF."

Table 9-1 (Cont.) Supported Query Parameter

Name	Data Type	Subscribe Id (Yes or No)	P	Description
preferred-locality	string	No	O	Preferred target NF location (for example, geographic location, data center). When present, the NRF shall prefer NF profiles with a locality attribute that matches the preferred locality. The NRF may return additional NFs in the response not matching the preferred target NF location, for example, if no NF profile is found matching the preferred target NF location. The NRF should set a lower priority for any additional NFs on the response not matching the preferred target NF location than those matching the preferred target NF location. (NOTE 6)
access-type	AccessType	No	C	If included, this IE shall contain the access type that is required to be supported by the target NF, such as SMF.
supported-features	Supported Features	No	O	List of features required to be supported by the target NF. This IE may be present only if the service-names attribute is present and if it contains a single service name. Otherwise, the NRF will ignore it. (NOTE 4)
required-features	array(SupportedFeatures)	No	O	List of features required to be supported by the target Network Function, as defined by the supportedFeatures attribute in NFService (see clauses 6.1.6.2.3 and 6.2.6.2.4). This IE may be present only if the service-names attribute is present. When present, the required-features attribute shall contain as many entries as the number of entries in the service-names attribute. The nth entry in the required-features attribute shall correspond to the nth entry in the service-names attribute. An entry corresponding to a service for which no specific feature is required shall be encoded as "0."
complex-query	ComplexQuery	No	O	This query parameter is used to override the default logical relationship of query parameters.
limit	integer	No	O	The maximum number of NFProfiles to be returned in the response. Minimum: 1
max-payload-size	integer	No	O	Maximum payload size (before compression, if any) of the response, expressed in kilo octets. When present, the NRF shall limit the number of NF profiles returned in the response so as not to exceed the maximum payload size indicated in the request. Default: 124. Maximum: 2000 (such as 2 million octets).

Table 9-1 (Cont.) Supported Query Parameter

Name	Data Type	Subscribe Id (Yes or No)	P	Description
max-payload-size-ext	integer	No	O	Maximum payload size (before compression, if any) of the response, expressed in kilo octets. When present, the NRF shall limit the number of NF profiles returned in the response so as not to exceed the maximum payload size indicated in the request. This query parameter is used when the consumer supports payload sizes larger than 2 million octets. Default: 124
pdu-session-types	array(Pdu SessionType)	No	O	List of the PDU session types requested to be supported by the target Network Function (such as UPF).
event-id-list	array(EventId)	No	O	If present, this attribute shall contain the list of events requested to be supported by the Nnwdaf AnalyticsInfo Service, and the NRF shall return NF, which supports all the requested events.
nwdaf-event-list	array(NwdafEvent)	No	O	If present, this attribute shall contain the list of events requested to be supported by the Nnwdaf_EventsSubscription service, and the NRF shall return NF, which supports all the requested events.
atsss-capability	AtsssCapability	No	O	When present, this IE indicates the ATSSS capability of the target UPF needs to be supported.
upf-ue-ip-addr-ind	boolean	No	O	When present, this IE indicates whether a UPF supporting allocating UE IP addresses or prefixes needs to be discovered. true: a UPF supporting UE IP addresses and prefixes is requested to be discovered; false: a UPF not supporting UE IP addresses or prefix allocation is requested to be discovered.
client-type	ExternalClientType	No	O	When present, this IE indicates that NF(s) dedicatedly serving the specified client type need to be discovered. This IE may be included when the target NF type is "LMF" or "GMLC". If no NF profile is found dedicatedly serving the requested client type, the NRF may return NF(s) not dedicatedly serving the requested client type in the response.
lmf-id	LMFIdentification	No	O	When present, this IE shall contain LMF identification to be discovered. This may be included if the target NF type is "LMF."
an-node-type	AnNodeType	No	O	If included, this IE shall contain the AN Node type, which is required to be supported by the target Network Function (such as LMF).
rat-type	RatType	No	O	If included, this IE shall contain the RAT type that is required to be supported by the target Network Function (such as LMF).
target-snpn	PlmnIdNid	No	C	This IE shall be included when the NF services of a specific SNPN need to be discovered. When included, this IE shall contain the PLMN ID and NID of the target NF.

Table 9-1 (Cont.) Supported Query Parameter

Name	Data Type	Subscribe Id (Yes or No)	P	Description
af-ee-data	AfEventExposureData	No	O	When present, this shall contain the application events and, optionally, application function identifiers and application identifiers of the AF(s). This may be included if the target NF type is "NEF."
w-agf-info	WAgfInfo	No	O	If included, this IE shall contain the W-AGF identifiers of N3 terminations which is received by the SMF to find the combined W-AGF/UPF.
tngf-info	TngfInfo	No	O	If included, this IE shall contain the TNGF identifiers of N3 terminations which is received by the SMF to find the combined TNGF or UPF.
twif-info	TwifInfo	No	O	If included, this IE shall contain the TWIF identifiers of N3 terminations which is received by the SMF to find the combined TWIF or UPF.
target-nf-set-id	NfSetId	No	O	When present, this IE shall contain the target NF Set ID (as defined in clause 28.12 of 3GPP TS 23.003 [12]) of the NF instances being discovered.
target-nf-service-set-id	NfServiceSetId	No	O	When present, this IE shall contain the target NF Service Set ID (as defined in clause 28.13 of 3GPP TS 23.003 [12]) of the NF service instances being discovered.
preferred-tai	Tai	No	O	When present, the NRF shall prefer NF profiles that can serve the TAI, or the NRF shall return NF profiles not matching the TAI if no NF profile is found matching the TAI. (NOTE 5)
nef-id	NefId	No	O	When present, this IE shall contain the NEF ID of the NEF to be discovered. This may be included if the target NF type is "NEF." (NOTE 7)
preferred-nf-instances	array(NfInstanceId)	No	O	When present, this IE shall contain a list of preferred candidate NF instance IDs. (NOTE 8)
notification-type	NotificationType	No	O	If included, this IE shall contain the notification type of default notification subscriptions that shall be registered in the NFProfile or NFService of the NF Instances being discovered. The NF profiles returned by the NRF shall contain all the registered default notification subscriptions, including the one corresponding to the notification-type parameter. (NOTE 9)
n1-msg-class	N1MessageClass	No	O	This IE may be included when the "notification-type" IE is present with the value "N1_MESSAGES." When included, this IE shall contain the N1 message class of default notification subscriptions that shall be registered in the NFProfile or NFService of the NF Instances being discovered. The NF profiles returned by the NRF shall contain all the registered default notification subscriptions, including the one corresponding to the n1-msg-class parameter. (NOTE 9)

Table 9-1 (Cont.) Supported Query Parameter

Name	Data Type	Subscribe Id (Yes or No)	P	Description
n2-info-class	N2InformationClass	No	O	This IE may be included when the "notification-type" IE is present with the value "N2_INFORMATION." If included, this IE shall contain the notification type of default notification subscriptions that shall be registered in the NFProfile or NFService of the NF Instances being discovered. The NF profiles returned by the NRF shall contain all the registered default notification subscriptions, including the one corresponding to the n2-info-class parameter. (NOTE 9)
serving-scope	array(string)	No	O	If present, this attribute shall contain the list of areas that can be served by the NF instances to be discovered. The NRF shall return NF profiles of NFs that can serve all the areas requested in this query parameter.
imsi	string	No	O	If included, this IE shall contain the IMSI of the requester UE to search for an appropriate NF. IMSI may be included if the target NF type is "HSS". pattern: "[0-9]{5,15}"
ims-private-identity	string	No	O	If included, this IE shall contain the IMS private identity of the requester UE to search for an appropriate NF. IMS Private Identity may be included if the target NF type is "HSS."
ims-public-identity	string			If included, this IE shall contain the IMS public identity of the requester UE to search for an appropriate NF. IMS Public Identity may be included if the target NF type is "HSS."
msisdn	string			If included, this IE shall contain the MSISDN of the requester UE to search for an appropriate NF. IMS Public Identity may be included if the target NF type is "HSS".
internal-group-identity	GroupId	No	O	If included, this IE shall contain the internal group identifier of the UE to search for an appropriate NF. This may be included if the target NF type is "UDM."

Table 9-1 (Cont.) Supported Query Parameter

Name	Data Type	Subscriber Id (Yes or No)	P	Description
preferred-api-versions	map(string)	No	O	When present, this IE indicates the preferred API version of the services that are supported by the target NF instances. The key of the map is the service name (see clause 6.1.6.3.11) for which the preferred API version is indicated. Each element carries the API version indication for the service indicated by the key. The NRF may return additional NFs in the response not matching the preferred API versions, for example, if no NF profile is found matching the preferred- API -versions. An API Version Indication is a string formatted as {operator}{API Version}. The following operators will be supported: • "=" matches a version equal to the version value indicated. • "<=" matches a version equal to the version value indicated. • ">" matches any version greater than the version value indicated. • ">=" matches any version greater than or equal to the version value indicated. • "<" matches any version less than the version value indicated. • "<=" matches any version less than or equal to the version value indicated. • "^" matches any version compatible with the version indicated, such as, any version with the same major version as the version indicated. Precedence between versions is identified by comparing the Major, Minor, and Patch version fields numerically, from left to right. If no operator or an unknown operator is provided in API Version Indication, "=" operator is applied. Example of API Version Indication: Case1: "=1.2.4.operator-ext" or "1.2.4.operator-ext" means matching the service with API version "1.2.4.operator-ext" Case2: ">1.2.4" means matching the service with API versions greater than "1.2.4" Case3: "^2.3.0" or "^2" means matching the service with all API versions with major version "2".
v2x-support-ind	boolean	No	O	When present, this IE indicates whether a PCF supporting V2X policy or parameter provisioning needs to be discovered. true: a PCF supporting V2X policy or parameter provisioning is requested to be discovered; false: a PCF not supporting V2X policy or parameter provisioning is requested to be discovered.

Table 9-1 (Cont.) Supported Query Parameter

Name	Data Type	Subscriber Id (Yes or No)	P	Description
redundant-gtpu	boolean	No	O	When present, this IE indicates whether a UPF supporting redundant GTP-U paths needs to be discovered. true: a UPF supporting redundant GTP-U paths is requested to be discovered; false: a UPF not supporting redundant GTP-U paths is requested to be discovered.
redundant-transport	boolean	No	O	When present, this IE indicates whether a UPF supporting redundant transport paths on the transport layer in the corresponding network slice needs to be discovered. true: a UPF supporting redundant transport paths on the transport layer is requested to be discovered; false: a UPF not supporting redundant transport paths on the transport layer is requested to be discovered. If the Snssai(s) are also included, the UPF supporting redundant transport paths on the transport layer shall be available in the network slice(s) identified by the Snssai(s).
ipups	boolean	No	O	When present, this IE indicates whether a UPF that is configured for IPUPS is requested to be discovered. true: a UPF that is configured for IPUPS is requested to be discovered; false: a UPF that is not configured for IPUPS is requested to be discovered.
scp-domain-list	array(string)	No	O	When present, this IE shall contain the SCP domains the target NF or SCP belongs to. The NRF shall return NF or SCP profiles that belong to all the SCP domains provided in this list.
address-domain	Fqdn	No	O	If included, this IE shall contain the address domain that shall be reachable through the SCP. This IE may be included when the target NF type is "SCP."
ipv4-addr	Ipv4Addr	No	O	If included, this IE shall contain the IPv4 address that shall be reachable through the SCP. This IE may be included when the target NF type is "SCP."
ipv6-prefix	Ipv6Prefix	No	O	If included, this IE shall contain the IPv6 prefix that shall be reachable through the SCP. This IE may be included when the target NF type is "SCP".
served-nf-set-id	NfSetId	No	O	When present, this IE shall contain the NF Set ID that shall be reachable through the SCP. This IE may be included when the target NF type is "SCP".
remote-plmn-id	PlmnId	No	O	If included, this IE shall contain the remote PLMN ID that shall be reachable through the SCP. This IE may be included when the target NF type is "SCP".

Table 9-1 (Cont.) Supported Query Parameter

Name	Data Type	Subscriber Id (Yes or No)	P	Description
data-forwarding	boolean	No	O	This may be included if the target NF type is "UPF." (NOTE 13) When present, the IE indicates whether UPFs configured for data forwarding need to be discovered. true: UPF(s) configured for data forwarding are requested to be discovered; false: UPF(s) not configured for data forwarding are requested to be discovered.
preferred-full-plmn	boolean	No	O	When present, the NRF shall prefer NF profile(s) that can serve the full PLMN (can serve any TAI in the PLMN), or the NRF shall return other NF profiles if no NF profile serving the full PLMN is found: true: NF instances serving the full PLMN are preferred; false: NF instances serving the full PLMN are not preferred. (NOTE 14)
requester-features	Supported Features	No	C	Nnrf_NFDiscovery features are supported by the Requester NF that is invoking the Nnrf_NFDiscovery service. The Requester NF shall include this IE if it supports at least one of the following features: - Service-Map This IE may be included elsewhere.
realm-id	string	No	O	May be included if the target NF type is "UDSF." If included, this IE shall contain the realm- ID for which a UDSF shall be discovered.
storage-id	string	No	O	May be included if the target NF type is "UDSF" and realm-id is included. If included, this IE shall contain the storage-id for the realm-id indicated in the realm-id IE for which a UDSF shall be discovered.
vsmf-support-ind	boolean	No	O	If included, this IE shall indicate that target SMF(s) that support V-SMF capability are preferred. This IE may be included when the target NF type is "SMF." (NOTE 15)
gmlc-number	string	No	O	If included, this IE shall contain the GMLC number, which should be supported by the target GMLC. It may be included if the target NF type is "GMLC." Pattern: "[0-9]{5,15}\$"

10

Supported NF Types and Services

SCP supports the following NF types and service types. These services can be managed using routing options.

Table 10-1 Supported NF Types and Services

Network Function	Service Types
5G_EIR	n5g-eir-eic
AMF	namf-comm, namf-evts, namf-loc, namf-mt
AUSF	nausf-auth, nausf-sorprotection, nausf-upuprotection
BSF	nbsf-management
CHF	nchf-convergedcharging, nchf-spendinglimitcontrol, nchf-spendinglimitcontrol
LMF	nlmf-loc
NEF	nnef-pfdmanagement, nnef-eventexposure, nnef-afsessionwithqos - Nnef_EventExposure: Allows NF service consumers to subscribe to and modify application events and to notify NF service consumers about observed events on the NEF. - Nnef_AFsessionWithQoS: Allows an AF to request the network to provide a specific Quality of service (QoS) for a session.
NRF	nnrf-disc, nnrf-nfm, nnrf-oauth2
NSSF	nssf-nssaiavailability, nssf-nsselection
NWDAF	nnwdaf-analyticsinfo, nnwdaf-eventssubscription
PCF	npcf-am-policy-control, npcf-bdtpolicycontrol, npcf-eventexposure, npcf-policyauthorization, npcf-smpolicycontrol, npcf-ue-policy-control
SEPP	NA
SCP	nscp-5g-sbi-proxy Note: This service is supported only in the Release 15 deployment model.
SMF	nsmf-event-exposure, nsmf-pdusession
SMSF	nsmsf-sms
UDM	nudm-ee, nudm-pp, nudm-sdm, nudm-ueau, nudm-uecm
UDR	nudr-dr, nudr-group-id-map
NA	5g-sbi-notification, nmediation-http, ocscp-sds, scpc-notification Note: Notifications from NRF to scpc-notification will be pegged with the service type as scpc-notification.

Supported 3GPP Headers

The following table lists the headers supported by SCP:

Table 11-1 Supported 3GPP Custom Headers

Custom Header Name	Description
3gpp-Sbi-Message-Priority	This header is used to specify the HTTP/2 message priority for 3GPP service based interfaces.
3gpp-Sbi-Callback	The header is used to identify a particular type of callback, for example, notification.
3gpp-Sbi-Target-apiRoot	The header contains the apiRoot of the target URI in a message request sent to an SCP instance when using indirect communication such as Model C and Model D. SCP uses this header to indicate the apiRoot of the target URI if a new HTTP server is selected or reselected and if there is no location header in the message response. SCP adds this header to message responses when: 3gpp-Sbi-Discovery-target-nf-set-id is present in the message request. 3gpp-Sbi-Target-apiRoot and 3GPP-Sbi-Disocvery-target-NfSetid are present in the message request. 3gpp-Sbi-Target-apiRoot and 3gpp-sbi-routing-binding are present in the message request.
3gpp-Sbi-Routing-Binding	This header is used in a service request to signal binding information to direct the service request to an HTTP server that has the targeted NF service resource context.
3gpp-Sbi-Binding	This header is used to signal binding information related to an NF service resource to a future consumer, such as an HTTP client, of that resource.
3gpp-Sbi-Discovery-*	This header, which begins with the prefix 3gpp-Sbi-Discovery, is used in indirect communication mode for discovery and selection of a suitable producer by the SCP. Such headers can be included in any SBI message and include information allowing an SCP to find a suitable producer based on the consumer's included delegated discovery parameters.
3gpp-Sbi-Oci	This header is utilized by overloaded SCPs, SEPPs, consumer NFs, and producer NFs to transmit Overload Control Information (OCI) within message requests and responses to peers. The OCI includes attributes such as Overload Timestamp, Overload-Reduction-Metric, Period-of-Validity, and Scope within the 3gpp-Sbi-Oci header.
3gpp-Sbi-Producer-Id	This header contains Instance ID of the producer NF. In case of alternate routing in indirect communication modes such as Model C or Model D, this header is used in a message response from SCP to the consumer NF to identify the producer NF. SCP adds this header to message responses when: 3gpp-Sbi-Discovery-target-nf-set-id is present in the message request. 3gpp-Sbi-Target-apiRoot and 3GPP-Sbi-Disocvery-target-NfSetid are present in the message request. 3gpp-Sbi-Target-apiRoot and 3gpp-sbi-routing-binding are present in the message request.
3gpp-Sbi-Lci	This header is used by a producer NF to send Load Control Information (LCI) to the consumer NF.
3gpp-Sbi-Client-Credentials	This header is used by a consumer NF to send a Client Credential Assertion (CCA) to the NRF or to the producer NF.

Table 11-1 (Cont.) Supported 3GPP Custom Headers

Custom Header Name	Description
3gpp-Sbi-Target-Nf-Id	This header is used in a 307 Temporary Redirect or 308 Permanent Redirect response to identify the target NF service instance towards which the request is redirected.
3gpp-Sbi-Access-Scope	This header is used in a service request for indirect communication to indicate the access scope of the service request for NF service access authorization.
3gpp-Sbi-Access-Token	This header is used in a service response forwarded by the SCP to an NF service consumer to provide an access token for possible reuse in subsequent service requests.
3gpp-Sbi-Correlation-Info	This header may be used to contain correlation information, for example, UE identity, that may be used by an operator in various offline network management, performance analysis and troubleshooting tools/applications to identify messages (requests, responses, subscriptions, notifications) related to a particular subscriber.

SCP also supports the HTTP standard headers. For more information, see *3GPP TS 29.500 V16.10.0, Table 5.2.2.2-1*.

A

HTTP Status Code and Applicability for Rerouting

This section describes the HTTP status codes usage on SBI. HTTP status codes are carried in the ":status" pseudo header field in HTTP/2, as defined in subclause 8.1.2.4 in IETF RFC 7540.

The following table specifies HTTP status codes per HTTP method which is supported on SBI. Support of the HTTP status code is:

- Mandatory, which is marked in table as "**M**". This signifies that all 3GPP NFs support the processing of the specific HTTP status code for the specific HTTP method, when received in a HTTP response message. In such cases the 3GPP NF also supports the handling of the "ProblemDetails" JSON object with the Content-Type header field set to the value "application/problem+json" for HTTP status codes 4xx and 5xx, if the corresponding API definition in the related technical specification does not specify another response body for the corresponding status code;
- Service specific, which is marked in table as "**SS**" and means that the requirement to process the HTTP status code depends on the definition of the specific API; or
- Not applicable, which is marked in table as "**N/A**". This means that the specific HTTP status code shall not be used for the specific HTTP method within the 3GPP NFs.
- Applicable for Rerouting column describes if the status code is applicable for rerouting at SCP. These Status codes can be configured in Routing options for each NF services.

NOTE 1: "200 OK" response used on SBI shall contain body.

NOTE 2: If the NF acting as an HTTP Client receives 2xx response code not appearing in table, the NF shall treat the received 2xx response: - as "204 No Content" if 2xx response does not contain body; and - as "200 OK" if 2xx response contains body.

HTTP status code supported on SBI

Table A-1 HTTP status code supported on SBI

HTTP Status Code						Applicable for Rerouting	HTTP Method
	DELETE	GET	PATCH	POST	PUT		
100 Continue	N/A	N/A	N/A	N/A	N/A	No	
200 OK (NOTE 1)	SS	M	SS	SS	SS	No	
201 Created	N/A	N/A	N/A	SS	SS	No	
202 Accepted	SS	N/A	SS	SS	SS	No	
204 No Content (NOTE 2)	M	N/A	SS	SS	SS	No	
300 Multiple Choices	N/A	N/A	N/A	N/A	N/A	No	
303 See Other	SS	SS	N/A	SS	SS	No	

Table A-1 (Cont.) HTTP status code supported on SBI

307 Temporary Redirect	SS	SS	SS	SS	SS	Yes	307 (Should be included as part of 3xx)
308 Permanent Redirect	SS	SS	SS	SS	SS	Yes	308 (Should be included as part of 3xx)
400 Bad Request	M	M	M	M	M	No	
401 Unauthorized	M	M	M	M	M	No	
403 Forbidden	SS	SS	SS	SS	SS	No	
404 Not Found	SS	SS	SS	SS	SS	Yes	404
405 Method Not Allowed	SS	SS	SS	SS	SS	No	
406 Not Acceptable	N/A	N/A	N/A	N/A	N/A	No	
408 Request Timeout	SS	SS	SS	SS	SS	Yes	408
409 Conflict	N/A	N/A	SS	SS	SS	Yes	409 (should be included as part of "retriable-4xx")
410 Gone	SS	SS	SS	SS	SS	Yes	410
411 Length Required	N/A	N/A	M	M	M	No	
412 Precondition Failed	SS	SS	SS	SS	SS	No	
413 Payload Too Large	N/A	N/A	M	M	M	No	
414 URI Too Long	N/A	M	N/A	N/A	N/A	No	
415 Unsupported Media Type	N/A	N/A	M	M	M	No	
429 Too Many Requests	N/A	N/A	N/A	N/A	N/A	Yes	
500 Internal Server Error	M	M	M	M	M	Yes	500
501 Not Implemented	SS	SS	SS	SS	SS	Yes	501
502 Bad Gateway	SS	SS	SS	SS	SS	Yes	502 (Should be included as part of gateway error and "5xx")
503 Service Unavailable	M	M	M	M	M	Yes	503 (Should be included as part of "5xx")
504 Gateway Timeout	SS	SS	SS	SS	SS	Yes	504 (Should be included as part of "5xx")

NF as HTTP Client

Besides the HTTP Status Codes defined in the API specification, a NF as HTTP client should support handling of 1xx, 3xx, 4xx and 5xx HTTP Status Codes specified in above table,

following the client behavior in corresponding IETF RFC where the received HTTP Status Code is defined.

When receiving a not recommended or not recognized 1xx, 3xx, 4xx or 5xx HTTP Status Code, a NF as HTTP client should treat it as x00 status code of the class, as described in clause 6 of IETF RFC 7231.

If 100, 200/204, 300, 400 or 500 response code is not defined by the API specification, the client may follow guidelines below:

- For 1xx (Informational):
 - Discard the response and wait for final response.
- For 2xx (Successful):
 - Consider the service operation is successful if no mandatory information is expected from the response payload in subsequent procedure.
 - If mandatory information is expected from response payload in subsequent procedure, parse the payload following description in subclause 6.2.1 of IETF RFC 7231 [11]. If parse is successful and mandatory information is extracted, continue with subsequent procedure.
 - Otherwise, consider service operation has failure and start failure handling.
- For 3xx (Redirection):
 - Retry the request towards the directed resource referred in the Location header, using same request method.
- For 4xx (Client Error):
 - Validate the request message and make correction before resending. Otherwise, stop process and go to error handling procedure.
- For 5xx (Server Error):
 - Stop process and go to error handling process.

NF as HTTP Server

A NF acting as an HTTP server is able to generate HTTP status codes specified in above table per indicated HTTP method.

An HTTP method which is not supported by 5GC SBI API specification is rejected with the HTTP status code "501 Not Implemented".

NOTE 1: In this case, the NF does not need to include in the HTTP response the "cause" attribute indicating corresponding error since the HTTP status code "501 Not Implemented" itself provides enough information of the error, that is, the NF does not recognize the HTTP method.

If the specified target resource does not exist, the NF rejects the HTTP method with the HTTP status code "404 Not Found".

If the NF supports the HTTP method but not by a target resource, the NF rejects the HTTP method with the HTTP status code "405 Method Not Allowed" and includes in the response an Allow header field containing the supported method(s) for that resource.

NOTE 2: In this case, the NF does not need to include in the HTTP response the "cause" attribute indicating corresponding error since the HTTP status code "405 Method Not Allowed" itself provides enough information of the error and hence the Allow header field lists HTTP method(s) supported by the target resource.

If the received HTTP request contains incorrect optional IE, the NF discards the incorrect IE.

If the NF supports the HTTP method by a target resource but the NF cannot successfully fulfill the received request, the following requirements apply.

A NF as HTTP Server should map application errors to the most similar 3xx/4xx/5xx HTTP status code specified in table 5.2.7.1-1. If no such code is applicable, it should use "400 Bad Request" status code for errors caused by client side or "500 Server Internal Error" status code for errors caused on server side.

If the received HTTP request contains unsupported payload format, the NF rejects the HTTP request with the HTTP status code "415 Unsupported Media Type". If the HTTP PATCH method is rejected, the NF includes the Accept-Patch header field set to the value of supported patch document media types for a target resource i.e. to "application/merge-patch+json" if the NF supports "JSON Merge Patch" and to "application/json-patch+json" if the NF supports "JSON Patch". If the received HTTP PATCH request contains both "JSON Merge Patch" and "JSON Patch" documents and the NF supports only one of them, the NF ignores unsupported patch document.

NOTE 3: The format problem might be due to the request's indicated Content-Type or Content-Encoding header fields, or as a result of inspecting the payload body directly.

If the received HTTP request contains payload body larger than the NF is able to process, the NF rejects the HTTP request with the HTTP status code "413 Payload Too Large".

If the result of the received HTTP POST request used for a resource creation is equivalent to the existing resource, the NF rejects the HTTP request with the HTTP status code "303 See Other" and includes in the HTTP response a Location header field set to the URI of the existing resource.

Protocol and application errors common to several 5GC SBI API specifications for which the NF includes in the HTTP response a payload body ("ProblemDetails" data structure or application specific error data structure) with the "cause" attribute indicating corresponding error are listed in below table.

Table A-2 Protocol and Application Errors

Parameters	HTTP status code	Description
INVALID_API	400 Bad Request	The HTTP request contains an unsupported API name or API version in the URI.
INVALID_MSG_FORMAT	400 Bad Request	The HTTP request has an invalid format.
INVALID_QUERY_PARAM	400 Bad Request	The HTTP request contains an unsupported query parameter in the URI.
MANDATORY_IE_INCORRECT	400 Bad Request	A mandatory IE or conditional IE in data structure, but mandatory required, for an HTTP method was received with a semantically incorrect value. (NOTE 1)
MANDATORY_IE_MISSING	400 Bad Request	IE which is defined as mandatory or as conditional in data structure, but mandatory required, for an HTTP method is not included in the payload body of the request. (NOTE 1)

Table A-2 (Cont.) Protocol and Application Errors

Parameters	HTTP status code	Description
UNSPECIFIED_MSG_FAILURE	400 Bad Request	The request is rejected due to unspecified client error. (NOTE 2)
MODIFICATION_NOT_ALLOWED	403 Forbidden	The request is rejected because the contained modification instructions attempt to modify IE which is not allowed to be modified.
SUBSCRIPTION_NOT_FOUND	404 Not Found	The request for modification or deletion of subscription is rejected because the subscription is not found in the NF.
RESOURCE_URI_STRUCTURE_NOT_FOUND	404 Not Found	The request is rejected because a fixed part after the first variable part of an "apiSpecificResourceUriPart" (as defined in subclause 4.4.1 of 3GPP TS 29.501) is not found in the NF. This fixed part of the URI may represent a sub-resource collection (e.g. contexts, subscriptions, policies) or a custom operation. (NOTE 5)
INCORRECT_LENGTH	411 Length Required	The request is rejected due to incorrect value of a Content-length header field.
NF_CONGESTION_RISK	429 Too Many Requests	The request is rejected due to excessive traffic which, if continued over time, may lead to (or may increase) an overload situation.
INSUFFICIENT_RESOURCES	500 Internal Server Error	The request is rejected due to insufficient resources.
UNSPECIFIED_NF_FAILURE	500 Internal Server Error	The request is rejected due to unspecified reason at the NF. (NOTE 3)
SYSTEM_FAILURE	500 Internal Server Error	The request is rejected due to generic error condition in the NF.
NF_CONGESTION	503 Service Unavailable	The NF experiences congestion and performs overload control, which does not allow the request to be processed. (NOTE 4)

NOTE 1: "invalidParams" attribute is included in the "ProblemDetails" data structure indicating missing or incorrect IE.

NOTE 2: This application error indicates error in the HTTP request and there is no other application error value that can be used instead.

NOTE 3: This application error indicates error condition in the NF and there is no other application error value that can be used instead.

NOTE 4: If the reason for rejection is a temporary overload, the NF may include in the response a Retry-After header field to indicate how long the service is expected to be unavailable.

NOTE 5: If the request is rejected because of an error in the URI before the first variable part of "apiSpecificResourceUriPart", then the "404 Not Found" HTTP status code may be sent without the "ProblemDetails" data structure indicating protocol or application error.

B

Error Code Generated by SCP

This section describes HTTP error responses generated by SCP:

Table B-1 SCP Generated Error Responses

HTTP Status Code	Sample Error Scenarios at SCP
400	When SCP receives a request with malformed URL (For example, % is added URL after apiRoot).
400 Not Found	When SCP receives request with the 3gpp-sbi-routing-binding header and an invalid version.
400 Bad Request	When SCP receives a request with the malformed 3GPP-SBI-routing-binding header.
411 Length Required	Not Applicable
413 Payload Too Large	Not Applicable
429 Too Many Requests	When the ingress rate exceeds the configured limit for a given consumer, SCP rejects the additional messages with the error code.
431 Request Header Fields Too Large	When SCP receives a request with the total length of the headers are larger than 8k.
500 Internal Error	- Sample 1: When payload size content length exceeds the 256 KB threshold limit. - Sample 2: When the response is bigger than configured. <code>maxPayloadSizeInKb</code>. - Sample 3: When SCP receives a request with incorrect content-length header field.
503 NF_CONGESTION	- Sample 1: When the producer is overloaded and no alternates are available. - Sample 2: When the SCP is overloaded based on configured thresholds.
503 Service Unavailable	When SCP receives a request with the 3gpp-sbi-routing-binding header and an invalid API name.
504 Gateway Timeout	When no response is received from a producer for the configured response timeout value.
508 Loop Detected	When SCP receives a via header with its own FQDN or IP.