

Oracle® Communications

Cloud Native Core, Security Edge Protection Proxy Troubleshooting Guide



Release 25.1.102

G20972-03

July 2025

ORACLE®

Copyright © 2021, 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	Overview	1
1.1.1	References	1
2	Logs	
2.1	Log Levels	1
2.2	Collecting Logs	1
2.3	Using Logs	2
3	Troubleshooting SEPP	
3.1	Generic Troubleshooting Scenarios	1
3.1.1	Generic Checklist	1
3.1.2	The environment is not working as expected	2
3.1.3	Kubernetes Node Failure	2
3.1.4	SEPP Installation Verification	3
3.1.5	Debugging General CNE	4
3.1.6	Collecting the SEPP Logs to Check the Error Scenarios	4
3.1.7	Helm Error During the Rollback	4
3.1.8	Upgrade or Rollback Failure	5
3.1.9	Helm Test Failure	5
3.1.10	Helm Rollback Failure with the Configmap with the Name not Found Error	6
3.1.11	Continuous Restart of coherence-svc Pods	6
3.1.12	IllegalReferenceCount Exception Occurrence in Logs of Ingress and Egress Gateways	6
3.1.13	False Message while Doing the Helm Uninstall	7
3.2	Feature Specific Troubleshooting Scenarios	7
3.2.1	Cat-2 Network ID Validation Feature	7
3.2.2	Cat-1 Service API Validation Feature	9
3.2.3	Overload Control Feature	9
3.2.4	Troubleshooting Steps for Rate Limiting Feature	10
3.2.5	Message Feed Feature	11
3.2.6	Hosted SEPP	13

3.2.7	Steering of Roaming (SOR) Feature	14
3.2.8	Rate Limiting for Ingress Roaming Signaling per Remote SEPP Set Feature	15
3.2.9	Cat-3 Previous Location Check feature	16
3.2.10	Cat-3 Time check for Roaming Subscribers	16
3.2.11	Cat-0 SBI Message Schema Validation Feature	19
3.2.12	Configuration Failure in Remote SEPP and Remote SEPP Set	21
3.2.13	Aspen Service Mesh	22
3.2.14	Rate Limiting for Egress Roaming Signaling per PLMN feature	24
3.2.15	Separate Port Configurations for N32c and N32f on the Egress Routes	25
3.2.16	Alternate Routing based on the DNS SRV Record for Home Network Functions	26
3.2.17	Load Sharing among Multiple Remote SEPP Nodes	27
3.2.18	5G SBI Message Mediation Support	29
3.2.19	Support for TLS 1.3	29
3.2.20	SEPP Deployment on OCI	30
3.2.21	Georedundancy Support	30
3.2.22	Support for Originating Network Id Header Validation, Insertion, and Transposition	32
3.2.23	Proactive status updates on SEPP	32
3.2.24	Multiple SEPP instances on Shared cnDBTier Cluster	33
3.2.25	Cat-1 NRF Service API Query Parameters Validation Feature	34
3.2.26	Integrating SEPP with 5G Network Intelligence Fabric (5GNIF) feature	36
3.2.27	LCI and OCI Header Support Feature	42
3.3	HTTP Response Codes and Error Codes	43

4 Debug Tool

4.1	Debug Tool Configuration Parameters	13
-----	-------------------------------------	----

Preface

- [Documentation Accessibility](#)
- [Diversity and Inclusion](#)
- [Conventions](#)

Documentation Accessibility

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

Access to Oracle Support

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

Diversity and Inclusion

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

Conventions

The following text conventions are used in this document:

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

My Oracle Support

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

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

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

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

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

Acronyms

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

Table Acronyms and Terminologies

Acronym	Description
CRD	Custom Resource Definition
CNE	Cloud Native Environment
C SEPP/C-SEPP	Consumer Security Edge Protection Proxy
DNS	Domain Name System
DD	Data Director
EGW	Egress Gateway
FQDN	Fully Qualified Domain Name
Hosted SEPP	Hosted SEPP functionality provides selective routing in Roaming Hub Mode
IGW	Ingress Gateway
IPX	Internetwork Packet Exchange
K8s	Kubernetes
Local PLMN	PLMN managed by Local SEPP
Local SEPP	SEPP in Local PLMN
MCC	Mobile Country Codes
MNC	Mobile Network Codes
MNO	Mobile Network Operator
NDB	Network Database
NF	Network Function
Network Function	A functional building block within a network infrastructure, which has well defined external interfaces and well defined functional behavior. In practical terms, a network function is often a network node or physical appliance.
NF Consumer	A generic way to refer to an NF which consumes services provided by another NF. Example: An AMF acts as a Consumer NF that consumes AMPolicy services provided by the PCF.
NF Instance	A specific instance of a network function type.
NF Producer or NF Provider	A generic way to refer to an NF which provides services that can be consumed by another NF. Example: A PCF acts as a Producer NF that provides AMPolicy Services to the AMF.
NRF	Network Repository Function
OCI	Oracle Cloud Infrastructure
OCIR	Oracle Cloud Infrastructure Registry
OHC	Oracle Help Center
OKE	Oracle Engine for Kubernetes
OTEL	OpenTelemetry
OSDC	Oracle Software Delivery Cloud
PDB	PodDisruptionBudget
PLMN	Public Land Mobile Network

Table (Cont.) Acronyms and Terminologies

Acronym	Description
P SEPP/P-SEPP	Producer Security Edge Protection Proxy
Remote PLMN	PLMN managed by Remote SEPP
Remote SEPP	SEPP in Remote PLMN
Remote SEPP Set	Set of Remote SEPPs to allow alternate routing across Remote SEPPs
REST API	Representational State Transfer Application Programming Interface
Roaming Hub	Roaming Hub is the deployment mode of SEPP. Roaming Hub is used as an intermediate proxy. Each SEPP connects to the Roaming Hub which further connect to another SEPP. All the Remote SEPPs can talk with each other through roaming hub.
Scaling	Ability to dynamically extend or reduce resources granted to the Virtual Network Function (VNF) as needed. This includes scaling out and in or scaling up and down.
SEPP	Security Edge Protection Proxy
SUPI	Subscription Permanent Identifier
SOR	Steering Of Roaming
SVC	Service
TLS	Transport Layer Security
TH	Topology Hiding
TUH	Topology Unhiding

What's New in This Guide

This section introduces the documentation updates for Release 25.1.1xx.

Release 25.1.102 - G20972-03, July 2025

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

Release 25.1.101 - G20972-02, June 2025

- Added the following alerts in the [System Level Alerts](#) section:

- ocseppPodsRestart
- ocseppAppinfoServiceDown
- ocseppCn32cServiceDown
- ocseppCn32fServiceDown
- ocseppConfigMgrServiceDown
- ocseppN32EgwServiceDown
- ocseppN32IgwServiceDown
- ocseppConfigserverServiceDown
- ocseppPerfinfoServiceDown
- ocseppPlmnEgwServiceDown
- ocseppPlmnIgwServiceDown
- ocseppPn32cServiceDown
- ocseppPn32fServiceDown
- ocseppNrfdiscServiceDown
- ocseppNrfgmgServiceDown
- ocseppAlternrouteServiceDown
- ocseppMediationServiceDown
- ocseppCoherenceServiceDown
- ocseppNfStatusUnavailable

Release 25.1.100 - G20972-01, April 2025

- Added the troubleshooting scenarios related to the [Cat-3 Time check for Roaming Subscribers](#) feature.
- Updated the troubleshooting scenarios related to the [Message Feed](#) feature.
- Updated the troubleshooting scenarios related to the [Aspen Service Mesh](#) feature.
- Added the log attribute details of following microservices in the [Logs](#) section:
 - nrf-client-nfmanagement
 - nrf-client-nfdiscovery
 - app-info

- perf-info
- config-server
- Added the following feature specific alerts in the [Cat-3 Time check for Roaming Subscribers](#) alerts section:
 - pn32fTimeUnauthLocChkValFailAlrtMinor
 - pn32fTimeUnauthLocChkValFailAlrtMajor
 - pn32fTimeUnauthLocChkValFailAlrtCritical
 - pn32fTimeUnauthLocChkExcepFailAlrtMinor
 - pn32fTimeUnauthLocChkExcepFailAlrtMajor
 - pn32fTimeUnauthLocChkExcepFailAlrtCritical
- Updated the name of the following alerts:
 - From ConfigManagerConnectionFailureWithDatabaseAlert to cfgMgrConnFailureWithDBAlert
 - From SEPPnPn32fPreviousLocationCheckValidationFailureAlertCritical to sEPPnPn32fPrevLocChkValFailAlrtCritical
 - From SEPPnPn32fPreviousLocationCheckExceptionFailureAlertCritical to EPPnPn32fPrevLoChkExcepFailAlrtCritical
 - From Egress Request Rate Limit per PLMN Message Drop Above 10 Percent Alert to EgressRequestRateLimitDropAbove10Percent
 - From Egress Request Rate Limit per PLMN Message Drop Above 25 Percent Alert to EgressRequestRateLimitDropAbove25Percent
 - From Egress Request Rate Limit per PLMN Message Drop Above 50 Percent Alert to EgressRequestRateLimitDropAbove50Percent
 - From Ingress RSS Rate Limit per RSS Message Drop Above Point one Percent Alert to IngressRssRateLimitPerRSSMessageDropAbovePointOnePercent
 - From Ingress RSS Rate Limit per RSS Message Drop Above 10 Percent Alert to IngressRssRateLimitPerRSSMessageDropAbove10Percent
 - From Ingress RSS Rate Limit per RSS Message Drop Above 25 Percent Alert to IngressRssRateLimitPerRSSMessageDropAbove25Percent
 - From Ingress RSS Rate Limit per RSS Message Drop Above 50 Percent Alert to IngressRssRateLimitPerRSSMessageDropAbove50Percent
 - From Ingress RSS Rate Limit Message Drop Above Point one Percent Alert to IngressRssRateLimitMessageDropAbovePointOnePercent
 - From Ingress RSS Rate Limit Message Drop Above one Percent Alert to IngressRssRateLimitMessageDropAboveOnePercent
 - From Ingress RSS Rate Limit Message Drop Above 10 Percent Alert is to IngressRssRateLimitMessageDropAbove10Percent
 - From Ingress RSS Rate Limit Message Drop Above 25 Percent Alert is to IngressRssRateLimitMessageDropAbove25Percent
 - From Ingress RSS Rate Limit Message Drop Above 50 Percent Alert to IngressRssRateLimitMessageDropAbove50Percent

1

Introduction

This document provides information about troubleshooting Oracle Communications Security Edge Protection Proxy (SEPP).

1.1 Overview

Security Edge Protection Proxy (SEPP) is a key component of the 5G Service Based Architecture. It is a proxy Network Function (NF) which is used for the secured communication for inter Public Land Mobile Network (PLMN) messages.

For more information about the SEPP architecture, see *Oracle Communications Cloud Native Core, Security Edge Protection Proxy User Guide*.

The user can install either SEPP or Roaming Hub/Hosted SEPP.

Note

The performance and capacity of the SEPP system may vary based on the call model, Feature/Interface configuration, and underlying CNE and hardware environment.

1.1.1 References

Following are the reference documents:

- *Oracle Communications Cloud Native Core, Cloud Native Environment Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, cnDBTier User Guide*
- *Oracle Communications Cloud Native Core, cnDBTier Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Data Collector User Guide*
- *Oracle Communications Cloud Native Core, Security Edge Protection Proxy REST Specification Guide*
- *Oracle Communications Cloud Native Core, Security Edge Protection Proxy Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Security Edge Protection Proxy User Guide*
- *Oracle Communications Cloud Native Core, Security Edge Protection Proxy Network Impact Report*
- *Oracle Communications Cloud Native Configuration Console User Guide*

2

Logs

This chapter explains the process to retrieve the logs and status that can be used for effective troubleshooting.

2.1 Log Levels

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

A log level helps in defining the severity level of a log message. For OCSEPP, the log level of a microservice can be set to any one of the following valid values:

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

Using this information, the logs can be filtered based on the system requirements. For instance, if you want to filter the critical information about your system from the informational log messages, set a filter to view messages with only WARN log level in Kibana.

2.2 Collecting Logs

This section describes the steps to collect logs from PODs and containers. Perform the following steps:

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

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

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

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

From the container:

```
$ kubectl logs <podname> -c <container name> -n <namespace>
```

Example:

From the pod:

```
$ kubectl logs ocsepp-release-xxxxxxxxxx-xxxxx -n seppsvc
```

From the container:

```
$ kubectl logs ocsepp-release-n32-egress-gateway-xxxxx -c n32-egress-gateway -n  
seppsvc
```

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

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

Example:

```
$ kubectl logs ocsepp-release-xxxxxxxxxx-xxxxx -n seppsvc > logs.txt
```

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

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

Example:

```
$ kubectl logs ocsepp-release-xxxxxxxxxx-xxxxx -n seppsvc -f --tail 100 >  
logs.txt
```

For more information on kubectl commands, see [Kubernetes website](#).

2.3 Using Logs

This section explains the logs you need to look at, to handle different SEPP debugging issues.

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

This section provides log level attribute details for the following services:

- config-mgr-svc
- cn32c-svc
- pn32c-svc
- cn32f-svc
- pn32f-svc

- nrf-client-nfmanagement
- nrf-client-nfdiscovery
- app-info
- perf-info
- config-server
- n32-ingress-gateway
- n32-egress-gateway
- plmn-ingress-gateway
- plmn-egress-gateway
- nf-mediation

Sample Logs

config-mgr-svc

Sample log statement for config-mgr-svc:

```
{"instant":
{"epochSecond":1636703617,"nanoOfSecond":449636327},"thread":"XNIO-1
task-4","level":"DEBUG","loggerName":"org.springframework.web.servlet.Dispatch
erServlet","message":"Completed 200
OK","endOfBatch":false,"loggerFqcn":"org.apache.commons.logging.LogAdapter$Log
4jLog","threadId":40,"threadPriority":5,"ts":"21-11-12
07:53:449.037+0000","instanceType":"prod","processId":"1","ocLogId":"$
{ctx:ocLogId}","vendor":"oracle"}
```

cn32c-svc

Sample log statement for cn32c-svc:

```
{"instant":{"epochSecond":1636456524,"nanoOfSecond":315917989},"thread":"sepp-
cn32c-
thread-2","level":"DEBUG","loggerName":"com.oracle.cgbu.cne.ocsepp.client.Http
2Client","message":"Http2 Client trying to connect with URL: http://ocsepp-
release-config-mgr-svc:9090/cn32c/Handshake-
success","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.Abstrac
tLogger","threadId":27,"threadPriority":5,"ts":"21-11-09
11:15:315.024+0000","instanceType":"prod","processId":"1","ocLogId":"$
{ctx:ocLogId}","vendor":"oracle"}
```

pn32c-svc

Sample log statement for pn32c-svc:

```
{"instant":
{"epochSecond":1636455735,"nanoOfSecond":301984153},"thread":"main","level":"I
NF0","loggerName":"com.oracle.cgbu.cne.ocsepp.pn32c.Pn32cApplication","message
":"Starting Pn32cApplication using Java 16.0.1 on ocsepp-release-pn32c-
svc-7fb7d866c6-sczjg with PID 1(/ocsepp-pn32c-svc.jar started by seppuser
in /)","endOfBatch":false,"loggerFqcn":"org.apache.commons.logging.LogAdapter$
Log4jLog","threadId":1,"threadPriority":5,"ts":"21-11-09
11:02:301.015+0000","instanceType":"prod","processId":"1","ocLogId":"$
{ctx:ocLogId}","vendor":"oracle"}
```

cn32f-svc

Sample log statement for cn32f-svc:

```
{
  "instant":
  {
    "epochSecond":1636457129,"nanoOfSecond":526138937},
    "thread":"main","level":"INFO",
    "loggerName":"com.oracle.cgbu.cne.ocsepp.cn32f.Cn32fApplication",
    "message":"Starting Cn32fApplication using Java 16.0.1 on ocsepp-release-cn32f-svc-9b8c6d7c6-dgmqv with PID 1 (/ocsepp-cn32f-svc.jar started by seppuser in /)",
    "endOfBatch":false,"loggerFqcn":"org.apache.commons.logging.LogAdapter$Log4jLog",
    "threadId":1,"threadPriority":5,"ts":"21-11-09 11:25:526.029+0000",
    "instanceType":"prod","processId":"1","ocLogId":"${ctx:ocLogId}",
    "vendor":"oracle"}
}
```

pn32f-svc

Sample log statement for pn32f-svc:

```
{
  "instant":
  {
    "epochSecond":1636457721,"nanoOfSecond":692849682},
    "thread":"main","level":"INFO",
    "loggerName":"com.oracle.cgbu.cne.ocsepp.pn32f.Pn32fApplication",
    "message":"Starting Pn32fApplication using Java 16.0.1 on ocsepp-release-pn32f-svc-85b4b9fd9d-gzph with PID 1 (/ocsepp-pn32f-svc.jar started by seppuser in /)",
    "endOfBatch":false,"loggerFqcn":"org.apache.commons.logging.LogAdapter$Log4jLog",
    "threadId":1,"threadPriority":5,"ts":"21-11-09 11:35:692.021+0000",
    "instanceType":"prod","processId":"1","ocLogId":"${ctx:ocLogId}",
    "vendor":"oracle"}
}
```

Log Attribute Details for n32f, n32c, and config-mgr-svc

Table 2-1 Log Attribute Details for n32f, n32c, and config-mgr-svc

Log Attribute	Details	Sample Value	Data Type
instant	Epoch time. It is the group of two values epochSecond and nanoOfSecond	{"epochSecond":1604655402,"nanoOfSecond":946649000}	Object
thread	Logging Thread Name	"reactor-http-epoll-2"	String
level	Log Level of the log printed	"DEBUG"	String
loggerName	Class or module which printed the log	"com.oracle.cgbu.cne.ocsepp.pn32f.iointerface.Pn32fSeppAsyncInterface"	String
message	Message related to the log providing brief details. Indicates that no NFProfiles found for mentioned search query	"{LoggingRequestDecorator::getBody() Query target-nf-type=AUSF&requester-nf-type=SEPP}"	String
endOfBatch	Log4j2 Internal Default from log4j2: false	false	Boolean
loggerFqcn	Log4j2 Internal Fully Qualified class name of logger module	org.apache.logging.log4j.spi.AbstractLogger	String
threadId	Thread Id generated internally by Log4j2	32	Integer

Table 2-1 (Cont.) Log Attribute Details for n32f, n32c, and config-mgr-svc

Log Attribute	Details	Sample Value	Data Type
threadPriority	Thread Priority set internally by Log4j2	5	Integer
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ Timestamp can be filtered using the following label :ts -> logs in containertimestamp -> logs on Kibana	"messageTimestamp":"2023-09-01T03:01:24.607+0000"	String
instanceType	Instance details. Example: dev, prod, qa. Note: Part of container logs but not in Kibana.	prod	String
processId	Process ID internally assigned. Note: Part of container logs but not in Kibana.	"1"	String
ocLogId	End to End Log Identifier across the OCSEPP microservices. Helps to correlate the logs across the microservices in OCSEPP application. Note: Part of container logs but not in Kibana.	\${ctx:ocLogId}	String
vendor	Vendor Name	"oracle"	String

nrf-client-nfmanagement

Sample log statement for nrf-client-nfmanagement:

```
{
  "instant": {
    "epochSecond": 1653141225,
    "nanoOfSecond": 57167090,
    "thread": "taskScheduler-2",
    "level": "WARN",
    "loggerName": "com.oracle.cgbu.cnc.nrf.NRFManagement",
    "message": "NfServices is not present inNfProfile.",
    "endOfBatch": false,
    "loggerFqcn": "org.apache.logging.slf4j.Log4jLogger",
    "threadId": 27,
    "threadPriority": 5,
    "source": {
      "class": "com.oracle.cgbu.cnc.nrf.NRFManagement",
      "method": "setPerformance",
      "file": "NRFManagement.java",
      "line": 1758,
      "messageTimestamp": "2022-05-21T13:53:45.057+0000"
    }
  }
}
```

Log Attribute Details for nrf-client-nfmanagement**Table 2-2 Log Attribute Details for nrf-client-nfmanagement**

Log Attribute	Details	Sample Value	Data Type
instant	Epoch time. It is the group of two values epochSecond and nanoOfSecond	{"epochSecond":1653141225,"nanoOfSecond":57167090}	Object
thread	Logging Thread Name	"taskScheduler-2"	String

Table 2-2 (Cont.) Log Attribute Details for nrf-client-nfmanagement

Log Attribute	Details	Sample Value	Data Type
level	Log Level of the log printed	"WARN"	String
loggerName	Class or module which printed the log	"com.oracle.cgbu.cnc.nrf.NRFManagement"	String
message	Message related to the log providing brief details.	"NfServices is not present in NfProfile"	String
endOfBatch	Log4j2 Internal Default from log4j2: false	false	Boolean
loggerFqdn	Log4j2 Internal Fully Qualified class name of logger module	org.apache.logging.slf4j.Log4jLogger	String
threadId	Thread Id generated internally by Log4j2	1	Integer
threadPriority	Thread Priority set internally by Log4j2	5	Integer
Source	Source code details including class name, method name, file name and line number	{"class": "com.oracle.cgbu.cnc.nrf.NRFManagement", "method": "setPerformance", "file": "NRFManagement.java", "line": 1758 }	Object
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ	"2022-05-21T13:53:45.057+0000"	String

nrf-client-nfdiscovery

Sample log statement for nrf-client-nfdiscovery:

```
{
  "instant": {
    "epochSecond": 1653141021,
    "nanoOfSecond": 819399951
  },
  "thread": "main",
  "level": "WARN",
  "loggerName": "com.oracle.cgbu.cnc.nrf.util.NrfClientProperties",
  "message": "getHttpsProxyPort(): Invalid httpsProxyPort",
  "endOfBatch": false,
  "loggerFqcn": "org.apache.logging.slf4j.Log4jLogger",
  "threadId": 1,
  "threadPriority": 5,
  "source": {
    "class": "com.oracle.cgbu.cnc.nrf.util.NrfClientProperties",
    "method": "getHttpsProxyPort",
    "file": "NrfClientProperties.java",
    "line": 260
  },
  "messageTimestamp": "2022-05-21T13:50:21.819+0000"
}
```

Log Attribute Details for nrf-client-nfdiscovery

Table 2-3 Log Attribute Details for nrf-client-nfdiscovery

Log Attribute	Details	Sample Value	Data Type
instant	Epoch time. It is the group of two values epochSecond and nanoOfSecond	{"epochSecond":1653141021,"nanoOfSecond":819399951}	Object
thread	Logging Thread Name	"main"	String
level	Log Level of the log printed	"WARN"	String
loggerName	Class or module which printed the log	"com.oracle.cgbu.cnc.nrf.util.NrfClientProperties"	String
message	Message related to the log providing brief details.	"getHttpsProxyPort():Invalid httpsProxyPort"	String
endOfBatch	Log4j2 Internal Default from log4j2: false	false	Boolean
loggerFqdn	Log4j2 Internal Fully Qualified class name of logger module	org.apache.logging.slf4j.Log4jLogger	String
threadId	Thread Id generated internally by Log4j2	1	Integer
threadPriority	Thread Priority set internally by Log4j2	5	Integer
Source	Source code details including class name, method name, file name and line number	{"class": "com.oracle.cgbu.cnc.nrf.util.NrfClientProperties", "method": "getHttpsProxyPort", "file": "NrfClientProperties.java", "line": 260}	Object
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ	"2022-05-21T13:50:21.819+0000"	String

app-info

Sample log statement for app-info

```
{
  "name": "unicorn.access",
  "message": "::ffff:10.244.1.106 - - [21/May/2022:13:54:29 +0000] \"GET /status/category/sepp HTTP/1.1\" 200 7 \"-\" \"okhttp/3.14.9\"",
  "level": "INFO",
  "filename": "glogging.py",
  "lineno": 349,
  "module": "glogging",
  "func": "access",
  "thread": "MainThread",
  "messageTimestamp": "2022-05-21T13:54:29.385+0000"}

```

Sample log statement - perf-info

Log Attribute Details for app-info

Table 2-4 Log Attribute Details for app-info

Log Attribute	Details	Sample Value	Data Type
thread	Logging Thread Name	"MainThread"	String
name	module name	"unicorn.access"	String
message	Message related to the log providing brief details.	"::ffff:10.244.1.106 - - [21/May/2022:13:54:29 +0000] \"GET /status/category/sepp HTTP/1.1\" 200 7 \"-\" \"okhttp/3.14.9\""	String
level	Log Level of the log printed	"INFO"	String
filename	Name of the file	"glogging.py"	String
lineno	line number of the execution step	349	Integer
module	name of the module	"glogging"	String
func	name of the executing function	"access"	String
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ	"2022-05-21T13:54:29.385+0000"	String

perf-info

Sample log statement for perf-info

```
{
  "name": "stat_helper",
  "message": "Failed to reach prometheus",
  "level": "ERROR",
  "filename": "stat_helper.py",
  "lineno": 106,
  "module": "stat_helper",
  "func": "get_db_param",
  "thread": "MainThread",
  "messageTimestamp": "2022-05-21T13:57:39.639+0000"
}
```

Log Attribute Details for perf-info**Table 2-5 Log Attribute Details for perf-info**

Log Attribute	Details	Sample Value	Data Type
thread	Logging Thread Name	"MainThread"	String
name	module name	"stat_helper"	String
message	Message related to the log providing brief details.	"Failed to reach prometheus"	String
level	Log Level of the log printed	"ERROR"	String
filename	Name of the file	"stat_helper.py"	String
lineno	line number of the execution step	106	Integer
module	name of the module	"stat_helper"	String

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

Log Attribute	Details	Sample Value	Data Type
func	name of the executing function	"get_db_param"	String
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ	"2022-05-21T13:57:39.639+0000"	String

config-server

Sample log statement for config-server

```
{
  "instant": {
    "epochSecond": 1653140996,
    "nanoOfSecond": 895472496,
    "thread": "main",
    "level": "INFO",
    "loggerName": "ocpm.cne.common.metrics.cgroup.CgroupMetricsHelper",
    "message": "Creating cgroup metricfinder",
    "endOfBatch": false,
    "loggerFqn": "org.apache.logging.slf4j.Log4jLogger",
    "threadId": 1,
    "threadPriority": 5,
    "messageTimestamp": "2022-05-21T13:49:56.895+0000"
  }
}
```

Log Attribute Details for config-server**Table 2-6 Log Attribute Details for config-server**

Log Attribute	Details	Sample Value	Data Type
thread	Logging Thread Name	"main"	String
instant	Epoch time. It is the group of two values epochSecond and nanoOfSecond	{"epochSecond": 1653140996, "nanoOfSecond": 895472496}	Object
level	Log Level of the log printed	"INFO"	String
loggerName	Class or module which printed the log	"ocpm.cne.common.metrics.cgroup.CgroupMetricsHelper"	String
message	Message related to the log providing brief details.	"Creating cgroup metricfinder"	String
endOfBatch	Log4j2 Internal Default from log4j2: false	false	Boolean
loggerFqn	Log4j2 Internal Fully Qualified class name of logger module	org.apache.logging.slf4j.Log4jLogger	String
threadId	Thread Id generated internally by Log4j2	1	Integer

Table 2-6 (Cont.) Log Attribute Details for config-server

Log Attribute	Details	Sample Value	Data Type
threadPriority	Thread Priority set internally by Log4j2	5	Integer
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ	"2022-05-21T13:49:56.895+0000"	String

Sample Logs for Ingress Gateway

This section provides log level attribute details for following service:

- n32-ingress-gateway
- plmn-ingress-gateway

Sample log statement n32-ingress-gateway:

```
{
  "instant": {
    "epochSecond": 1643968884, "nanoOfSecond": 549874972, "thread": "ingress-h2-epoll-2",
    "level": "DEBUG", "loggerName": "org.springframework.cloud.gateway.handler.RoutePredicateHandlerMapping",
    "message": "Route matched: n32c2", "endOfBatch": false, "loggerFqcn": "org.apache.commons.logging.LogAdapter$Log4jLog",
    "contextMap": {
      "ocLogId": "1643968884534_142_ocsepp-release-chandra-n32-ingress-gateway-6dfb6fc446-9phs6"
    }, "threadId": 142, "threadPriority": 5, "messageTimestamp": "2022-02-04T10:01:24.549+0000",
    "ocLogId": "1643968884534_142_ocsepp-release-chandra-n32-ingress-gateway-6dfb6fc446-9phs6",
    "pod": "${ctx:hostname}", "processId": "1", "instanceType": "prod", "ingressTxId": "${ctx:ingressTxId}"
  }
}
```

Sample log statement plmn-ingress-gateway:

```
{
  "instant": {
    "epochSecond": 1643971167, "nanoOfSecond": 783195870, "thread": "pool-11-thread-6",
    "level": "DEBUG", "loggerName": "com.oracle.common.scheduler.ReloadConfig",
    "message": "Config server URL: http://ocsepp-release-chandra-config-mgr-svc:9090/config/igw/plmn/22.2.2/1",
    "endOfBatch": false, "loggerFqcn": "org.apache.logging.log4j.spi.AbstractLogger",
    "contextMap": {
      "threadId": 79, "threadPriority": 5, "messageTimestamp": "2022-02-04T10:39:27.783+0000",
      "ocLogId": "${ctx:ocLogId}", "pod": "${ctx:hostname}", "processId": "1", "instanceType": "prod",
      "ingressTxId": "${ctx:ingressTxId}"
    }
  }
}
```

Table 2-7 Log Attribute Details for Ingress Gateway

Log Attribute	Details	Sample Value	Data Type
thread	Logging Thread Name	"ingress-h2c-epoll-3"	String

Table 2-7 (Cont.) Log Attribute Details for Ingress Gateway

Log Attribute	Details	Sample Value	Data Type
level	Log Level of the log printed	"DEBUG"	String
loggerName	Class/Module which printed the log	"ocpm.cne.gateway.filters.PreGatewayFilter"	String
message	Message related to the log providing brief details. Indicates that the method PreGatewayFilter is being exited.	"Exiting PreGatewayFilter"	String
endOfBatch	Log4j2 Internal Default from log4j2: false	false	boolean
loggerFqn	Log4j2 Internal Fully Qualified class name of logger module	org.apache.logging.log4j.spi.AbstractLogger	String
instant	Epoch timestamp It is group of two values epochSecond and nanoOfSecond	{"epochSecond":1604650229,"nanoOfSecond":4993000}	Object
contextMap	contents of log4j ThreadContext map	{"hostname":"ocsepp-ingressgateway-69f6544b8d-cdbgx", "ingressTxId":"ingress-tx-1087436877", "ocLogId":"160465022902_72_ocsepp-ingressgateway-69f6544b8d-cdbgx"}	Object
threadId	Thread Id generated internally by Log4j2	72	Integer
threadPriority	Thread Priority set internally by Log4j2	5	Integer
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ	"2020-11-06 08:10:29.004"	String
ocLogId	End to End Log Identifier across the OCSEPP microservices. Helps to correlate the logs across the microservices	"1604650229002_72_ocsepp-ingressgateway-69f6544b8d-cdbgx"	String
pod	Pod Name	"ocsepp-ingressgateway-69f6544b8d-cdbgx"	String
processId	Process ID internally assigned	"1"	String
instanceType	Instance type	"prod"	String

Table 2-7 (Cont.) Log Attribute Details for Ingress Gateway

Log Attribute	Details	Sample Value	Data Type
ingressTxId	Transaction id that is added to log4j ThreadContext map and is unique to every transaction	"ingress-tx-1087436877"	String

Egress Gateway

This section provides log level attribute details for following services:

- n32-egress-gateway
- plmn-egress-gateway

Sample log statement n32-egress-gateway:

```
{
  "instant":
  {
    "epochSecond":1643968532, "nanoOfSecond":801113787, "thread":"scheduling-1", "level":"DEBUG", "loggerName":"ocpm.cne.gateway.config.DynamicRouteConfiguration", "message":"Validated the following route successfully: RoutesConfiguration [id=n32d, uri=https://ocsepp.com, order=81, predicates=[PredicateDefinition{name='Path', args={pattern=/*/n32c-handshake/**}}, filters=null, metadata={}, httpRuriOnly=null, httpsTargetOnly=null, sbiRoutingConfiguration=null]", "endOfBatch":false, "loggerFqcn":"org.apache.logging.log4j.spi.AbstractLogger", "contextMap": {}, "threadId":72, "threadPriority":5, "messageTimestamp":"2022-02-04T09:55:32.801+0000", "ocLogId":"${ctx:ocLogId}", "pod":"${ctx:hostname}", "processId":"1", "instanceType":"prod", "egressTxId":"${ctx:egressTxId}"
  }
}
```

Sample log statement plmn-egress-gateway:

```
{
  "instant":
  {
    "epochSecond":1643971148, "nanoOfSecond":705331370, "thread":"scheduling-1", "level":"INFO", "loggerName":"com.oracle.common.metrics.ConfigClientMetrics", "message":"Pegged ConfigClient Response metric with releaseVersion 22.2.2, configVersion 1 and updated parameter false", "endOfBatch":false, "loggerFqcn":"org.apache.logging.log4j.spi.AbstractLogger", "contextMap": {}, "threadId":72, "threadPriority":5, "messageTimestamp":"2022-02-04T10:39:08.705+0000", "ocLogId":"${ctx:ocLogId}", "pod":"${ctx:hostname}", "processId":"1", "instanceType":"prod", "egressTxId":"${ctx:egressTxId}"
  }
}
```

Table 2-8 Log Attribute Details for Egress Gateway

Log Attribute	Details	Sample Value	Data Type
thread	Logging Thread Name	"main"	String
level	Log Level of the log printed	"DEBUG"	String
loggerName	Class/Module which printed the log	"ocpm.cne.gateway.config.DynamicRouteConfiguration"	String
message	Message related to the log providing brief details	"Property name: server.port and value: 8080"	String
endOfBatch	Log4j2 Internal Default from log4j2: false	false	boolean
loggerFqn	Log4j2 Internal Fully Qualified class name of logger module	org.apache.logging.log4j.spi.AbstractLogger	String
instant	Epoch timestamp. It is group of two values epochSecond and nanoOfSecond	{"epochSecond":1604564777,"nanoOfSecond":135977000}	Object
contextMap	Elements in log4j ThreadContext map	{}	Object
threadId	Thread Id generated internally by Log4j2	1	Integer
threadPriority	Thread Priority set internally by Log4j2	5	Integer
messageTimestamp	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ	"2020-11-05 08:26:17.135"	String
ocLogId	End to End Log Identifier across the OCSEPP microservices. Helps to correlate the logs across the microservices in OCSEPP application	"1604650229002_72_ocsepp-ingressgateway-69f6544b8d-cdbgx"	String
pod	Name of the egress pod	"ocsepp-egressgateway-69f6544b8d-cdbgx"	String
processId	Process ID internally assigned	"1"	String
instanceType	Instance type	"prod"	String
egressTxId	Transaction id that is added to log4j ThreadContext map and is unique to every transaction	"egress-tx-1087436877"	String

Nf-Mediation service

This section provides log level attribute details for mediation service:

Sample log statement mediation service:

```
{
  "instant": {
    "epochSecond": 1661413301, "nanoOfSecond": 856094544, "thread": "Thread-0", "level": "RULE_TRAIL", "loggerName": "com.oracle.cgbu.ocmediation.ruleengine.DroolsRuleEngine", "message": "Mediation Rule files reloading successful", "endOfBatch": false, "loggerFqcn": "org.apache.logging.log4j.spi.AbstractLogger", "threadId": 15, "threadPriority": 5, "ts": "22-08-25 07:41:856.041+0000", "namespace": "gwnrf", "node_name": "cnejac0106.jacvla.morrisville.us.lab.oracle.com", "pod": "ocsepp-release-seppsvc-nf-mediation-7679c47c77-zbqd2", "instanceType": "prod", "processId": "1", "ocLogId": "$ {ctx:ocLogId}", "vendor": "oracle"}
}
```

Log Attribute Details for Nf-Mediation service

Table 2-9 Log Attribute Details for Nf-Mediation service

Log Attribute	Details	Sample Value	Data Type
instant	Epoch time. It is the group of two values epochSecond and nanoOfSecond	{"epochSecond":1604655402, "nanoOfSecond":946649000}	Object
thread	Logging Thread Name	" Thread-0"	String
level	Log Level of the log printed	"RULE_TRAIL"	String
loggerName	Class or module which printed the log	"com.oracle.cgbu.ocmediation.ruleengine.DroolsRuleEngine"	String
message	Message related to the log providing brief details.	"Mediation Rule files reloading successful"	String
endOfBatch	Log4j2 Internal Default from log4j2: false	false	Boolean
loggerFqcn	Log4j2 Internal Fully Qualified class name of logger module	org.apache.logging.log4j.spi.AbstractLogger	String
threadId	Thread Id generated internally by Log4j2	15	Integer
threadPriority	Thread Priority set internally by Log4j2	5	Integer
namespace	Namespace for which log is generated	"gwnrf"	
node_name	Name of the worker node on which this pod is allocated	"cnejac0106.jacvla.morrisville.us.lab.oracle.com"	
pod	Name of the pod which generated these logs	"ocsepp-release-seppsvc-nf-mediation-7679c47c77-zbqd2"	

Table 2-9 (Cont.) Log Attribute Details for Nf-Mediation service

Log Attribute	Details	Sample Value	Data Type
ts	Timestamp of log from application container. Format: yyyy-MM-dd'T'HH:mm:ss.SSSZ Timestamp can be filtered using the following label :ts -> logs in containertimestamp -> logs on Kibana	"ts":"22-02-04 03:49:162.039+0000"	String
instanceType	Instance details. Example: dev, prod, qa. Note: Part of container logs but not in Kibana.	prod	String
processId	Process ID internally assigned. Note: Part of container logs but not in Kibana.	"1"	String
ocLogId	End to End Log Identifier across the OCSEPP microservices. Helps to correlate the logs across the microservices in OCSEPP application. Note: Part of container logs but not in Kibana.	\${ctx:ocLogId}	String
vendor	Vendor Name	"oracle"	String

Common Useful log attributes

The following log attributes are available only through Kibana. These attribute names are part of Kubernetes Labels which are added in SEPPs each POD.

Table 2-10 Common useful log attributes

Log Attribute	Details	Sample Value	Data Type
engVersion	Engineering version	"23.3.0"	String
mktgVersion	Marketing version	"23.3.0.0.0"	String
vendor	Vendor Name	"Oracle"	String

3

Troubleshooting SEPP

This section provides information to troubleshoot the common errors which can be encountered during the installation and upgrade of SEPP:

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.

Caution

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

3.1 Generic Troubleshooting Scenarios

The following are the generic troubleshooting scenarios:

3.1.1 Generic Checklist

Environment Verification

The following sections provide generic checklist for troubleshooting tips:

a. Deployment related tips

Perform the following checks before the deployment:

- Are OCSEPP deployment, pods, and services created, running, and available? .

To check this, run the following command:

```
# kubectl -n get deployments,pods,svc
```

Inspect the output and check the following columns:

- AVAILABLE of deployment
- READY, STATUS, and RESTARTS of pod

- PORT(S) of service

b. Is the correct image used and the correct environment variables set in the deployment?

To check this, run the following the command:

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

c. Inspect the output, check the environment and image.

```
# kubectl -n seppsvc get deployment sepp-release-1-n32-egress-gateway -o yaml
```

d. Check if the microservices can access each other through REST interface.

To check this, run following command:

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

3.1.2 The environment is not working as expected

Problem:

The environment is not working as expected.

Solution:

1. Check if kubectl is installed and working as expected.
2. Check if kubectl version command works: This must display the versions of client and server.
3. Check if \$ kubectl create namespace test command works.
4. Check if kubectl delete namespace test command works.
5. Check if Helm is installed and working as expected.
6. Check if helm version command works: This must display the versions of client and server.

3.1.3 Kubernetes Node Failure

Problem

Kubernetes nodes goes down.

Error Code/Error Message

"NotReady" status is displayed against the Kubernetes node.

Symptom

On running the command `kubectl get nodes`, "NotReady" status is displayed.

Solution

Following is the procedure to identify the kubernetes nodes failure:

1. Run the following command to describe the node:

```
kubectl describe node <kubernete_node_name>
```

2. Check the nodes utilization by running the following command:

```
kubectl top nodes
```

3.1.4 SEPP Installation Verification

Problem: The SEPP installation is not successful.

Solution:

1. Verify if SEPP specific pods are working as expected by running the following command:

```
kubectl get pods -o wide -n <ocsepp _namespace>
```

Check whether all the pods are up and running.

Sample output:

NAME	TYPE	CLUSTER-IP	EXTERNAL-IP	PORT(S)	AGE	
ocsepp-release-appinfo-55b8d4f687-wqtgj						1/1
Running	0	141m				
ocsepp-release-cn32c-svc-64cd9c555c-ftd8z						1/1
Running	0	113m				
ocsepp-release-cn32f-svc-dd886fbcc-xr2z8						1/1
Running	0	4m4s				
ocsepp-release-config-mgr-svc-6c8ddf4c4f-lb4zj						1/1
Running	0	141m				
ocsepp-release-n32-egress-gateway-5b575bbf5f-z5bbx						2/2
Running	0	131m				
ocsepp-release-n32-ingress-gateway-76874c967b-btp46						2/2
Running	0	131m				
ocsepp-release-ocpm-config-65978858dc-t4t5k						1/1
Running	0	141m				
ocsepp-release-performance-67d76d9d58-llwmt						1/1
Running	0	141m				
ocsepp-release-plmn-egress-gateway-6dc4759cc7-wn6r8						2/2
Running	0	31m				
ocsepp-release-plmn-ingress-gateway-56c9b45658-hfcxx						2/2
Running	0	131m				
ocsepp-release-pn32c-svc-57774fdc4-2qpvx						1/1
Running	0	141m				
ocsepp-release-pn32f-svc-586cd87c7b-pxk6m						1/1
Running	0	3m47s				
ocsepp-release-sepp-nrf-client-nfdiscovery-65747884cd-qblqn						1/1
Running	0	141m				
ocsepp-release-sepp-nrf-client-nfmanagement-5dd6ff98d6-cr7s7						1/1
Running	0	141m				
ocsepp-release-nf-mediation-74bd4dc799-d9ks2						1/1
Running	0	141m				

2. If status of any pod is shown as ImagePullBackOff or ErrImagePull, then it can be due to:
 - a. Incorrect ImageName provided in ocsepp_custom_values_<versions>.yaml.

Then, double check the image name and tags in `ocsepp_custom_values_<versions>.yaml`.

- b. Docker registry is incorrectly configured.
Then, check docker registry is properly configured in all master and worker nodes.
3. If RESTARTS count of the pods is continuously increasing, then it can happen due to the following reasons:
 - a. MySQL primary and secondary hosts may not be configured properly in `ocsepp_custom_values_<versions>.yaml`
 - b. MySQL servers may not be configured properly. For more information about the MySQL configuration, see the *SEPP Predeployment Configuration* section in *Cloud Native Core. Security Edge Protection Proxy Installation, Upgrade ,and Fault Recovery Guide*.

3.1.5 Debugging General CNE

Problem: The environment is not working as expected

Solution:

Run the command `kubectl get events -n <ocsepp_namespace>` to get all the events related to a particular namespace.

3.1.6 Collecting the SEPP Logs to Check the Error Scenarios

Problem: The error scenarios are checked by collecting the SEPP logs.

Solution:

Run the following commands to get the logs from SEPP specific pods:

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

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

2. Run the following command to collect the logs from the specific pods or containers:

```
kubectl get pods -n <ocsepp_namespace>
```

3. Collect the logs from the pod and redirect to file by running `kubectl logs <pod_name> -n <ocsepp_namespace> > <Log File>`

Example:

```
kubectl logs - seppsvc-cn32f-svc-57cff5665c-skk41 -n seppsvc > seppsvc_logs1.log
```

3.1.7 Helm Error During the Rollback

Problem

The Helm rollback causes failure and displays the following error:

"Duplicate value: "cnc-metrics" && cannot patch"

Symptom

The error indicates that the Helm is not able to merge the current and rollback charts.

Solution

Run the helm rollback command again with `--force` to resolve the issue.

Note

If the rollback is to be performed using `--force`, take the backup of configmap data as the config map data can be cleaned by Helm.

3.1.8 Upgrade or Rollback Failure

When Security Edge Protection Proxy (SEPP) upgrade or rollback fails, perform the following procedure:

- Check the pre or post-upgrade or rollback hook logs in Kibana as applicable.
- Users can filter upgrade or rollback logs using the pod name filter

example: `ocsepp-release-update-db`

- Check the pod logs in Kibana to analyze the cause of failure.

After detecting the cause of failure, do the following:

For upgrade failure:

- If the cause of upgrade failure is database or network connectivity issue, contact your system administrator. When the issue is resolved, rerun the upgrade command.
- If the upgrade failure occurs during the preupgrade phase, resolve the issue, then perform a upgrade. Do not perform rollback because SEPP deployment remains in the source or older release.
- If the upgrade failure occurs during the postupgrade phase, for example, post upgrade hook failure due to target release pod not moving to ready state, then perform a rollback.

For rollback failure:

- If the cause of rollback failure is database or network connectivity issue, contact your system administrator. When the issue is resolved, rerun the rollback command.

If the issue persists, contact [My Oracle Support](#).

3.1.9 Helm Test Failure

Following are the troubleshooting steps if helm test is not getting initiated:

1. Run the following command to get the cn32f-svc name:

```
kubectl get svc -n namespace
```

Example:

```
kubectl get svc -n seppsvc
```

2. Add the following property under the global section in the custom-values.yaml file.

```
.global.seppServiceAccountName = cn32f-svc name
```
3. Upgrade to the same app version using updated custom-values.yaml file for changes to be updated in the installation.
4. Run helm test.

3.1.10 Helm Rollback Failure with the Configmap with the Name not Found Error

Problem

Helm rollback results in failure and displays the following error:

Rollback "ocsepp" failed: no ConfigMap with the name "rss-ratelimit-map" found

Symptom

The above-mentioned error indicates that Helm is not able to merge current and rollback charts.

Solution

Run the Helm rollback again with **--force** to resolve the issue.

Note

If the rollback is to be performed using **--force**, take the backup of configmap data as the config map data can be cleaned by Helm.

3.1.11 Continuous Restart of coherence-svc Pods

Problem: Helm install might fail if the coherence-svc pod is restarting repeatedly.

When you run `kubectl get pods -n <ocsepp_namespace>`, the coherence-svc pods restart count increases continuously.

Solution: Delete the coherence-svc pod using `kubectl delete po -n <namespace> <coherence pod>` and the pod will be up and running.

3.1.12 IllegalReferenceCount Exception Occurrence in Logs of Ingress and Egress Gateways

Problem: In some environments, there can be `IllegalReferenceCount` exception in the Ingress or Egress logs which results in an unexpected traffic drop. This is visible if the incoming traffic at Gateway is not equal to the outgoing traffic.

Solution:

1. Check the per second occurrence of `IllegalReferenceCount` exception on the Gateway pods.

2. Update the following configurations in the Config Map of the affected Gateway:

```
nettyInboundExceptions:
  exceptions:
    - io.netty.util.IllegalReferenceCountException
  count: 1000 //Update this to a value less than the per second occurrence
  timePeriod: 1
```

This resets the HTTP2 connection when the count reaches the configured value for this exception within the given time period.

3.1.13 False Message while Doing the Helm Uninstall

Problem: The user gets the following false message while doing the Helm uninstall:

```
These resources were kept due to the resource policy:
[ConfigMap] egress-ratelimit-map
[ConfigMap] rss-ratelimit-map
release "ocsepp-release" uninstalled.
```

Solution:

Run the following command to reverify whether the SEPP uninstallation is successful and the config maps are deleted:

```
kubectl get cm -n <namespace>
```

Output:

```
[seppuser@thrust6-bastion-1 ~]$ kubectl get cm -n <namespace>
NAME                DATA  AGE
istio-ca-root-cert  1      87m
kube-root-ca.crt    1      87m
[seppuser@thrust6-bastion-1 ~]$
```

Note

The listed output should not have 'egress-ratelimit-map' and 'rss-ratelimit-map'.

3.2 Feature Specific Troubleshooting Scenarios

The following are the feature specific troubleshooting scenarios:

3.2.1 Cat-2 Network ID Validation Feature

The following are the troubleshooting scenarios of Cat-2 Network ID Validation feature:

The incoming request is rejected at CN32F microservice**Problem:**

Incoming request gets rejected with error code configured (406 - default error code) at CN32F microservice.

Solution:

1. Search for error code SEPP-CN32FSEPP-ERROR-0013 or SEPP-CN32FSEPP-ERROR-0014 in CN32F logs.
2. Verify if correct regex is configured under **Header** or **Body IE** tab under Cat 2 – Network ID Validation Section of **Security Countermeasure** tab under **SEPP**.
3. Verify whether PLMN ID sent in the headers is part of PLMN ID List based on the associated SEPP configured.

The incoming request is rejected at PN32F microservice**Problem:**

Incoming request gets rejected with error code configured (406 - default error code) at PN32F microservice.

Solution:

1. Search for error code SEPP-PN32FSEPP-ERROR-0016 or SEPP-PN32FSEPP-ERROR-0017 in PN32F logs.
2. Verify if correct regex is configured under **Header** or **Body IE** tab under Cat 2 – Network ID Validation Section of **Security Countermeasure** tab under **SEPP**.
3. Verify whether PLMN ID sent in the headers is present in the PLMN ID List based on the associated SEPP configured.

Invalid PLMN ID in Header configurations**Problem:**

Invalid PLMN ID in Header configurations is received on SEPP.

Solution:

1. Verify if correct regex is configured against the header identifier in Header Configuration.
2. If error is thrown from CN32F microservices, verify if MCC and MNC combination is present in the PLMN ID List based on the associated SEPP configurations.
3. If error is thrown from PN32F microservices, verify proper configurations PLMN ID list based on the associated SEPP configurations.

Invalid PLMN ID in body configurations**Problem:**

Invalid PLMN ID in body configurations is received on SEPP.

Solution:

1. Verify if correct regex is configured against the body IE in Body IE Configuration.
2. If the error is from CN32F microservices, verify if MCC and MNC combination is present in the PLMN ID List based on associated SEPP configured.

3. If the error is from PN32F microservices, verify if MCC and MNC combination is present in the PLMN ID list based on associated SEPP configured.

3.2.2 Cat-1 Service API Validation Feature

The incoming request is rejected at CN32F:

Problem: The incoming request is rejected with the configured status code (default status code is 406) at CN32F microservice.

Solution:

1. Search for error code SEPP-CN32FSEPP-ERROR-0012 in CN32F logs.
2. Verify whether the proper HTTP method and Resource URI combination is sent in the request in CN32F logs.
3. Check allowed list name configured against Remote SEPP Set.
4. Verify whether the correct Resource URI and HTTP method is configured in the CNC Console GUI. Go to the **Security Countermeasure** section, check under the **Service API Allowed List** for that particular Allowed list name for N32 Egress or N32 Ingress Direction.
5. If the user is configuring a new Resource URI, ensure to configure the correct regular expression.

3.2.3 Overload Control Feature

Problem:

Incoming request does not get rejected with error code configured in CNC Console (429 - default error code) at N32 Ingress Gateway.

Solution:

1. Check whether the feature is enabled using the API: `curl -XGET http://<config-server>:<port>/sepp/nf-common-component/v1/igw/n32/ocpolicymapping`
2. Check if the correct policy is applied using the API: `curl -XGET http://<config-server>:port/sepp/nf-common-component/v1/igw/n32/ocpolicymapping`
3. Check the svcName parameter to verify whether the release name is correct or not for pn32f-svc.

Problem:

Scenario 1:

Feature is configured and enabled using REST API, still request is not getting rejected with the configured error code.

Solution

1. Fetch the current load level for N32 Ingress Gateway using the following API:

```
curl 'http://<release-name>-n32-ingress-gateway:80/igw/load-level?
svcName=<relase-name>-pn32f-svc' --http2-prior-knowledge
```

2. If the above API has the output "Normal", check CPU and memory thresholds defined using the API:

```
curl -XGET http://<config-server>:port/sepp/nf-common-component/v1/perf-info/overloadLevelThreshold
```

3. Check the CPU and memory statistics from Grafana or Prometheus to check the current CPU and memory usage.
4. Use the cgroup_cpu_nanoseconds and cgroup_memory_bytes metrics for the service mapping.
5. Either of the metrics value should reach the "onsetvalue" for a particular threshold level (defined in step 2) to be applied and feature to run.

Scenario 2:

The API mentioned in step 1 for scenario 1 returns the following error:

```
{"type":null,"title":"Service
      Unavailable","status":503,"detail":"Load level
      for service ocsepp-release-pn32f-svc is not Configured at Ingress-
Gateway","instance":null,"cause":"Load level for service ocsepp-release-pn32f-
svc is not Configured at
      Ingress-Gateway","invalidParams":null}
```

Solution

1. Check the ocsepp_custom_values_<version>.yaml file.
2. In the Perf-info section, check the tagNamespace value. The value must be either "namespace" or "kubernetes_namespace" depending on the CNE version used.
3. Check the configMap.prometheus value. This should map to Prometheus IP and port or service IP path used to access Prometheus.
4. If any of the above have been incorrectly set, change and re-deploy SEPP.

Scenario 3:

The API mentioned in step 1 for scenario 1 returns "Connection refused" error.

Solution

1. Run the following command:

```
kubectl get svc -n <namespace> | grep n32-ingress-gateway
```

2. If the output does not have port 80 present in service, do the following:
 - a. In the ocsepp_custom_values_<version>.yaml, set the enableIncomingHttp to true in the N32 Ingress Gateway section.
 - b. Re-deploy SEPP or upgrade the N32-Ingress-gateway service.
 - c. Verify that the port 80 is enabled by running the step 1.

3.2.4 Troubleshooting Steps for Rate Limiting Feature

Problem

Request not getting rejected with configured code.

Solution

1. Check **rateLimiting.enabled**. This parameter must be set to True.
2. Check **globalIngressRateLimiting.enabled**. This parameter must be set to True.
3. For Egress rate limiting, check **egressRateLimiting.enabled** must be set to True.

Problem

Request not getting rejected with configured error code.

Solution

1. In Ingress Gateway check for errorCodeProfiles in `ocsepp_custom_values_<version>.yaml` file.
2. Check profile name: ERR_1200.
3. Change the error code from 503 to desired value.
4. Upgrade or re-deploy SEPP

3.2.5 Message Feed Feature

The following are the troubleshooting scenarios of Message Feed feature:

Problem:

Messages of same transaction are getting copied to different partitions.

Solution:

Verify the configurations on all four gateways. `keybasedKafkaProducer` parameter should be set to true on all 4 gateways (n32-egress-gateway, plmn-ingress-gateway, n32-egress-gateway, and plmn-egress-gateway).

Problem:

The feature is not working in SASL_SSL or SSL mode.

Solution:

Verify the Data Director configurations and secrets in the *Oracle Communications Network Analytics Data Director Installation, Upgrade, and Fault Recovery Guide*.

Problem

Message Copy feature not copying JSON data to Data Director (DD).

Solution

1. Check whether the feature is enabled or not.
2. Check if **copyPayload** is set to false. If yes, set to **True**.
3. After re-deploying OCSEPP, verify if the data is copied at DD.

Problem

Message copy not copying data to DD as incorrect IP Port.

Solution

1. Check whether the feature is enabled or not.

2. Check whether the security enabled or not.
3. If security set to false, check whether DD Unreachable<GW> has been raised.
4. If yes , then **Kafka.bootstrapAddress** parameter must be set to correct listener IP and port
5. After re-deploying OCSEPP, verify whether data is copied at DD.

Problem

Message copy not copying data to DD as topic name incorrect.

Solution

1. Check whether the feature is enabled or not.
2. Check whether the security enabled or not.
3. If security set to false, check whether the DDUnreachable<GW> has been raised.
4. If not, check the **topicName** parameter. This topic should be created in DD so that data copied can be seen on DD.
5. After creating topic, verify if data is copied at DD.

Problem

Message copy not copying data to DD (security enabled) (Case A).

Solution

1. Check whether the feature is enabled or not.
2. Check whether the security enabled or not.
3. If security set to true ,check whether the DDUnreachable<GW> has been raised
4. If yes, then **Kafka.bootstrapAddress** parameter must be set to correct listener IP and security port for DD.
5. After re-deploying OCSEPP, verify if data is copied at DD.

Problem

Message copy not copying data to DD (security enabled) (Case B).

Solution

1. Check whether the feature is enabled or not.
2. Check whether the security enabled or not.
3. If security set to true, check whether the DDUnreachable<GW> has been raised
4. If not , then check the security configurations for DD.
5. Check the following parameters:
 - a. **userName**: must be the same as used to configure DD.
 - b. **password**: Check the secret name and Namespace details if correct.
6. After re-deploying OCSEPP, verify if data is copied at DD.

Note

All the values must be checked in `ocsepp_custom_values_<version>.yaml` file, as this is a Helm based feature.

3.2.6 Hosted SEPP

Problem:

The feature is Enabled and Consumer Remote SEPP Set not found (Default error code = 400).

Error Code or Error Message

Consumer Remote SEPP Set not found.

Solution:

1. Check whether the following error is displayed in logs. The error is displayed if allowed P-RSS Validation is enabled and no consumer RSS is configured.

```
{
  "instant":
  {"epochSecond":1668703429,"nanoOfSecond":698428472},"thread":"reactor-http-epoll-4","level":"ERROR","loggerName":"com.oracle.cgbu.cne.ocsepp.cn32f.handler.Cn32fSeppHandler","message":"HostedSEPPEXception: Request not allowed as source remote sepp set not found","contextMap":
  {"ocLogId":"1668703429683_71_ocsepp-release-mohit-plmn-ingress-gateway-7b86f4855c-ph9xj"},"endOfBatch":true,"loggerFqcn":"org.apache.logging.log4j.spi.AbstractLogger","threadId":15,"threadPriority":5,"instanceType":"prod","vendor":"oracle","ts":"22-11-17 16:43:49.698+0000","processId":"1","ocLogId":"1668703429683_71_ocsepp-release-mohit-plmn-ingress-gateway-7b86f4855c-ph9xj"}
```

2. Verify if consumer Remote SEPP Set is present on Hosted SEPP.

Problem:

The feature is enabled and destination Roaming Partner Set is null (Error code = 400) or Remote SEPP Set is not found (Error code = 404).

Error Code or Error Message

Destination RPS is null

Solution:

1. Above error is displayed if Allowed P-RSS Validation is enabled and no producer Remote SEPP Set is configured.
2. Verify if the producer Remote SEPP Set is present on Hosted SEPP.

Error Code or Error Message

destinationRPS not present

Problem:

The feature is enabled and destination Roaming Partner Set not present (Default error code = 400).

Solution:

1. Following logs is displayed and the following error is displayed if destination Sepp Set is not configured in **allowedProducerRemoteSeppSets** of Consumer SEPP Set:

```
{
  "instant": {
    "epochSecond": 1668705561,
    "nanoOfSecond": 940785763,
    "thread": "reactor-http-epoll-3",
    "level": "ERROR",
    "loggerName": "com.oracle.cgbu.cne.ocsepp.cn32f.handler.Cn32fSeppHandler",
    "message": "HostedSEPPEXception: Request not allowed as remote sepp set psepp not present in allowed list",
    "contextMap": {
      "ocLogId": "1668705561928_135_ocsepp-release-mohit-plmn-ingress-gateway-7b86f4855c-ph9xj"
    },
    "endOfBatch": true,
    "loggerFqcn": "org.apache.logging.log4j.spi.AbstractLogger",
    "threadId": 15,
    "threadPriority": 5,
    "instanceType": "prod",
    "vendor": "oracle",
    "ts": "22-11-17 17:19:21.940+0000",
    "processId": "1",
    "ocLogId": "1668705561928_135_ocsepp-release-mohit-plmn-ingress-gateway-7b86f4855c-ph9xj"
  }
}
```

2. Verify whether the destinationRPS is present in **allowedProducerRemoteSeppSets** configured At RSS of Consumer.
3. If it is present, then wait for cache refresh to take place as configured.

3.2.7 Steering of Roaming (SOR) Feature

The following are the troubleshooting scenarios of Steering of Roaming (SOR) feature:

SOR feature is not enabled.

Problem:

SOR feature is not enabled.

Solution:

Verify the following scenarios:

- Check whether the SOR feature is enabled at CNC Console or REST API.
- Check the Remote SEPP Set, validate SOR is enabled for the given RSS.
- Check Roaming Hub is disabled, and SEPP is deployed in SEPP Mode.

SOR feature is enabled at Global or RSS level but SOR is still disabled.

Problem:

SOR is enabled at Global or RSS level but SOR is still disabled.

Solution:

Verify the following scenarios:

- Check Remote SEPP Set Configuration, check the SOR list name associated with RSS.
- Verify the method plus URI that is passed in the message request exists in the SOR List.

SOR is configured with Retry as true and server header value is provided but retry is not working.

Problem:

SOR is configured with Retry as true and server header value is provided but retry is not working.

Solution:

- Verify that the server header value given at the time of configuration matches the value that reaches in server header in the response.
- Example: SOR server header value is configured as SOR-sorfqdn.com, and message request is sent, error response is received with server header value as 'server': 'SOR-sorfqdn.com'.

In this case, if retry is true then retry will be performed.

- If server header value is not matched, even when retry is true, retry will not be performed.

3.2.8 Rate Limiting for Ingress Roaming Signaling per Remote SEPP Set Feature

The following are the troubleshooting scenarios of Rate Limiting for Ingress Roaming Signaling per Remote SEPP Set feature:

Problem: Unable to see reject list for traffic sent

Solution:

1. Check in `ocsepp_custom_values_<version>.yaml` file if the `rssRateLimiter.enabled` parameter for N32 Ingress Gateway is set to true.
2. Check whether the feature is enabled on **Options** screen under **Remote SEPP Set**, which is under **Ingress Rate Limiting**, at CNC Console by checking whether Remote SEPP Set Ingress Rate Limiting Enabled parameter is set to true.
3. Check whether the feature is enabled on RSS level by checking RSS Ingress Rate Limiting Enabled parameter is set to true on **Remote SEPP Set** screen of CNC Console for the particular PLMN traffic.
4. Check whether the header configured in Originating Network ID Header parameter on **Options** screen under **Remote SEPP Set**, which is under **Ingress Rate Limiting**, at CNC Console, is being sent in traffic.

Problem: Unable to see status code in traffic set on Ingress Rate Limiting at CNC Console

Solution:

1. Check whether the status code configured is present in the RSS by checking Error configuration under **Ingress Rate Limiting** option in **Remote SEPP Set** screen for which the PLMN is being extracted.
2. Change the error code in RSS by editing **Error Configuration** under **Ingress Rate Limiting** option in **Remote SEPP Set** screen.

Problem: Unable to see error detail in traffic set on Ingress Rate Limiting at CNC Console

Solution:

1. Check whether the status code configured is present in the RSS by checking Error configuration under **Ingress Rate Limiting** option in **Remote SEPP Set** screen for which the PLMN is being extracted.

2. Change the error code in RSS by editing **Error Configuration** under **Ingress Rate Limiting** option in **Remote SEPP Set** screen.

Problem: Status code set to a different code in Error Configuration but Status code 429 is seen in rejected requests

Solution:

- Check if status code set on CNC Console is a valid HTTP Status code or in the series of 3xx. By default, these will be modified to 429.

Problem: Server header observed in logs

Solution:

- Server Header is added for the following Status codes - 408, 404, 400, and 429.

3.2.9 Cat-3 Previous Location Check feature

Problem:

Ingress request message gets rejected and displays the error code configured in the CNC Console (406 - default error code) at PN32F microservice.

Solution:

1. Search for the error codes SEPP-PREVIOUS-LOCATION-CHECK-VALIDATION-ERROR-0019 or SEPP-PN32FSEPP-ERROR-0018 or SEPP-PREVIOUS-LOCATION-CHECK-VALIDATION-EXCEPTION-0020 in PN32F microservice logs.
2. Verify if the correct regex is configured in Header or Body IE configuration for UE ID and Serving Network ID under Cat 3 – Previous Location Check Section under **Security Countermeasure** of SEPP CNC Console.
3. Verify if the MCC and MNC from serving network configured in either Header or Body is matching with the serving network name. The MCC and MNC values are part of the UDR response. Check whether the UDR response is success.
4. SUPI must be present in the incoming message, if it is configured for Cat-3 Previous Location Check.
5. UDR discovery procedure must be successful.
6. Coherence service must be up and running.
7. SUPI must be part of the IMSI range coming as part of the UDR profile received in UDR discovery response.
8. FQDN or IP of UDR must be reachable.
9. Proper DNS resolutions must be done for UDR discovery call, pn32f-svc for subscription use case.

3.2.10 Cat-3 Time check for Roaming Subscribers

The following are the troubleshooting scenarios of Cat-3 Time check for Roaming Subscribers feature:

Problem:

The Ingress Request message is rejected and shows the error code configured in the CNC Console (406 - default error code) at the PN32F microservice.

Solution:

1. Ensure that SUPI or SUCI is included in the incoming message.
2. Confirm that the UDR discovery procedure is successful.
3. Confirm that the UDM discovery procedure is successful.
4. Ensure that the Coherence service is up and running.
5. Check that SUPI is within the IMSI range provided in the UDR profile from the UDR discovery response.
6. Verify that the FQDN or IP of the UDR is reachable.
7. Make sure proper DNS resolution is done for the UDR discovery call, as well as for the pn32f-svc in the subscription use case.
8. Look for the error codes SEPP-UNAUTHENTICATED-LOCATION-TIME-CHECK-VALIDATION-ERROR-0021 or SEPP-UNAUTHENTICATED-LOCATION-TIME-CHECK-VALIDATION-EXCEPTION-0022 in the PN32F microservice logs.
9. Check if Cat-3 Time Check for Roaming Subscribers is enabled at both global and remote levels. If it's disabled at either level, the feature won't work.
10. Ensure that supiOrSuci and servingNetworkName (containing MCC and MNC) are present in the /nausf-auth/v1/ue-authentications Request JSON body.
11. Verify that the UDR response is successful, and check if the servingNetworkName in the /nausf-auth/v1/ue-authentications Request JSON body differs from the servingNetworkName in the UDR response.

Detailed Debugging Steps of the Feature

Following are the detailed Debugging steps of the feature:

Feature Configurations:

Verify the configurations on CNC Console for Cat-3 Time check for Roaming Subscribers feature.

Perform the following procedure to verify the **Cat-3 Time check for Roaming Subscribers** feature configurations:

1. From the left navigation menu, navigate to **SEPP** and then click **Security Countermeasure**.
2. Click **Cat-3 Time Location Check** under Security Counter Measure, **Unauthenticated Location** page appears underneath.
3. Click **Unauthenticated Location** under Security Countermeasure. The **Option** appears underneath.
4. Click **Option**, the option screen appears at the right pane. The Cat-3 Time check for Roaming Subscribers feature details are available on the screen.
5. Click **Edit** icon to modify the Option. The **Edit Option** page appears.
6. Set the **Cat 3 Time Check Unauthenticated Location Enabled** to True.
7. To access the **Remote SEPP Set** screen, click **SEPP** and then click **Remote SEPP Set**.
8. To enable the feature, the user also needs to enable the **Cat3-Time Location Check - Unauthenticated Location Enabled** parameter available at **Remote SEPP Set**.

Verify the configurations on REST API for Cat-3 Time check for Roaming Subscribers feature.

1. Using `/sepp-configuration/v1/security-counter-measure/time-location-check/unauthenticated-location` API, configure `timeUnAuthenticatedCheckValidationEnabled` to true along with other mandatory parameters to enable the feature at global level.
2. Using `/sepp-configuration/v1/remoteseppset` API, configure `messageFilterOnTimeUnAuthCheckEnabled` to true along with other mandatory parameters to enable the feature at RSS level.

Check Metrics for Error or Exception Scenario Analysis:

1. Evaluate the metric `ocsepp_time_unauthenticated_location_validation_requests_total` to check the total number of validation requests received.
2. Evaluate the metrics `ocsepp_time_unauthenticated_location_validation_failure_total` and `ocsepp_time_unauthenticated_location_exception_failure_total` to identify any failures or exceptions encountered during the validation process.
3. Evaluate the metrics `ocsepp_time_unauthenticated_location_blacklist_requests_total` to obtain information about requests that have been blacklisted.

Confirm Measurement Units:

If a request is being rejected, it might be due to a mismatch in the units of measurement used. Ensure to use the correct units when configuring the parameters.

1. From the left navigation menu, navigate to **SEPP** and then click **Security Countermeasure**.
2. Click **Cat-3 Time Location Check** under Security Counter Measure, **Unauthenticated Location** page appears underneath.
3. Click **Unauthenticated Location** under Security Countermeasure. The **Option** appears underneath.
4. Click **Option**, the option screen appears at the right pane. The Cat-3 Time check for Roaming Subscribers feature details are available on the screen.
5. Click **Edit** icon to modify the Option. The **Edit Option** page appears.
6. Ensure to set the **Blocklist Refresh Timer Value** and **Blocklist Refresh Time Unit** to the correct value.
7. Ensure to set the **Cache Refresh Timer (milliseconds)** to the milliseconds.

Check UDM Availability:

1. Verify that the FQDN or IP address of the UDR is reachable from the network. This can be checked through configuration details if verbosity is disabled.
2. Additionally, check the `ocsepp_time_unauthenticated_location_exception_failure` exception metric.

Validate Velocity Configuration:

1. If a request is being rejected, it may be due to a mismatch in the travel time calculated by the Cat-3 Time check for Roaming Subscribers feature. This can happen if the velocity used for the calculation is not aligned with the expected measurement unit. Ensure that the configured velocity aligns with the unit of measurement for Average Flight Velocity, which should be set in kilometers per hour (km/h). This ensures that the calculations are consistent and that the request passes the validation.
2. To check velocity related configuration using CNC Console:

- a. From the left navigation menu, navigate to **SEPP** and then click **Security Countermeasure**.
 - b. Click **Cat-3 Time Location Check** under Security Counter Measure, **Unauthenticated Location** page appears underneath.
 - c. Click **Unauthenticated Location** under Security Countermeasure. The **Option** appears underneath.
 - d. Click **Option**, the option screen appears at the right pane. The Cat-3 Time check for Roaming Subscribers feature details are available on the screen.
 - e. Click **Edit** icon to modify the Option. The **Edit Option** page appears.
 - f. Ensure that the **Average Flight Velocity (km/hr)** is set in kilometers per hour (km/hr).
3. To check velocity related configurations using REST API, in the `/sepp-configuration/v1/security-counter-measure/time-location-check/unauthenticated-location` REST API, configure the parameter `avgFlightVelocity` in kilometers per hour (km/hr).

Ensure UDM Availability:

1. Check that the UDM is present and reachable in the network. The UDM discovery procedure must be successful for the feature to function correctly. This can be checked through configuration details if verbosity is disabled.
2. Additionally, check the `ocsepp_time_unauthenticated_location_exception_failure` exception metric.

Check SUPI and SUCI in Incoming Requests:

1. Ensure that the incoming request includes a valid SUPI or SUCI.
2. If the incoming request contains a SUCI, the message will be forwarded to the UDM, which must be correctly discovered through NRF.
3. This can be checked using problem details if verbose is disabled or through logs.
4. Additionally, check the `ocsepp_time_unauthenticated_location_exception_failure` exception metric.

Validate Country Specific Parameters:

1. Confirm that the country related parameters such as longitude, latitude, and MCC are correctly configured to ensure accurate location validation.

3.2.11 Cat-0 SBI Message Schema Validation Feature

The following are the troubleshooting scenarios for Cat-0 SBI Message Schema Validation feature:

Problem:

The incoming request gets rejected at CN32F and PN32F microservices.

Solution:

1. Check the logs or metrics (`ocsepp_message_validation_on_body_failure` and `ocsepp_message_validation_on_header_failure`) to find the request has failed for which resource URI and HTTP method, do the following:

- a. If there is a request body failure, the following logs can be find by searching the text "Message validation failed for request body for request" :

```
{
  "instant":
  {"epochSecond":1680084693,"nanoOfSecond":192915132},"thread":"reactor-
http-
epoll-1","level":"ERROR","loggerName":"com.oracle.cgbu.cne.ocsepp.cn32f.
handler.Cn32fSeppHandler","message":"OUT:
      Cn32fSeppHandler::Message validation failed for request body
for request:
      /nausf-auth/v1/ue-authentications for method:
POST","contextMap":{"ocLogId":"1680084693177_151_ocsepp-release-plmn-
ingress-
gateway-77c69f7bbc-2fxvg"},"endOfBatch":true,"loggerFqcn":"org.apache.lo
gging.log4j.spi.AbstractLogger","threadId":16,"threadPriority":5,"instan
ceType":"prod","vendor":"oracle","ts":"23-03-29
10:11:33.192+0000","processId":"7","ocLogId":"1680084693177_151_ocsepp-
release-shafali-plmn-ingress-gateway-77c69f7bbc-2fxvg"}
```

- b. If there is a request query parameters failure, the following logs can be find by searching the text "Message validation failed for request query parameter(s) for request" :

```
{
  "instant":
  {"epochSecond":1678638067,"nanoOfSecond":537933800},"thread":"reactor-
http-
nio-4","level":"ERROR","loggerName":"com.oracle.cgbu.cne.ocsepp.cn32f.ha
ndler.Cn32fSeppHandler","message":"OUT:
      Cn32fSeppHandler:: Message validation failed for
      request query parameter(s) for request:
      //nnssf-nssselection/v2/network-slice-information for method:
GET","contextMap":
{"ocLogId":"1678638061928_34_"},"endOfBatch":true,"loggerFqcn":"org.apac
he.logging.log4j.spi.AbstractLogger","threadId":22,"threadPriority":5,"i
nstanceType":"prod","vendor":"oracle","ts":"23-03-12
21:51:07.537+0530","processId":"37136","ocLogId":"1678638061928_34_ocsep
p-release-plmn-ingress-gateway-77c69f7bbc-2fxvg"}
```

2. From the left navigation menu, navigate to **SEPP** and then click **Security Countermeasure**. Click **Cat 0 - SBI Message Schema Validation feature** under **Security Countermeasure**, the **Message Validation List** appears underneath. Do the following:
 - a. Search for the problematic resource URI and can get the corresponding schema.
 - b. Compare the request body or request query parameter value(s) against the corresponding schema and ensure that either the request is complaint with its schema or existing schema needs updation.
3. If the user wants to know the detailed causes of message validation failures user can generate the debug logs, search for the configured error code and title or text "Error in Request Body" or "Error in Request Parameter(s)" and can get the following logs:
 - a. Request body failure case log:

```
{
  "instant":
  {"epochSecond":1678392753,"nanoOfSecond":435438500},"thread":"reactor-
http-
nio-4","level":"DEBUG","loggerName":"com.oracle.cgbu.cne.ocsepp.webfluxl
```

```
og.LoggingResponseDecorator","message":"LoggingResponseDecorator::getBod
y() Response {"title":"Message validation
failed","status":406,"detail":"Message
validation for request /nausf-auth/v1/ue-authentications failed
for remote sepp set:
RS","instance":"/nausf-auth/v1/ue-
authentications","cause":"Error
in Request Body","invalidParams":
["requestBody.traceData.traceDepth:
should be valid to any of the schemas
string","requestBody.resynchronizationInfo.rand:
does not match the regex pattern ^[A-Fa-f0-9]{32}$
","requestBody.servingNetworkName:
does not match the regex pattern
^5G:mnc[0-9]{3}[.]mcc[0-9]{3}[.]3gppnetwork[.]org(:[A-F0-9]
{11})? $"","requestBody.traceData.eventList:
is missing but it is required","requestBody.supiOrSuci: is
missing but it is
required"}],"contextMap":
{"ocLogId":"1678392742101_34_"},"endOfBatch":true,"loggerFqcn":"org.apac
he.logging.log4j.spi.AbstractLogger","threadId":22,"threadPriority":5,"i
nstanceType":"prod","vendor":"oracle","ts":"23-03-10
01:42:33.435+0530","processId":"23320","ocLogId":"1678392742101_34_"}
```

b. Request query parameter failure case log:

```
{"instant":
{"epochSecond":1680685967,"nanoOfSecond":535845100},"thread":"reactor-
http-
nio-4","level":"DEBUG","loggerName":"com.oracle.cgbu.cne.ocsepp.webfluxl
og.LoggingResponseDecorator","message":"LoggingResponseDecorator::getBod
y() Response {"title":"Message validation
failed","status":406,"detail":"Message
validation for request /nudm-sdm/v2/imsi-9876540000000001
failed for remote sepp set:
RS","instance":"/nudm-sdm/v2/
imsi-9876540000000001","cause":"Error
in Request Parameter(s)","invalidParams":
["supported-features:
does not match the regex pattern ^[A-Fa-f0-9]*$
","parameters.dataset-names: is
missing but it is required"}],"contextMap":
{"ocLogId":"1680685963154_34_"},"endOfBatch":true,"loggerFqcn":"org.apac
he.logging.log4j.spi.AbstractLogger","threadId":22,"threadPriority":5,"i
nstanceType":"prod","vendor":"oracle","ts":"23-04-05
14:42:47.535+0530","processId":"13060","ocLogId":"1680685963154_34_"}
```

4. On the basis of failure reasons, the user can either correct the request body or request query parameter values or user can update the schema as mentioned in the step 2.

3.2.12 Configuration Failure in Remote SEPP and Remote SEPP Set

Problem:

Configuration operations (Add/ Delete/ Modify) failure in Remote SEPP and Remote SEPP Set, but user receives a 200 OK response code.

Solution:

- The user must check the value of metrics `ocsepp_configmgr_routefailure_total` before and after the configuration operations (Edit/Add/Delete).
- An increment in the counter indicates that the operation needs to be triggered again.

3.2.13 Aspen Service Mesh

Problem: SEPP Deployment fails in ASM mode.

Solution:

1. Check whether istio enabled flag set in namespace. If not, run the following command and deploy again:

```
kubectl label ns seppsvc istio-injection=enabled
```

2. Check PeerAuthentication is STRICT or PERMISSIVE . If it is set to STRICT, then change to PERMISSIVE and deploy again.
3. Run the following command to check the IP and host name in Service Entry for Kube-api-server:

```
kubectl get svc
```

Sample Output:

```
NAME TYPE CLUSTER-IP EXTERNAL-IP PORT(S)
kubernetes ClusterIP 10.96.0.1 <none> 443/TCP
```

4. Check whether the SEPP is able to connect to Database. If cnDBTier is deployed in another namespace, create the DestinationRule(DR) as given below and deploy again:

```
apiVersion: networking.istio.io/v1alpha3
kind: DestinationRule
metadata:
  name: ocsepp-db-service-dr
  namespace: <ocsepp-namespace>
spec:
  exportTo:
  - "."
  host: <db-service-fqdn>.<db-namespace>.svc.<domain>
  trafficPolicy:
    tls:
      mode: DISABLE
```


5. Check whether the Service Account has all the Roles and RoleBindings permissions to access all the resources. If not, give all permissions as given below:

```
...
verbs:
- '*'
```

Problem: nrf-client-nfmanagement and nrf-client-nfdiscovery pods are visible unhealthy on OSO.

Solution:

1. The nrfClientCommonServicePort parameter must be updated with the value 9091.
2. The port parameter must be updated with the value 9091 in the startupProbe, readinessProbe, and livenessProbe under nrf-client-nfmanagement section.

```
startupProbe:
  httpGet:
    path: /actuator/health
    port: 9091
  initialDelaySeconds: 60
  periodSeconds: 15
  timeoutSeconds: 10
  successThreshold: 1
  failureThreshold: 10
readinessProbe:
  httpGet:
    path: /actuator/health
    port: 9091
  initialDelaySeconds: 10
  periodSeconds: 10
  timeoutSeconds: 10
  successThreshold: 1
  failureThreshold: 10
livenessProbe:
  httpGet:
    path: /actuator/health
    port: 9091
  initialDelaySeconds: 15
  periodSeconds: 10
  timeoutSeconds: 10
  successThreshold: 1
  failureThreshold: 10
```

3. The port parameter must be updated with the value 9091 in the startupProbe, readinessProbe, and livenessProbe under nrf-client-nfdiscovery section.

```
startupProbe:
  httpGet:
    path: /actuator/health
    port: 9091
  initialDelaySeconds: 60
  periodSeconds: 15
  timeoutSeconds: 10
```

```

      successThreshold: 1
      failureThreshold: 10
    readinessProbe:
      httpGet:
        path: /actuator/health
        port: 9091
      initialDelaySeconds: 10
      periodSeconds: 10
      timeoutSeconds: 10
      successThreshold: 1
      failureThreshold: 10
    livenessProbe:
      httpGet:
        path: /actuator/health
        port: 9091
      initialDelaySeconds: 15
      periodSeconds: 10
      timeoutSeconds: 10
      successThreshold: 1
      failureThreshold: 10

```

4. In the `nrf-client-nfmanagement` and `nrf-client-nfdiscovery` section, add the value 9091 to the `istioExcludePorts` parameter.

```
istioExcludePorts: 53, 9091
```

3.2.14 Rate Limiting for Egress Roaming Signaling per PLMN feature

The following are the troubleshooting scenarios of Rate Limiting for Egress Roaming Signaling per PLMN feature:

Problem: Unable to see the discarded messages for the traffic sent.

Solution:

- Check `ocsepp_custom_values_<version>.yaml` file for the following:
 - In SEPP mode, check whether the `egressRateLimiter.enabled` parameter is set to true in the PLMN Ingress Gateway section of `ocsepp_custom_values_<version>.yaml` file.
 - In Roaming Hub mode, check whether the `egressRateLimiter.enabled` parameter is set to true in the N32 Ingress Gateway section of `ocsepp_custom_values_roaming_hub_<version>.yaml` file.

CNC Console

1. In the CNC Console GUI, from the left navigation menu, navigate to **SEPP** and click **Rate Limiting**.
2. Select **Egress Rate Limiting** which is defined under **Rate Limiting**.
3. The **Option** and **EgressRateLimitingList** appears underneath.
4. Click **Option**. The option screen appears at the right pane. Check whether **Egress Rate Limiting Enabled** is **true**.
5. Check whether request's **PLMN ID** is present in any of the **EgressRateLimitingList** and **Egress Rate Limiting Enabled** is set to **true**.

REST API

Check whether the `egressRateLimitingEnabled` is set to `True` using the REST APIs. For more details, see the Egress Rate Limiting Option Configuration and Egress Rate Limiting List Configuration REST APIs sections in the *Cloud Native Core, Security Edge Protection Proxy REST Specification Guide*.

Problem: Traffic is being forwarded even if tokens for the Egress Rate Limiting List are exhausted.

Solution:

- The `3gpp-Sbi-Message-Priority` header of the request must be verified before the message is dropped. If the priority in the header is less than (not equal to) `Discard Message Priority` property of the message in the Egress Rate Limiting List, then the message is not dropped.
- If the `3gpp-Sbi-Message-Priority` header is not present, then the priority is checked in the route configuration. If a value for `3gpp-Sbi-Message-Priority` is present in the route configuration, then the above mentioned condition is considered and the same solution is applied.
- If the priority is unknown for the request, 24 is considered as the default value for the request priority, then, the same condition as above is applied.

Problem: Status code is set to a different code in Error Configuration, but status code 429 is seen in rejected requests.

Solution: Check if status code set on CNC Console is a valid HTTP Status code or in the series of 3xx. By default, it should be 429.

Problem: The server header observed in response or logs.

Solution: The server header is observed in the response or logs, if the user configured error code is present in the Helm custom values. By default the status codes 400,404,408, and 429 are configured in Helm custom values.

3.2.15 Separate Port Configurations for N32c and N32f on the Egress Routes

The following are the troubleshooting scenarios for separate port configurations for `n32c` and `n32f` on the Egress routes feature:

Scenario: The Remote SEPP is changed with new fields of N32F configuration, and the traffic is not proper after changing the profile.

Solution:

1. Check whether the Remote Partner Set is created. If not, create the Remote Partner Set. For more information about API path, see *Oracle Communications Cloud Native Core, Security Edge Protection Proxy REST Specification Guide*.
2. Check if the configuration is stored inside the database correctly. Get the Remote SEPP profile using the following command and verify the configuration:

```
curl -X GET 'http://<ocsepp-release-name>-config-mgr-svc:9091/sepp-configuration/v1/remotesepp/<name>'
```

- a. Check if **N32fFQDN** is correct; whether it is mapped to correct DNS entry and is reachable.

- b. Check the **N32fAddress**, whether it is correctly mapped to the service. To confirm that the IP is mapped to correct service, run the following command:
- ```
kubectl get endpoints -n <namespace>
```
- c. Check whether the IP is mapped correctly to the intended service.
  - d. Check if **N32fPort** is configured correctly.
3. Verify in the database that the routes at n32-egress-gateway are updated according to the new configuration done at Remote SEPP. Run the following command to get the routes created:

```
curl -X GET 'http://<ocsepp-release-name>-config-mgr-svc:9090/sepp/nf-common-component/v1/egw/n32/peerconfiguration'#Sample
output[{"id":"psepp1","apiPrefix":"","Host":"<n32f-fqdn/IP>", "port":
"8888"}]
```

- a. Check in the above output if the Host and port parameter are the N32f IP and FQDN and port respectively.
- b. Run the following command to check that the ID is mapped correctly in the peer set:

```
curl -X GET 'http://<ocsepp-release-name>-config-mgr-svc:9090/sepp/nf-common-component/v1/egw/n32/peerconfiguration'#Sample
output[{"id":"RSS-2","httpConfiguration":
[{"priority":1,"peerIdentifier":"psepp1"}]]]
```

- c. Run the following command to check in routes configuration that the peerset Id is mapped correctly in the **peerSetIdentifier** parameter:

```
curl -X GET 'http://<ocsepp-release-name>-config-mgr-svc:9090/sepp/nf-common-component/v1/egw/n32/routesconfiguration'
```

## 3.2.16 Alternate Routing based on the DNS SRV Record for Home Network Functions

The following are the troubleshooting scenarios of alternate routing based on the DNS SRV Record for home network functions feature:

**Problem:** Virtual FQDNs are configured, but incoming request doesn't match any configured route.

**Solution:** Verify the routes and the matching criteria (URI and header) associated with each route. If the request is not matching any route, then the request will be routed via the configured default route.

**Problem:** The incoming requests are not routed according to the configuration defined at plmn-egress-gateway.

**Solution:**

- Get all the routes by using GET API.

Example:

```
curl -X 'GET' \
 'http://<config-mgr-svc-ip>:<port>/sepp/nf-common-component/v1/egw/plmn/
 routesconfiguration'
```

- Verify whether the Order id of the each route is configured correctly. Lower the order id, higher will be priority of routes.
- User must reconfigure the routes by using REST APIs.

**Problem:** The incoming requests are not routed to the target FQDNs associated with the virtual FQDNs in the DNS service.

**Solution:**

You can run below commands for debugging:

- Check if alternate-route-svc is up and running.
- Use "dig" command to verify if virtual FQDN is resolvable. Example dig -t srv "virtualFqdn". This command should return the list of the target FQDNs associated with the virtual FQDN.

**Problem:** Configurational issues at plmn-egress-gateway.

**Solution:** If the user faces difficulty while updating DNS SRV records, the configuration must be cleared in the following order:

1. Routes Configuration
2. Peerset Configuration
3. Peer Configuration

The order for the configurations must be as follows:

1. sbiroutingerrorcriteriasets
2. sbiroutingerroractionsets
3. Peer Configuration
4. Peerset Configuration
5. Routes Configuration

## 3.2.17 Load Sharing among Multiple Remote SEPP Nodes

The following are the troubleshooting scenarios of load sharing among multiple Remote SEPP nodes feature:

**Problem: The Remote SEPP is changed with virtualHost, and the traffic is not working properly after changing the profile**

**Solution:**

1. Check whether the Remote Partner Set is created. If not, create the Remote Partner Set. For more information about API path, see "Remote Partner Set" section of *Oracle Communications Cloud Native Core, Security Edge Protection Proxy REST Specification Guide*.

2. Check if the configuration is stored inside the database correctly. Get the Remote SEPP profile using the following command and verify the configuration:

```
curl -X GET 'http://<ocsepp-release-name>-config-mgr-svc:9091/sepp-configuration/v1/remotesepp/<name>'
```

- a. Check if the virtualHost is correct; whether it is mapped to correct DNS entry and is reachable.
- b. To verify that the virtual route is created at N32 egress gateway correctly, run the following command:

```
curl -X GET 'http://<ocsepp-release-name>-config-mgr-svc:9090/sepp/nf-common-component/v1/egw/n32/peerconfiguration'
#Sample
output[{"id":"psepp1","apiPrefix":"","virtualHost":"<virtualHost>"}]
```

- c. Check the above output, and whether the virtualHost is mapped to the virtualHost configuration in the Remote SEPP.
- d. Check if ID is configured correctly in the peer-set:

```
curl -X GET 'http://<ocsepp-release-name>-config-mgr-svc:9090/sepp/nf-common-component/v1/egw/n32/peerconfiguration'
#Sample output[{"id":"RSS-2","httpConfiguration":
[{"priority":1,"peerIdentifier":"psepp1"}]]]
```

3. Verify that the peerset Id in routes configuration is mapped correctly in peerSetIdentifier parameter:

```
curl -X GET 'http://<ocsepp-release-name>-config-mgr-svc:9090/sepp/nf-common-component/v1/egw/n32/routesconfiguration'
```

**Problem: Check whether target host is mapped against virtual host correctly.**

**Solution:**

Run the following curl from config mgr pod:

```
curl --no-proxy "*" --http2-prior-knowledge -X GET -H 'Accept: application/json' -H 'Content-Type: application/json' 'http://<ocsepp-release-name>-alternate-route:80/lookup?fqdn=<virtualhost>g&scheme=http'
```

**Sample output**

```
[{"target":"ocsepp-release-adiity-n32-ingress-gateway.sepp3.3gppnetwork.org","port":443,"ttl":60,"type":"SRV","dclass":"IN","priority":10,"weight":10000}, {"target":"ocsepp-release-adiity-n32-ingress-gateway.sepp2.3gppnetwork.org","port":443,"ttl":60,"type":"SRV","dclass":"IN","priority":10,"weight":10000}]bash-4.4$
```

## 3.2.18 5G SBI Message Mediation Support

For troubleshooting the mediation rules using Drools Rule Language (DRL) related scenarios, see "Error Messages for Mediation Rule Configuration" section in *Cloud Native Core, Service Communication Proxy Troubleshooting Guide*.

## 3.2.19 Support for TLS 1.3

The following are the troubleshooting scenarios of the feature:

**Problem:** Handshake is not established between SEPPs.

**Solution:**

- Check whether both the Remote Partner profiles posted are correct.
- Check the logs for N32 Ingress Gateway and N32 Egress gateway.
- If the error logs have the SSL exception, do the following:
  - Check the TLS version of both SEPPs, if both support different and single TLS versions, (that is, SEPP1 supports TLS 1.2 only and SEPP2 supports TLS 1.3 only or vice versa), handshake fails. Ensure that the TLS version is same for both SEPPs or revert to default config for both SEPPs.
  - The TLS version communication supported are:

**Table 3-1 TLS Version**

| Client TLS Version | Server TLS Version | Negotiated TLS Version |
|--------------------|--------------------|------------------------|
| TLS1.2+1.3         | TLS1.2+1.3         | TLSv1.3                |
| TLSv1.3            | TLSv1.3            | TLSv1.3                |
| TLSv1.3            | TLSv1.2+1.3        | TLSv1.3                |
| TLSv1.2+1.3        | TLSv1.3            | TLSv1.3                |
| TLSv1.2            | TLSv1.2+1.3        | TLSv1.2                |
| TLSv1.2+1.3        | TLSv1.2            | TLSv1.2                |

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

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

**Solution:**

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

If any of the above values is present, remove them or revert to default configuration for the pod to come up.

**Problem:** Pods not coming up after populating `clientSignatureSchemes` parameter.

**Solution:**

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

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

### 3.2.20 SEPP Deployment on OCI

The following is the troubleshooting scenario of SEPP Deployment on OCI:

**Title:** SEPP OCI metric dashboard does not display data.

**Problem:** The message "Query cannot result in more than 2000 streams" is displayed on the dashboard panel, instead of displaying the data.

**Solution:** The customer must add label filters to the query of the panels, which are giving the error.

Example of the default query:

```
oc_ingressgateway_http_requests_total[1m]{k8namespace="cnadb-test2"}.sum()
```

Example of the query with the label filter:

```
oc_ingressgateway_http_requests_total[1m]{app = "plmn-ingress-gateway",
k8namespace="cnadb-test2"}.sum()
```

Here, the label filter `app = "plmn-ingress-gateway"` is added to the query.

### 3.2.21 Georedundancy Support

The following are the troubleshooting scenarios for the Georedundancy Support feature:

**Problem: One of the cnDBTier site data is not reflected on other sites.**

**Solution:**

1. In the CNC Console GUI, from the left navigation menu, navigate to **SEPP** and then click **Georeplication Status**.
2. If the Replication Status is **Down**, then user need to perform Recovering a Failed Site procedure.  
For more information on how to perform Recovering a Failed Site procedure, see "Recovering a Failed Site" section in *Oracle Communications Cloud Native Core, Security Edge Protection Proxy User Guide*.

**Problem: Traffic is failing at C SEPP.**



**Solution:**

Verify the following on the Grafana dashboard:

- If the error is coming from alternate-route service, then check DNS configuration.
- If the error is coming on n32-egress-gateway service, then verify routing configuration in Remote SEPP Set.

**Problem: Traffic is failed on one of the producer SEPP instances even with equal weights and priorities.**

**Solution:**

1. Verify n32-ingress-gateway pod of P SEPP is up and running.
2. Verify n32-egress-gateway logs at C SEPP and n32-ingress-gateway logs at P SEPP to identity the reason for call drop.

**Problem: DNS SRV configuration is not reflecting.**

**Solution:**

1. Verify the DNS settings and run service restart.
2. Restart the pods on C SEPP:

```
<release>-alternate-route
<release>-n32-egress-gateway
```

**Problem: cnDBTier health APIs are not working.**

**Solution:**

Check the SEPP and cnDBTier compatibility in SEPP User Guide. If the health APIs are supported from cnDBTier 24.1.x onwards. For the earlier versions of cnDBTier, the health APIs were not supported.

For more details, see 'Support for cnDBTier APIs in CNC Console' section of *Oracle Communications Cloud Native Core, Security Edge Protection Proxy User Guide*.

**Problem: NDB health status Screen is blank.**

**Solution:** Enable the following parameters in cnDBTier yaml file and run the helm upgrade.

pvchealth:

```
enable:

 all: true

 mgm: true

 ndb: true

 api: true
```

## 3.2.22 Support for Originating Network Id Header Validation, Insertion, and Transposition

The following are the troubleshooting scenarios of the feature:

**Problem:** The incoming SBI request does not have any of the headers 3gpp-Sbi-Originating-Network-Id or 3gpp-Sbi-Asserted-Plmn-Id and the headers is not added by the feature (applicable only in SEPP mode).

**Solution:**

1. Check whether the feature is enabled or not global and remotely.
2. Check whether the metric `ocsepp_originating_id_header_added_total` is raised. The metric is for a successful header addition with useful information like the added header name, its value, the remote sepp, and the request URI.
3. Check whether the metrics `ocsepp_originating_header_addition_failed_total` (at CSEPP) and `ocsepp_originating_header_add_or_transpose_failed` (at PSEPP) are raised. The metrics are raised with the incoming request path, if the addition of missing header is failed due to any internal unforeseen error.
4. Check for the error log: "Error while adding missing originating network id header at consumer sepp" at C SEPP and log message: "Error while adding/transposing missing originating network id header at producer sepp" at P SEPP. The log is printed with the incoming request path, if the addition of missing header is failed due to any internal unforeseen error.

**Problem:** The header transposition is not working at P SEPP.

**Solution:**

1. Check whether the feature is enabled or not global and remotely.
2. Check whether the metric `ocsepp_originating_id_header_transposed_total` is raised. The metric is for a successful header addition with useful information like the added header name, its value, the remote sepp, and the request URI.
3. Check whether the metric `ocsepp_originating_header_add_or_transpose_failed_total` is raised. The metric is raised with the incoming request path at PSEPP in case header transposition is failed due to any internal unforeseen error.
4. Check for the error log: "Error while adding/transposing missing originating network id header at producer sepp" at P SEPP. The log is printed with the incoming request path, if the header transposition is failed due to any internal unforeseen error.

## 3.2.23 Proactive status updates on SEPP

The following are the troubleshooting scenarios of Proactive status updates on SEPP feature:

**Problem:** Peer marked as unhealthy or `oc_egressgateway_peer_health_status` is 1 even when peer is up or metric is not pegged.

**Solution:**

1. Ensure the Remote Peer is UP and running.

2. Ensure the flag `seppPeerHealthCheck` is set to "true" in `ocsepp_custom_values_<version>.yaml` file. If not, change it to true and run the helm upgrade.
3. Ensure in the Remote SEPP profile the **healthApiPath** and **healthApiMethod** parameters are present. If not, follow the steps to enable the feature on the CNC Console console.
4. Check the dimension `statusCode` of metric `oc_egressgateway_peer_health_ping_response_total`. If the code is expected code from peer and not present in `seppPeerHealthCheckCodes` custom values, then add in the custom values and run helm upgrade.
5. Ensure in the API Peer Configuration, the enable flag is set to "false". If not, change to false by running the REST API.

**Problem:** Peer is not sending configured response to health API requests.

**Solution:**

1. Ensure microservice is up and running
2. Ensure in the `ocsepp_custom_values_<version>.yaml` file the flag `healthCheckMonitoring` enabled is "true". If not, change to true and run Helm upgrade.
3. Check on the CNC Console to see if the feature is enabled.
4. Check the expected request Method and URI are same as configured on CNC Console GUI.
5. Check the configured response code on CNC Console.

### 3.2.24 Multiple SEPP instances on Shared cnDBTier Cluster

The following are the trouble scenarios of multiple SEPP instances on shared cnDBTier cluster feature:

**Problem:** After the cnDBTier upgrade, if the SEPP pods are stuck in an "Unready" state and the logs show a database connection failure.

**Solution:**

1. For cnDBTier release 23.4.0 and 24.2.0, the user should check the plugin using the following query:

```
SELECT user, host, plugin FROM mysql.user;
```

2. If the plugin value is "mysql\_native\_password," use an ALTER query to change the plugin from `mysql_native_password` to `caching_sha2_password`, and then proceed with the upgrade.

```
ALTER USER 'seppuser1'@'%' IDENTIFIED WITH caching_sha2_password BY 'NextGenCne1';
```

**Problem:** If a geo-replication failure occurs and disaster recovery is needed due to a fatal error, the unhealthy site will be reinstalled. Then the SEPP instances associated with the restored site are unable to connect to cnDBTier and are showing an access restriction message.

```
{"instant":
{"epochSecond":1724681253,"nanoOfSecond":107976541},"thread":"main","level":"E
```

```
RROR", "loggerName": "com.zaxxer.hikari.pool.HikariPool", "message": "HikariPool-1
- Exception during pool initialization.", "thrown": {"message": "Access denied
for user 'seppuser1'@'aclmx0466-ilom.us.oracle.com' (using password: YES)"}
```

**Solution:**

Create the necessary NF-specific user accounts and grants to match the NF users and grants of the working site in the reinstalled cnDBTier cluster, if those user accounts do not already exist. When disaster recovery is performed due to a fatal error, only the SEPP databases are replicated on the restored cluster.

**Note**

NF-specific user accounts and grants must be created manually because they are not replicated.

**Problem:** NF-specific roles are not visible in the CNC Console GUI.

**Solution:**

Ensure the flag `instanceLevelAuthorizationEnabled` is set to "true" in `cncc_custom_values_<version>.yaml` file and run the Helm upgrade.

**Problem:** The roles that are assigned roles are not applied to the user.

**Solution:**

Follow the correct combination of NF-specific roles as outlined in the *Oracle Communications Cloud Native Configuration Console User Guide*.

## 3.2.25 Cat-1 NRF Service API Query Parameters Validation Feature

The following are the troubleshooting scenarios of Cat-1 NRF service API query parameters validation feature:

**Problem:**

The Ingress Request message is rejected and shows the error code (406 - default error code) with the message "Query Param Validation failed."

**Solution:****1. Enable Required Features:**

- The Cat-1 Service API Validation feature must be enabled for the Cat-1 Service API Query Parameters Validation feature to function accurately.
- If the request fails, make sure both the Cat-1 Service API Validation feature and the Cat-1 Service API Query Parameters Validation feature are enabled.

**2. List Name Configuration:**

- Check that the list name set at the RSS level matches the list for query parameter validation.
- Ensure the same list name is selected across all lists configured for query parameter validation.

**3. NF Pair Validation:**

- Verify the pair of Network Functions (NFs) in the request. Make sure they match the configuration set of values for this feature.
4. Correct API Request Format:
    - The request must be a NRF discovery request in the following format:
 

```
nnrf-disc/v1/nf-instances?requester-nf-type={} & target-nf-type={}
```
    - The requester-nf-type={} and target-nf-type={} should have the configured values for NFs.
    - Use only the GET method with this API, as the feature supports this method only.
    - Ensure both requester-nf-type and target-nf-type are included in the request.

Here are the detailed debugging steps to follow:

#### Feature Configurations:

1. Verify CNC Console Configurations for Cat-1 Service API Query Parameters Validation feature:
  - In the CNC Console GUI, from the left navigation menu, navigate to **SEPP** and then click **Security Countermeasure**.
  - Click **Cat 1 -Service API Validation** under **Security Countermeasure. Option, Service API Allowed List** , and **Cat-1 Query Parameter Validation List** appears.
  - Click **Option**, the **Options** page appears on the right pane.
  - Ensure that Enable Cat 1-Query Parameter Validation parameter and Enable Cat 1 - Service API Validation are set to True to enable the feature.
  - Add any additional configurations in the **Cat 1 - Service API Query Param Validation List** page.
2. Verify Configurations via REST API for Cat-1 Service API Query Parameters Validation feature:
  - Use the following API to configure the "queryParamValidationEnabled" parameter to 'true':
 

```
/sepp-configuration/v1/security-counter-measure/feature
```
  - Use this API to configure other mandatory parameters for the feature:
 

```
/sepp-configuration/v1/security-counter-measure/service-api-query-param-validation-list
```

Check the following metrics and alerts for error scenario analysis:

1. Successful Requests:
 

To analyze all successful requests for the feature, check the metric:

```
ocsepp_security_service_api_query_param_validation_success_total
```
2. Failure Requests:
 

To analyze all failed requests for the feature, check the metric:

```
ocsepp_security_service_api_query_param_validation_failure_total
```
3. To analyze the error scenario, check the following alerts:
 

```
SEPPN32fServiceApiQueryParamValidationFailureAlertWarn
```

SEPPN32fServiceApiQueryParamValidationFailureAlertMinor

SEPPN32fServiceApiQueryParamValidationFailureAlertMajor

SEPPN32fServiceApiQueryParamValidationFailureAlertCritical

**Problem:**

Unable to save configurations for the Cat-1 Service API Query Parameters Validation successfully.

**Solution:****1. NF Configuration:**

- Ensure that one of the following eight Network Functions (NFs) is configured: NRF, UDM, AMF, SMF, AUSF, PCF, SEPP, and SCP.

**2. Correct NF Type Names and Values:**

- Verify that the **requester-nf-type** and **target-nf-type** are correctly named and assigned valid values.

**3. Validation List Configuration:**

- Confirm that the **CAT1 Service API Validation** screen already contains the list you intend to configure for the SCM CAT1 Query Parameter Validation feature.

**4. Mandatory Parameters:**

- Ensure that all required parameters are configured, including:
  - Name and value of **requester-nf-type** and **target-nf-type**
  - **Resource URI**
  - **Method**
  - **List name**

- The request must be an NRF discovery request in the following format:

/nnrf-disc/v1/nf-instances?requester-nf-type={}&target-nf-type={}

- Only the **GET** method should be used with this API, as this feature supports only the GET method.

**5. Avoid Redundant Entries:**

- Duplicate entries for the name and value of **requester-nf-type** and **target-nf-type** are not allowed in the configuration.

## 3.2.26 Integrating SEPP with 5G Network Intelligence Fabric (5GNIF) feature

The following are the troubleshooting scenarios of integrating SEPP with 5G Network Intelligence Fabric (5GNIF) feature:

**Problem:** 5GNIF Discovery Request Not Sent to NRF.

**Solution:****1. Verify feature enablement at the Helm level:**

- Check whether the 5GNIF feature is enabled in the Helm configuration:

- Open the `ocsepp_custom_values_<version>.yaml` file.
- Navigate to the `nif` section of `config-mgr-svc`.
- Confirm that the parameter `enableNif` is set to `true`.

You can also validate this directly from the deployment using:

```
kubectl describe deploy <release-name>-config-mgr-svc -n <namespace> |
grep ENABLE_NIF
```

- Ensure the output shows `ENABLE_NIF` is set to `true`.
- If not, update the value in the custom values yaml file and perform a Helm upgrade to apply the change.

2. Confirm feature enablement through REST API:  
Use the following curl command to check if 5GNIF is enabled in the runtime configuration:

```
curl 'http://<config-mgr-svc>:<port>/sepp-configuration/v1/nif/options' -X
GET
```

- The returned JSON should show `"enabled": true`.

If this is not the case, refer to the relevant configuration section for corrective action.

3. Check discovery delay settings:  
Make sure the discovery process is not being delayed due to configuration:

- Verify the values of `nifDiscoveryInitialDelay` and `nifDiscoveryScheduledDelay`.
- The first discovery request will only be triggered after the configured `nifDiscoveryInitialDelay`.

4. Verify HTTP enablement for NIF:  
Ensure that HTTP is enabled for NIF discovery:

- Run the following command:

```
kubectl describe deploy <release-name>-config-mgr-svc -n <namespace> |
grep HTTP_ENABLED_NIF
```

- Check that `httpEnabledNif` is set to `true` in the `ocsepp_custom_values_<version>.yaml` file.

If not, update the value and perform a Helm upgrade to apply the change.

**Problem:** No Peers Visible on PLMN Egress Gateway (CNCC Screen / REST API) Even After 5GNIF is Registered on NRF.

#### Solution:

If the PLMN Egress Gateway does not show any NIF peers, even though 5GNIF is successfully registered on the NRF, follow these steps to troubleshoot and resolve the issue:

1. Verify That the 5GNIF feature is enabled: Ensure the feature is enabled in both the Helm deployment and runtime configuration, as outlined in Scenario 1.
2. Check the 5GNIF Discovery Name Configuration:  
Confirm that the `nifDiscoveryName` in `config-mgr-svc` matches the `nfType` set in the `NFProfile`:

- Run the following command:

```
kubectl describe deploy <release-name>-config-mgr-svc -n <namespace> |
grep NIF_DISCOVERY_NAME
```

- If the discovery name differs from the nfType in the NFProfile, update the value in the ocsepp\_custom\_values\_<version>.yaml file:

```
config-mgr-svc:
 nif:
 nifDiscoveryName: <correct_nfType_value>
```

- Perform a Helm upgrade to apply the changes.

### 3. Validate PLMN Egress Gateway Configuration for header handling:

Ensure the PLMN Egress Gateway is correctly configured to accept requests where certain headers are absent:

- Check the configMap for the following parameter: sepp.headerAbsentPredicate:  
true
- If it's set to false or missing, update ocsepp\_custom\_values\_<version>.yaml under the plmn-egress-gateway section:

```
plmn-egress-gateway:
 sepp:
 headerAbsentPredicate: true
```

- Perform a Helm upgrade to apply the configuration.

### 4. Verify NRF Client Configuration Parameters.

In the nrfclient section of the configuration, verify the following parameters are set correctly:

- primaryNrfApiRoot
- enableVirtualNrfResolution
- virtualNrfFqdn
- virtualNrfScheme

Also, ensure DNS entries are properly configured and resolvable for the NRF FQDN.

### 5. Check the NRF Route Format in Egress Gateway:

Ensure the correct nrf\_route is created with the appropriate structure. It should look like the following:

```
[
 {
 "id": "nrf_route",
 "uri": "egress://request.uri",
 "order": 1,
 "filters": [
 {
 "args": null,
 "name": "DefaultRouteRetry"
 }
]
 }
],
```



```

 "predicates": [
 {
 "args": {
 "pattern": "/nnrf-*/**"
 },
 "name": "Path"
 },
 {
 "args": {
 "headerName": "oc-xfcc-dns"
 },
 "name": "HeaderAbsent"
 }
]
 }
]

```

- Ensure that the route ID, URI, filters, and predicates match exactly.
- Missing or misconfigured routes can prevent proper peer registration.

**Problem:** 500 Internal Server Error When Sending Traffic.

**Solution:**

1. Check routes Configuration:  
If the `nifReject` route is present, it means all NIF peers have been removed.
2. Verify Peer Configuration: If the peer list is empty, do the following:
  - Ensure all 5GNIF instances are in the REGISTERED state in the NRF. Re-register if needed.
  - If the feature was recently disabled and re-enabled, wait for the configured `nifDiscoveryInitialDelay`, then recheck the peer list.

**Problem:** SBI Request Returning 404 (N32F Context Not Found).

**Solution:**

1. • Run the following command:

```
kubectl describe deploy <release-name>-config-mgr-svc -n <namespace> |
grep SAN_HEADER_NAME
```

- Ensure `SAN_HEADER_NAME` is set to `oc-xfcc-dns`. If not, update the `ocsepp_custom_values_<version>.yaml`:

```

configs:
 sanHeaderName: "oc-xfcc-dns"

```

- Perform a Helm upgrade after making the change.
2. Check Header Predicate in PLMN Egress Gateway
    - In the `configMap`, verify:

- If it's missing or false, update the `ocsepp_custom_values_<version>.yaml` under `plmn-egress-gateway`:

```
plmn-egress-gateway:
 sepp:
 headerAbsentPredicate: true
```

- Perform a Helm upgrade.

### 3. Verify Feature Configuration (SoR):

- Ensure Steering of Roaming (SoR) is not enabled.
- If SoR was previously enabled, then go to CNCC > Configurations > Gateways > EGW > PLMN Egress Gateway > Routes Configuration, and Remove all SoR routes and Disable SoR.

**Problem:** Alternate Routing Issues.

**Solution:**

#### 1. Check Current Routing Criteria and Actions:

Run the following curl commands to inspect current error handling configurations:

```
curl http://127.0.0.1:9090/sepp/nf-common-component/v1/egw/plmn/
sbiroutingerrorcriteriasets
```

```
curl http://127.0.0.1:9090/sepp/nf-common-component/v1/egw/plmn/
sbiroutingerroractionsets
```

#### 2. Validate Server Header:

Ensure the server header in the response matches the expected pattern defined in `sbiroutingerrorcriteriasets` (Exmple: matches `.*NIF.*`).

#### 3. Check Response Codes

- If the response code is not one of 500, 503, or 504, you'll need to update the configuration to include the response code that caused the issue.
- In the `ocsepp_custom_values_<version>.yaml`, update the following under `nif.nifRoutingErrorCriteriaSets`:

```
[{
 "id": "nif_criteria_1",
 "method": ["GET", "POST", "PUT", "DELETE", "PATCH"],
 "response": {
 "statuses": [
 {"status": [500, 503, 504, 502], "statusSeries": "5xx"}
],
 "headersMatchingScript": "headerCheck,server,via,.*NIF.*"
 }
},
{
 "id": "nif_criteria_0",
 "method": ["GET", "POST", "PUT", "DELETE", "PATCH"],
 "exceptions": [
 "java.util.concurrent.TimeoutException",
 "java.net.SocketException",
 "java.net.SocketTimeoutException",
```

```

 "java.net.UnknownHostException",
 "java.net.ConnectException",
 "java.net.NoRouteToHostException"
]
}]

or

[
 {
 "id": "nif_criteria_1",
 "method": ["GET", "POST", "PUT", "DELETE", "PATCH"],
 "response": {
 "statuses": [
 {
 "status": [400, 404],
 "statusSeries": "4xx"
 },
 {
 "status": [500, 503, 504],
 "statusSeries": "5xx"
 }
],
 "headersMatchingScript": "headerCheck,server,via,.*NIF.*"
 }
 },
 {
 "id": "nif_criteria_0",
 "method": ["GET", "POST", "PUT", "DELETE", "PATCH"],
 "exceptions": [
 "java.util.concurrent.TimeoutException",
 "java.net.SocketException",
 "java.net.SocketTimeoutException",
 "java.net.UnknownHostException",
 "java.net.ConnectException",
 "java.net.NoRouteToHostException"
]
 }
]
}]

```

#### 4. Validate EnvoyFilter:

Check that the EnvoyFilter responsible for handling headers is correctly applied:

```
kubectl get envoyfilter -n <namespace>
```

```

NAME AGE
serverheaderfilter 22h

```

Run the following command:

```
kubectl describe envoyfilter serverheaderfilter -n <namespace>
```

Sample output:

```

kubectl describe envoyfilter serverheaderfilter -nsepp-1
Name: serverheaderfilter
Namespace: sepp-1
Labels: app.kubernetes.io/managed-by=Helm
Annotations: meta.helm.sh/release-name: servicemesh

```

```

meta.helm.sh/release-namespace: sepp-1
API Version: networking.istio.io/v1alpha3
Kind: EnvoyFilter
Metadata:
 Creation Timestamp: 2025-07-23T12:11:15Z
 Generation: 1
 Resource Version: 306427514
 UID: b4bb357d-a8ac-48e7-937d-629ba10ebfc5
Spec:
 Config Patches:
 Apply To: NETWORK_FILTER
 Match:
 Listener:
 Filter Chain:
 Filter:
 Name: envoy.filters.network.http_connection_manager
 Patch:
 Operation: MERGE
 Value:
 typed_config:
 @type: type.googleapis.com/
envoy.extensions.filters.network.http_connection_manager.v3.HttpConnectionMana
ger
 server_header_transformation: PASS_THROUGH
Workload Selector:
 Labels:
 app.kubernetes.io/part-of: ocsepp

```

### 3.2.27 LCI and OCI Header Support Feature

The following are the troubleshooting scenarios of LCI and OCI header support feature:

**Problem: Unable to see LCI header in the response when the feature is enabled**

**Solution:**

1. Confirm that the feature is enabled in the Helm configuration.
2. In the perf-info section, verify that the parameter configmapPerformance.prometheus is properly configured to match the Prometheus service deployed in the cluster. Ensure the perf-info service can report CPU load for the cn32f-svc and pn32f-svc microservices.
3. Ensure that any consumer NF Identity (OAuth token, User-Agent, or Via header) is included in the request.
4. Check the localLciHeaderValidity: if multiple requests occur within the validity period without breaching configured thresholds, the SEPP will not add LCI headers.

**Problem: Unable to see OCI header in the response when the feature is enabled**

**Solution:**

1. Confirm that the feature is enabled in the Helm configuration.
2. In the perf-info section, verify that the parameter configmapPerformance.prometheus is properly configured according to the Prometheus service deployed in the cluster. Confirm that the perf-info service reports CPU load for the cn32f-svc and pn32f-svc microservices.

3. Ensure that any consumer NF Identity (OAuth token, User-Agent, or Via header) is included in the request.
4. Verify that overloadConfigRange is correctly set in Helm and that the SEPP is in an overloaded state that triggers OCI header generation.
5. Check the validity period: if multiple requests happen within this period without breaching the configured thresholds, the SEPP will not add OCI headers.

## 3.3 HTTP Response Codes and Error Codes

The following are the HTTP Response Codes:

**Table 3-2 HTTP Response Codes and Error Codes**

| Data Type                               | Mandatory(M)/Optional(O)/Conditional(C) | Cardinality | Response Code               | Description                                                                                                  |
|-----------------------------------------|-----------------------------------------|-------------|-----------------------------|--------------------------------------------------------------------------------------------------------------|
| ProblemDetails                          | C                                       | 1           | 400 - BAD REQUEST           | SEPP config-mgr-svc shall send the response when the Request body received is not as per defined Data Model. |
| ProblemDetails                          | C                                       | 1           | 500 - INTERNAL SERVER ERROR | SEPP config-mgr-svc shall send the response when an internal error has occurred.                             |
| Response Body as per Data Model defined | C                                       | 1           | 200 - OK                    | SEPP config-mgr-svc shall send the response when the request is successful.                                  |
| ProblemDetails                          | C                                       | 1           | 404 -NOT FOUND              | SEPP config-mgr-svc shall send the response when the requested entry is not present in the database.         |

### Error Codes and Recovery Steps

The following tables list the various SEPP error codes and the recovery steps:

**Table 3-3 Error Codes and Recovery Steps**

| Error Code             | Error Text                   | Command/Method                                                               | Description                                                                | Recovery Steps                                                      |
|------------------------|------------------------------|------------------------------------------------------------------------------|----------------------------------------------------------------------------|---------------------------------------------------------------------|
| SEPP-COM-DB-ERROR-0002 | Remote Sepp record not found | GET remotesepp/{name}<br>DELETE remotesepp/{name}<br>PATCH remotesepp/{name} | This error is observed if the provided name in the request is not present. | Verify that Remote SEPP name given in request parameter is present. |

**Table 3-3 (Cont.) Error Codes and Recovery Steps**

| Error Code             | Error Text                                               | Command/ Method                                                                                     | Description                                                                                                                                                                                                | Recovery Steps                                                                                                                                                          |
|------------------------|----------------------------------------------------------|-----------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SEPP-COM-DB-ERROR-0003 | N32F Context not found record not found                  | GET handshakestatus /fqdn/{fqdn} GET handshakestatus /name/{name}                                   | This error is observed if the context with given name or fqdn in the request parameter is not found.                                                                                                       | Verify that context with given name or fqdn in request parameter is present in DB.                                                                                      |
| SEPP-COM-DB-ERROR-0005 | Database connection is down                              | Can be thrown from any Method (Generic Exception if application is unable to connect with Database) | This error is observed if the application is unable to make the connection with Database.                                                                                                                  | Verify that DB is up and running.                                                                                                                                       |
| SEPP-COM-DB-ERROR-0007 | SQL Grammer exception                                    | PUT remoteseppset/ {name}                                                                           | This error is observed if there is some corruption in the Database related to Remote SEPP Set table (This error gets generated for all the commands that is trying to access corrupted table or Database). | Recheck the corruption in database and re-install SEPP                                                                                                                  |
| SEPP-COM-DB-ERROR-0008 | Constraint violation exception of database table columns | PUT {remoteSepp}                                                                                    | This error is observed when one of the mandatory parameters required for remote SEPP are not present.                                                                                                      | Verify that mandatory parameter like name, seppfqdn are present.                                                                                                        |
| SEPP-COM-DB-ERROR-0009 | Unsupported security capability list exception           | POST / remotesepp PUT remotesepp/ {name} PATCH remotesepp/ {name}                                   | This error is observed if unsupported securityCapabilityList is provided in request.                                                                                                                       | Make sure securityCapabilityList provided is supported by SEPP. Allowed security capability list is 'TLS' and 'TLS and PRINS.                                           |
| SEPP-COM-DB-ERROR-0010 | Update not allowed on table entry exception              | PUT remotesepp/ {name} PATCH remotesepp/ {name}                                                     | This error is observed if given parameter to update is same as configured one or trying to update mandatory parameter.                                                                                     | Verify that the parameter provided in the request to update is different from the configured one or not while updating any mandatory parameter like name, seppfqdn etc. |
| SEPP-COM-DB-ERROR-0011 | Remote Sepp already present                              | POST / remotesepp                                                                                   | This error is observed if Remote SEPP with same seppfqdn is already present in DB.                                                                                                                         | Verify that seppfqdn provided in configuration request is not already present in DB.                                                                                    |

**Table 3-3 (Cont.) Error Codes and Recovery Steps**

| Error Code             | Error Text                                  | Command/Method                                                                                    | Description                                                                                                   | Recovery Steps                                                                                                                                                                                                                                                                          |
|------------------------|---------------------------------------------|---------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SEPP-COM-DB-ERROR-0012 | Mandatory Parameter Update Not Allowed      | PUT<br>remotesepp/<br>{name}<br>PATCH<br>remotesepp/<br>{name}                                    | This error is observed if the user is trying to update the value of mandatory parameter which is not allowed. | Verify that some of the mandatory parameters are not allowed to be updated. Those allowed should match with the value in the request.                                                                                                                                                   |
| SEPP-COM-ERROR-0013    | Invalid PLMN List in Request                | POST /<br>remotesepp<br>PUT /<br>remotesepp                                                       | This error is observed if the user has configured PLMN in incorrect format in PLMNID List.                    | Verify the entered PLMN (mcc and mnc) in PLMNID List while configuring Remote Sepp.                                                                                                                                                                                                     |
| SEPP-COM-DB-ERROR-0020 | Remote Sepp Set not found                   | GET<br>RemoteSeppSet/<br>{name}                                                                   | This error is observed if the Remote Sepp Set is not present                                                  | Give the Remote SEPP Set name that exist in the database                                                                                                                                                                                                                                |
| SEPP-COM-DB-ERROR-0021 | Remote Sepp Set associated with Remote SEPP | DELETE<br>RemoteSepp/<br>{name}<br>PUT<br>RemoteSepp/<br>{name}<br>PATCH<br>RemoteSepp/<br>{name} | This error is observed if Remote SEPP is associated with Remote Sepp Set, it gives this error on deletion.    | Disassociate the Remote Sepp from Remote SEPP Set by executing the Remote SEPP Set PUT command.<br><br>Or<br>Do the following: <ul style="list-style-type: none"> <li>• Delete the Remote Sepp Set</li> <li>• Delete the Remote Sepp</li> <li>• Create Remote Sepp Set again</li> </ul> |
| SEPP-COM-DB-ERROR-0022 | Remote Sepp Set already exists              | POST<br>RemoteSeppSet/<br>{name}                                                                  | This error is observed if Remote Sepp Set already exists and same entry is added again.                       | Use a unique name in Remote Sepp Set                                                                                                                                                                                                                                                    |

Table 3-3 (Cont.) Error Codes and Recovery Steps

| Error Code             | Error Text                          | Command/Method                                        | Description                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                    | Recovery Steps                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|------------------------|-------------------------------------|-------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SEPP-COM-DB-ERROR-0023 | Remote Sepp Set configuration error | PUT RemoteSeppSet/{name}<br>POST RemoteSeppSet/{name} | <ul style="list-style-type: none"> <li>Remote Sepp Set does not exists</li> <li>Requested Remote Sepp does not exists</li> <li>At least one Remote Sepp should be associated with Remote Sepp Set</li> <li>Configured Domains are different between peers</li> <li>Remote Sepp Set exists with same domain</li> <li>Configured PLMNs are different between peers</li> <li>Remote Sepp Set exists with same PLMN</li> <li>Configured PLMNs/Domains are different between peers</li> <li>Associating Remote Sepp should be unique for Remote Sepp Set</li> </ul> | <ul style="list-style-type: none"> <li>Give the Remote SEPP Set name that exists in the database</li> <li>Give the Remote SEPP name that exists in the database</li> <li>Give at least a single Remote Sepp name while creating a Remote Sepp Set</li> <li>Check that each Remote Sepp has same Domain when creating a Remote Sepp Set</li> <li>Check that each Remote Sepp has same PLMN when creating a Remote Sepp Set</li> <li>Check that each Remote Sepp name is unique while creating a Remote Sepp Set</li> </ul> |
| SEPP-COM-xx-ERROR-0101 | Config Not Acceptable               | PUT POST PATCH /v1/remotesep/                         | This error is observed if PLMNidList is empty or PLMNidList size is greater than max size allowed or if domain is null.                                                                                                                                                                                                                                                                                                                                                                                                                                        | Verify that PLMNidList is not empty or PLMNidList size is not greater than max size allowed or if domain is not null.                                                                                                                                                                                                                                                                                                                                                                                                     |
| SEPP-COM-xx-ERROR-0102 | Mandatory Parameter Missing         | POST /remotesep<br>POST /remotesepset                 | This error is observed if mandatory parameter is missing in request.                                                                                                                                                                                                                                                                                                                                                                                                                                                                                           | Verify that all mandatory parameter for configuration of Remote SEPP or Remote SEPP Set is present                                                                                                                                                                                                                                                                                                                                                                                                                        |



Table 3-3 (Cont.) Error Codes and Recovery Steps

| Error Code                | Error Text                                            | Command/Method                                                                                    | Description                                                                                               | Recovery Steps                                                                                                                                                                                                                               |
|---------------------------|-------------------------------------------------------|---------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SEPP-COM-xx-ERROR-0103    | Connection could not be established on N32c interface | POST / remotesepp<br>DELETE remotesepp/{name}<br>PUT remotesepp/{name}<br>PATCH remotesepp/{name} | This error is observed if n32c service is down or not up and running.                                     | Verify that n32c service is up and running                                                                                                                                                                                                   |
| SEPP-COM-xx-ERROR-0104    | Invalid Value for Parameter                           |                                                                                                   | This Error occurs when user enters the invalid value for Enum Field in SEPP.                              |                                                                                                                                                                                                                                              |
| SEPP-COM-SVR-ERROR-0404   | Unable to connect to EGW to sync config               | PUT peerconfiguration / peersetconfiguration / routesconfiguration                                | This error is observed if config mgr is not able to update peer/peerse/ routes configurations at EGW.     | Verify that common configuration server is up and running.                                                                                                                                                                                   |
| SEPP-CN32FSEPP-ERROR-0013 | PLMN ID Validation In Header Failed                   | NA                                                                                                | This error is observed if PLMN ID is not matched in header of the incoming request on CN32F microservice. | <ul style="list-style-type: none"> <li>Verify if correct regex is configured against the header identifier in Header Configuration.</li> <li>Verify if MCC &amp; MNC combination is present in the Remote PLMN ID List.</li> </ul>           |
| SEPP-CN32FSEPP-ERROR-0014 | PLMN ID Validation In Body Failed                     | NA                                                                                                | This error is observed if PLMN ID is not matched in body of the incoming request on CN32F microservice.   | <ul style="list-style-type: none"> <li>Verify if correct regex is configured against the body IE in Body IE Configuration.</li> <li>Verify if MCC &amp; MNC combination is present in the Remote PLMN ID List.</li> </ul>                    |
| SEPP-PN32FSEPP-ERROR-0016 | PLMN ID Validation In Header Failed                   | NA                                                                                                | This error is observed if PLMN ID is not matched in header of the incoming request on PN32F microservice. | <ul style="list-style-type: none"> <li>Verify if correct regex is configured against the header identifier in Header Configuration.</li> <li>Verify if MCC &amp; MNC combination is present in the helm based local PLMN ID list.</li> </ul> |

**Table 3-3 (Cont.) Error Codes and Recovery Steps**

| Error Code                           | Error Text                          | Command/Method | Description                                                                                                                                                                                                                       | Recovery Steps                                                                                                                                                                                                                      |
|--------------------------------------|-------------------------------------|----------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SEPP-PN32FSEPP-ERROR-0017            | PLMN ID Validation In Body Failed   | NA             | This error is observed if PLMN ID is not matched in body of the incoming request on PN32F microservice.                                                                                                                           | <ul style="list-style-type: none"> <li>Verify if correct regex is configured against the body IE in Body IE Configuration.</li> <li>Verify if MCC &amp; MNC combination is present in the helm based local PLMN ID list.</li> </ul> |
| SEPP-SECURITY-PLMN-HEADER-ERROR-0015 | PLMN ID Validation In Header Failed | NA             | <ul style="list-style-type: none"> <li>Verify if correct regex is configured against the header in header Configuration.</li> <li>Verify if MCC &amp; MNC combination is present in the helm based local PLMN ID list.</li> </ul> | if PLMN ID is not matched in the header of the incoming request. Metrics can be checked for the details for which it has failed.                                                                                                    |
| SEPP-SECURITY-PLMN-BODY-ERROR-0016   | PLMN ID Validation in body failed   | NA             | <ul style="list-style-type: none"> <li>Verify if correct regex is configured against the body IE in Body IE Configuration.</li> <li>Verify if MCC and MNC combination is present in the helm based local PLMN ID list.</li> </ul> | if PLMN ID is not matched in the body of the incoming request. Metrics can be checked for the details for which it has failed.                                                                                                      |

The following are the error codes of Mediation feature:

**Table 3-4 Mediation Error Codes and Recovery Steps**

| Error Code               | Error Text                        | Command/Method                                                                                                                        | Description                                 | Recovery Steps                                          |
|--------------------------|-----------------------------------|---------------------------------------------------------------------------------------------------------------------------------------|---------------------------------------------|---------------------------------------------------------|
| SEPP-MEDIATION-ERROR-001 | Mediation Trigger Rule Not Found. | GET<br>x`/sepp-mediation-trigger-rule-list/{triggerRuleListName}<br>DELETE<br>/sepp-mediation-trigger-rule-list/{triggerRuleListName} | Requested Trigger Rule List does not exist. | Give the Trigger Rule List Name that exist in Database. |

Table 3-4 (Cont.) Mediation Error Codes and Recovery Steps

| Error Code               | Error Text                                                | Command/ Method                                                   | Description                                                                                    | Recovery Steps                                                                                                                                                               |
|--------------------------|-----------------------------------------------------------|-------------------------------------------------------------------|------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SEPP-MEDIATION-ERROR-002 | Unsupported Trigger Points                                | PUT<br>/sepp-mediation-trigger-rule-list/{triggerRuleListName}    | Trigger Points provided in configuration request body is not supported.                        | Provide valid Trigger Points in Configuration Request.<br>Valid Trigger Points :<br>N32_Egress_Request<br>N32_Ingress_Response<br>N32_Ingress_Request<br>N32_Egress_Response |
| SEPP-MEDIATION-ERROR-003 | Mediation Trigger Mandatory Parameter Update Not Allowed. | PUT<br>/sepp-mediation-trigger-rule-list/{triggerRuleListName}    | Trigger Rule List mandatory parameter like TriggerRuleListName can not be updated.             | Make sure TriggerRuleListName you are providing in configuration request url is same as name in request url path.                                                            |
| SEPP-MEDIATION-ERROR-004 | Trigger Rule is Mandatory Parameter.                      | PUT<br>/sepp-mediation-trigger-rule-list/{triggerRuleListName}    | If MediationAllEnabled is false and there is no TriggerRules or empty TriggerRules in request. | If MediationAllEnabled is false, then make sure there is TriggerRules Provided in Request.                                                                                   |
| SEPP-MEDIATION-ERROR-005 | Invalid Error Status Code                                 | PUT<br>/mediation/feature                                         | Invalid Http Status Code is provided in Error Configuration Request.                           | Make sure to provide valid HTTP Status Code in statusCode field in Error Configuration Request                                                                               |
| SEPP-MEDIATION-ERROR-006 | Invalid ResourceURI and HTTPMethod Error                  | PUT<br>/sepp-mediation-trigger-rule-list/{triggerRuleListName}    | ResourceURI and HttpMethod provided in Request is not valid.                                   | Make sure to provide ResourceURI and HttpMethod combination that is configuration for SEPP, already present in Database.                                                     |
| SEPP-MEDIATION-ERROR-007 | DELETE Not Allowed                                        | DELETE<br>/sepp-mediation-trigger-rule-list/{triggerRuleListName} | Trigger Rule List Delete Not Allowed.                                                          | Ensure you are deleting only that Trigger Rule List that is not associated with Remote SEPP Set.                                                                             |

**Table 3-4 (Cont.) Mediation Error Codes and Recovery Steps**

| Error Code               | Error Text                                                              | Command/ Method                                                | Description                                                                                                         | Recovery Steps                                                                                                                                                          |
|--------------------------|-------------------------------------------------------------------------|----------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SEPP-MEDIATION-ERROR-008 | Trigger Rule Configuration Error                                        | PUT<br>/sepp-mediation-trigger-rule-list/{triggerRuleListName} | Error in Configuration of Trigger Rule List.                                                                        | Ensure there is no duplicated ResourceURI and Method in request body.                                                                                                   |
| SEPP-MEDIATION-ERROR-009 | Mediation Trigger Rules Configuration Mandatory Parameter Missing Error | PUT<br>/sepp-mediation-trigger-rule-list/{triggerRuleListName} | Mediation Trigger Rules Configuration Mandatory Parameter Missing in configuration request.                         | Ensure all mandatory parameters are present in Mediation Trigger Rule Configuration request.                                                                            |
| SEPP-MEDIATION-ERROR-010 | Multiple Local Trigger Rule List Configuration Error                    | PUT<br>/sepp-mediation-trigger-rule-list/{triggerRuleListName} | Multiple Local Trigger Rule List Configuration is Not Allowed.                                                      | Make sure we are not configuring another Local Trigger Rule List if there is already one configured in DB. Only one Local Trigger Rule List can be configured for SEPP. |
| SEPP-MEDIATION-ERROR-011 | Mediation Feature Mandatory Parameter Error                             | PUT<br>/mediation/feature                                      | Mediation Feature Configuration Mandatory Parameter is missing.                                                     | Make sure if FeatureEnabled is true in request then all field Error Configuration is present in request.                                                                |
| SEPP-MEDIATION-ERROR-012 | Mediation Local Trigger Rule IPX Mode Error                             | PUT<br>/sepp-mediation-trigger-rule-list/{triggerRuleListName} | In IPX Mode SEPP allow only 2 Trigger Points (N32 Ingress Request, N32 Egress Response) in local TRL configuration. | Make sure there is not any invalid Trigger Points like N32 Egress Request or N32 Ingress Response in Local Trigger Rule Configuration Request.                          |
| SEPP-MEDIATION-ERROR-013 | Mediation service is not available                                      | PUT<br>/mediation/feature                                      | Mediation Service Not deployed.                                                                                     | Before Enabling Mediation Feature through API , make sure Mediation Service is being deployed for SEPP.                                                                 |
| SEPP-MEDIATION-ERROR-014 | Invalid Error Action                                                    | PUT<br>/mediation/feature                                      | Invalid Error Action in Mediation Feature Configuration request.                                                    | Make sure to provide valid ErrorAction in Mediation Feature Configuration request in ErrorConfiguration section.                                                        |

The following are the error codes of Cat-1 feature:

**Table 3-5 Cat-1 Error Codes and Recovery Steps**

| Error Code                | Error Text                      | Description                                                                                                                         | Recovery                                                                                                                                                                                                                                                        |
|---------------------------|---------------------------------|-------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SEPP-SECURITY-ERROR-001   | Service API not in allowed list | If resource URI and Http Method is not matched as per the configured allowed list on SEPP.                                          | Verify whether the proper Resource URI and HTTP method is configured in the CNC Console GUI. Go to the Security Countermeasure section, check under the Service API Allowed List for that particular Allowed list name for N32 Egress or N32 Ingress Direction. |
| SEPP-CN32FSEPP-ERROR-0012 | Service API Validation Failed   | This error occurs on CN32F microservice. If resource URI and Http Method is not matched as per the configured allowed list on SEPP. | Verify whether the proper Resource URI and HTTP method is configured in the CNC Console GUI. Go to the Security Countermeasure section, check under the Service API Allowed List for that particular Allowed list name for N32 Egress or N32 Ingress Direction. |
| SEPP-PN32FSEPP-ERROR-0015 | Service API Validation Failed   | This error occurs on PN32F microservice. If resource URI and Http Method is not matched as per the configured allowed list on SEPP. | Verify whether the proper Resource URI and HTTP method is configured in the CNC Console GUI. Go to the Security Countermeasure section, check under the Service API Allowed List for that particular Allowed list name for N32 Egress or N32 Ingress Direction. |

The following are the error codes of Cat-2 Network ID Validation feature:

**Table 3-6 Cat-2 Network ID Error Codes and Recovery Steps**

| Error Code                | Error Text                             | Description                                                                                   | Recovery                                                                                                                                                                                                                                                                                    |
|---------------------------|----------------------------------------|-----------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SEPP-CN32FSEPP-ERROR-0013 | Network ID Validation In Header Failed | Check whether PLMN ID is not matched in header of the incoming request on CN32F microservice. | <ol style="list-style-type: none"> <li>1. Verify if correct regex is configured against the header identifier in Header Configuration.</li> <li>2. Verify if MCC and MNC combination is present in the visitor or target PLMN ID List based on the associated SEPP configuration</li> </ol> |

**Table 3-6 (Cont.) Cat-2 Network ID Error Codes and Recovery Steps**

| Error Code                | Error Text                             | Description                                                                                   | Recovery                                                                                                                                                                                                                                                                                     |
|---------------------------|----------------------------------------|-----------------------------------------------------------------------------------------------|----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SEPP-CN32FSEPP-ERROR-0014 | Network ID Validation In Body Failed   | Check whether PLMN ID is not matched in body of the incoming request on CN32F microservice.   | <ol style="list-style-type: none"> <li>1. Verify if correct regex is configured against the body IE in Body IE Configuration.</li> <li>2. Verify if MCC and MNC combination is present in the home or visitor PLMN ID List based on configuration of associated SEPP.</li> </ol>             |
| SEPP-PN32FSEPP-ERROR-0016 | Network ID Validation In Header Failed | Check whether PLMN ID is not matched in header of the incoming request on PN32F microservice. | <ol style="list-style-type: none"> <li>1. Verify if correct regex is configured against the header identifier in Header Configuration.</li> <li>2. Verify if MCC and MNC combination is present in the target or visitor PLMN ID list based on the associated SEPP configuration.</li> </ol> |
| SEPP-PN32FSEPP-ERROR-0017 | Network ID Validation In Body Failed   | Check whether PLMN ID is not matched in body of the incoming request on PN32F microservice.   | <ol style="list-style-type: none"> <li>1. Verify if correct regex is configured against the body IE in Body IE Configuration.</li> <li>2. Verify if MCC and MNC combination is present in the target or visitor PLMN ID list based on the associated SEPP configuration.</li> </ol>          |

**Table 3-7 Cat-3 Previous Location Check Error Codes**

| Error Code                                                                         | Error Text                                                 | Description                                                                                                                                                                                                                                                                                | Recovery                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                   |
|------------------------------------------------------------------------------------|------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| SEPP-PREVIOUS-LOCATION-CHECK-VALIDATION-ERROR-0019 or<br>SEPP-PN32FSEPP-ERROR-0018 | Previous Location Check Validation Failed                  | <p>This error code is observed only on PN32F microservice.</p> <p>If Serving Network ID is not matching against the serving network ID coming from UDR response, to check whether UE authentication is success.</p> <p>This error also occurs if the authentication from UDR is false.</p> | <ol style="list-style-type: none"> <li>1. Verify whether the correct regex is configured against the serving network identifier in either Header or Body Configuration.</li> <li>2. Verify whether the MCC and MNC combination is present in the Serving Network ID.</li> <li>3. Verify whether the identifier coming in the ingress request on PN32F microservice is also same as the serving Network name coming as part of the UDR response if UE authentication is successful.</li> </ol>                                                                                              |
| SEPP-PREVIOUS-LOCATION-CHECK-VALIDATION-EXCEPTION-0020                             | Previous Location Check Validation Failed Due To Exception | <p>This error will occur if the system is not able to extract the SUPI, or if the incoming message doesn't contain SUPI, or if there are any sort of connectivity issues with NRF or UDR.</p>                                                                                              | <ol style="list-style-type: none"> <li>1. Verify if correct regex for UE ID in Header or Body configuration screen is configured due to which correct UE ID value is extracted.</li> <li>2. Verify if the incoming message has SUPI.</li> <li>3. Verify whether the FQDN or IP fetched for the UDR as part of NRF discovery call is reachable.</li> <li>4. Verify if UDR discovery procedure from NRF is successful.</li> <li>5. Verify if the SUPI received in the Ingress request message is part of the SUPI range received in UDR profile from discovery response from NRF.</li> </ol> |





# 4

## Debug Tool

### Overview

The Debug Tool provides third-party troubleshooting tools for debugging the runtime issues in a lab environment.

Following are the available tools:

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

### Preconfiguration Steps

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

#### Note

- For the CNE 23.2.0 and later versions, follow the [Step a](#) of Configuration in CNE to Update the Cluster Policies and Add Namespace.
- For the CNE 23.1.x and previous versions, follow the [Step b](#) of Configuration in CNE for PodSecurityPolicy (PSP) Creation, Role Creation, and RoleBinding Creation.

### 1. Configuration in CNE

Perform the following configurations in the Bastion Host. You need admin privileges to perform these configurations.

- a. When NEF is installed on CNE version 23.2.0 or above

#### Note

- In CNE version 23.2.0 or above, the default CNE 23.2.0 Kyverno policy, disallow-capabilities, do not allow NET\_ADMIN and NET\_RAW capabilities that are required for debug tool.
- To run Debug tool on CNE 23.2.0 and above, the user must modify the existing Kyverno policy, disallow-capabilities, as below.

### Adding a Namespace to an Empty Resource

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

Example:

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

Sample output:

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

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

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

Example:

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

Sample output:

```
apiVersion: kyverno.io/v1
kind: ClusterPolicy
...
...
spec:
 rules:
 -exclude:
 resources:
 namespaces:
 - sepp1
```

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

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

Sample output:

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

### Adding a Namespace to an Existing Namespace List

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

*Example:*

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

Sample output:

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

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

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

*Example:*

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

Sample output:

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

```
...
spec:
 rules:
 -exclude:
 resources:
 namespaces:
 -namespace1
 -namespace2
 -namespace3
 - sepp1
```

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

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

Example:

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

Sample output:

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

#### Note

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

- b. When NEF is installed on CNE version prior to 23.2.0

### PodSecurityPolicy (PSP) Creation

1. Log in to the Bastion Host.
2. Create a new PSP by running the following command from the bastion host. The parameters **readOnlyRootFilesystem**, **allowPrivilegeEscalation**, **allowedCapabilities** are required by the debug container.

**Note**

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

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

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

**Role Creation**

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

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

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

### RoleBinding Creation

Run the following command to attach the service account for your NF namespace with the role created for the tool PSP:

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

For parameter details, see [Debug Tool Configuration Parameters](#).

#### 1. Configuration in NF specific Helm

Following updates must be performed in custom\_values.yaml file.

- a. Log in to the NF server.
- b. Open the custom\_values file:

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

- c. Under global configuration, add the following:

```
Allowed Values: DISABLED, ENABLED
podSecurityPolicy: "DISABLED"
extraContainers: "DISABLED"
debugToolContainerMemoryLimit: 4Gi
extraContainersImageDetails:
 image: ocdebugtool/ocdebug-tools
 tag: debug_container_tag
 imagePullPolicy: Always
extraContainersVolumesTpl: |
- name: debug-tools-dir
 emptyDir:
 medium: Memory
 sizeLimit: {{ .Values.global.debugToolContainerMemoryLimit |
quote }}
extraContainersTpl: |-
- command:
 - /bin/sleep
 - infinity
```

```

name: tools
resources:
 requests:
 ephemeral-storage: "512Mi"
 cpu: "0.5"
 memory: {{ .Values.global.debugToolContainerMemoryLimit |
quote }}
 limits:
 ephemeral-storage: "512Mi"
 cpu: "1"
 memory: {{ .Values.global.debugToolContainerMemoryLimit |
quote }}
securityContext:
 allowPrivilegeEscalation: true
capabilities:
 drop:
 - ALL
 add:
 - NET_RAW
 - NET_ADMIN
 runAsUser: 1012
volumeMounts:
- mountPath: /tmp/tools
 name: debug-tools-dir

```

#### **Note**

- Debug Tool Container comes up with the default user ID - 7000. If you want to override this default value, use the `runAsUser` field, or else, you can skip the field.

Default value: uid=7000(debugtool) gid=7000(debugtool)  
groups=7000(debugtool)

- In case you want to customize the container name, replace the `name` field in the above values.yaml with the following:

```

name: {{ printf "%s-tools-%s" (include "getprefix" .)
(include "getsuffix" .) | trunc 63 | trimPrefix "-" |
trimSuffix "-" }}

```

This will ensure that the container name is prefixed and suffixed with the necessary values.

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

```

Allowed Values: DISABLED, ENABLED, USE_GLOBAL_VALUE
extraContainers: USE_GLOBAL_VALUE

```

### Note

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

## Run the Debug Tool

Following is the procedure to run Debug Tool.

Run the following command to enter Debug Tool Container:

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

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

Example:

```
$ kubectl get pods -n seppsvc
```

Sample Output:

| NAME                                                 | READY | STATUS |
|------------------------------------------------------|-------|--------|
| ocsepp-release-appinfo-75894d8d8c-4zzkt              |       | 2/2    |
| Running 0 5m54s                                      |       |        |
| ocsepp-release-cn32c-svc-5f5cdbfb7f-kspw6            |       | 2/2    |
| Running 0 5m55s                                      |       |        |
| ocsepp-release-cn32f-svc-5458886cc7-nm7c8            |       | 2/2    |
| Running 0 5m55s                                      |       |        |
| ocsepp-release-config-mgr-svc-6c94c449f-v8qnv        |       | 2/2    |
| Running 0 5m55s                                      |       |        |
| ocsepp-release-n32-egress-gateway-55ccbbf46f-bb4tp   |       | 3/3    |
| Running 0 5m54s                                      |       |        |
| ocsepp-release-n32-ingress-gateway-7bd984c9c6-pcpqd  |       | 3/3    |
| Running 0 5m54s                                      |       |        |
| ocsepp-release-ocpm-config-65dd85d96d-59t4w          |       | 2/2    |
| Running 0 5m54s                                      |       |        |
| ocsepp-release-performance-7456bbd8-2j7dx            |       | 2/2    |
| Running 0 5m54s                                      |       |        |
| ocsepp-release-plmn-egress-gateway-67b7864664-cmcf8  |       | 3/3    |
| Running 0 5m54s                                      |       |        |
| ocsepp-release-plmn-egress-gateway-67b7864664-lwhxz  |       | 3/3    |
| Running 0 4m31s                                      |       |        |
| ocsepp-release-plmn-ingress-gateway-596c78f967-sc44c |       | 3/3    |
| Running 0 5m53s                                      |       |        |
| ocsepp-release-pn32c-svc-6498f6dc-lrvtt              |       | 2/2    |
| Running 0 5m53s                                      |       |        |
| ocsepp-release-pn32f-svc-59bcb4c545-c4pqj            |       | 2/2    |



```

Running 0 5m53s
ocsepp-release-sepp-nrf-client-nfdiscovery-9db8957cb-47j6g 1/1
Running 0 5m54s
ocsepp-release-sepp-nrf-client-nfmanagement-5ddfd8d754-nbx69 1/1
Running 0 5m54s
sepp-mysql-54b7c5699d-5nmzc 1/1
Running 0 3d23h

```

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

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

Example:

```
$ kubectl exec -it ocsepp-release-cn32c-svc-5f5cdbfb7f-kspw6 -c tools -n
seppsvc bash
```

3. Run the commands supported by debug tools:

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

Example:

```
bash -4.2$ tcpdump
```

4. Copy the output files from container to host:

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

Example:

```
$ kubectl cp -c tools ocsepp-release-cn32c-svc-5f5cdbfb7f-kspw6:/tmp/
capture.pcap -n seppsvc /tmp/
```

### Tools Tested in Debug Container

Following is the list of debugging tools that are tested.

#### tcpdump

Table 4-1 tcpdump

| Options Tested | Description                                                                                                       | Output                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                     | Capabilities          |
|----------------|-------------------------------------------------------------------------------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------|
| -D             | Print the list of the network interfaces available on the system and on which <i>tcpdump</i> can capture packets. | <pre>tcpdump -D</pre> <ol style="list-style-type: none"> <li>1. eth02.</li> <li>2. nflog (Linux netfilter log (NFLOG) interface)</li> <li>3. nfqueue (Linux netfilter queue (NFQUEUE) interface)</li> <li>4. any (Pseudo-device that captures on all interfaces)</li> <li>5. lo [Loopback]</li> </ol>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                      | NET_ADMIN,<br>NET_RAW |
| -i             | Listen on <i>interface</i>                                                                                        | <pre>tcpdump -i eth0</pre> <pre>tcpdump: verbose output suppressed, use -v or -vv for full protocol decoding</pre> <pre>listening on eth0, link-type EN10MB (Ethernet), capture size 262144 bytes</pre> <pre>12:10:37.381199 IP ocsepp-plmn-ingress-gateway-7ffc49bb7f-2kkhc.46519 &gt; kubernetes.default.svc.cluster.local.https: Flags [P.], seq 1986927241:1986927276, ack 1334332290, win 626, options [nop,nop,TS val 849591834 ecr 849561833], length 35</pre> <pre>12:10:37.381952 IP ocsepp-plmn-ingress-gateway-7ffc49bb7f-2kkhc.45868 &gt; kube-dns.kube-system.svc.cluster.local.domain: 62870+ PTR? 1.0.96.10.in-addr.arpa. (40)</pre>                                                                                                                                                                                        | NET_ADMIN,<br>NET_RAW |
| -w             | Write the raw packets to file rather than parsing and printing them out.                                          | <pre>tcpdump -w capture.pcap -i eth0</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                 | NET_ADMIN,<br>NET_RAW |
| -r             | Read packets from <i>file</i> (which was created with the <b>-w</b> option).                                      | <pre>tcpdump -r capture.pcap</pre> <pre>reading from file /tmp/capture.pcap, link-type EN10MB (Ethernet)</pre> <pre>12:13:07.381019 IP ocsepp-plmn-ingress-gateway-7ffc49bb7f-2kkhc.46519 &gt; kubernetes.default.svc.cluster.local.https: Flags [P.], seq 1986927416:1986927451, ack 1334332445, win 626, options [nop,nop,TS val 849741834 ecr 849711834], length 35</pre> <pre>12:13:07.381194 IP kubernetes.default.svc.cluster.local.https &gt; ocsepp-plmn-ingress-gateway-7ffc49bb7f-2kkhc.46519: Flags [P.], seq 1:32, ack 35, win 247, options [nop,nop,TS val 849741834 ecr 849741834], length 31</pre> <pre>12:13:07.381207 IP ocsepp-plmn-ingress-gateway-7ffc49bb7f-2kkhc.46519 &gt; kubernetes.default.svc.cluster.local.https: Flags [.], ack 32, win 626, options [nop,nop,TS val 849741834 ecr 849741834], length 0</pre> | NET_ADMIN,<br>NET_RAW |

ip

Table 4-2 ip

| Options Tested | Description                 | Output                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                               | Capabilities |
|----------------|-----------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|
| addr show      | Look at protocol addresses. | <pre>ip addr show 1: lo: &lt;LOOPBACK,UP,LOWER_UP&gt; mtu 65536 qdisc noqueue state UNKNOWN group defaultlink/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00inet 127.0.0.1/8 scope host lovalid_lft forever preferred_lft forever2: tunl0@NONE: &lt;NOARP&gt; mtu 1480 qdisc noop state DOWN group defaultlink/ipip 0.0.0.0 brd 0.0.0.04: eth0@if190: &lt;BROADCAST,MULTICAST,UP,LOWER_UP&gt; mtu 1440 qdisc noqueue state UP group defaultlink/ether aa:5a:27:8d:74:6f brd ff:ff:ff:ff:ff:ff link-netnsid 0inet 192.168.219.112/32 scope global eth0valid_lft forever preferred_lft forever</pre> | --           |
| route show     | List routes                 | <pre>ip route show default via 169.254.1.1 dev eth0 169.254.1.1 dev eth0 scope link</pre>                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                            | --           |
| addrlabel list | List address labels         | <pre>ip addrlabel list prefix ::1/128 label 0 prefix ::/96 label 3 prefix ::ffff:0.0.0.0/96 label 4 prefix 2001::/32 label 6 prefix 2001:10::/28 label 7 prefix 3ffe::/16 label 12 prefix 2002::/16 label 2 prefix fec0::/10 label 11 prefix fc00::/7 label 5 prefix ::/0 label 1</pre>                                                                                                                                                                                                                                                                                                              | --           |

**netstat**

Table 4-3 netstat

| Options Tested | Description                                                                                  | Output                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                        | Capabilities |
|----------------|----------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|
| -a             | Show both listening and non-listening (for TCP, this means established connections) sockets. | <pre>netstat -a Active Internet connections (servers and established)Proto Recv-Q Send-Q Local Address Foreign Address Statetcp 0 0 0.0.0.0:tpoxy 0.0.0.0:* LISTENTcp 0 0 0.0.0.0:websm 0.0.0.0:* LISTENTcp 0 0 cncc-core-ingress:websm 10-178-254-194.ku:47292 TIME_WAITtcp 0 0 cncc-core-ingress:46519 kubernetes.default:https ESTABLISHEDtcp 0 0 cncc-core-ingress:websm 10-178-254-194.ku:47240 TIME_WAITtcp 0 0 cncc-core-ingress:websm 10-178-254-194.ku:47347 TIME_WAITudp 0 0 localhost:59351 localhost:ambit-lm ESTABLISHEDActive UNIX domain sockets (servers and established)Proto RefCnt Flags Type State I-Node Pathunix 2 [ ] STREAM CONNECTED 576064861</pre> | --           |

Table 4-3 (Cont.) netstat

| Options Tested | Description                                   | Output                                                                                                                                                                                                                                                                                                           | Capabilities |
|----------------|-----------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|--------------|
| -l             | Show only listening sockets.                  | netstat -l<br>Active Internet connections (only servers)Proto Recv-Q Send-Q Local Address Foreign Address State tcp 0 0 0.0.0.0:tpoxy 0.0.0.0:* LISTEN tcp 0 0 0.0.0.0:websm 0.0.0.0:* LISTEN Active UNIX domain sockets (only servers)Proto RefCnt Flags Type State I-Node Path                                 | --           |
| -s             | Display summary statistics for each protocol. | netstat -s<br>Ip:4070 total packets received0 forwarded0 incoming packets discarded4070 incoming packets delivered4315 requests sent outicmp:0 ICMP messages received0 input ICMP message failed.ICMP input histogram:2 ICMP messages sent0 ICMP messages failedICMP output histogram:destination unreachable: 2 | --           |
| -i             | Display a table of all network interfaces.    | netstat -i<br>Kernel Interface tableIface MTU RX-OK RX-ERR RX-DRP RX-OVR TX-OK TX-ERR TX-DRP TX-OVR Flgeth0 1440 4131 0 0 0 4355 0 0 0 BMRUIo 65536 0 0 0 0 0 0 0 LRU                                                                                                                                            | --           |

**curl**

Table 4-4 curl

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

**ping**

Table 4-5 ping

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

**dig**

Table 4-6 dig

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

## 4.1 Debug Tool Configuration Parameters

Following are the parameters used to configure debug tool.

### OCCNE Parameters

Table 4-7 OCCNE Parameters

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

### Role Creation Parameters

Table 4-8 Role Creation

| Parameter  | Description                                                                   |
|------------|-------------------------------------------------------------------------------|
| apiVersion | APIVersion defines the versioned schema of this representation of an object.  |
| kind       | Kind is a string value representing the REST resource this object represents. |

**Table 4-8 (Cont.) Role Creation**

| Parameter           | Description                                                                                                    |
|---------------------|----------------------------------------------------------------------------------------------------------------|
| metadata            | Standard object's metadata.                                                                                    |
| metadata.name       | Name must be unique within a namespace.                                                                        |
| metadata.namespace  | Namespace defines the space within which each name must be unique.                                             |
| rules               | Rules holds all the PolicyRules for this Role                                                                  |
| apiGroups           | APIGroups is the name of the APIGroup that contains the resources.                                             |
| rules.resources     | Resources is a list of resources this rule applies to.                                                         |
| rules.verbs         | Verbs is a list of Verbs that apply to ALL the ResourceKinds and AttributeRestrictions contained in this rule. |
| rules.resourceNames | ResourceNames is an optional allowed list of names that the rule applies to.                                   |

**Table 4-9 Role Binding Creation**

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

**Debug Tool Configuration Parameters****Table 4-10 Debug Tool Configuration Parameters**

| Parameter                     | Description                                                                       |
|-------------------------------|-----------------------------------------------------------------------------------|
| extraContainers               | Specifies the spawns debug container along with application container in the pod. |
| debugToolContainerMemoryLimit | Indicates the memory assigned for the debug tool container.                       |
| extraContainersVolumesTpl     | Specifies the extra container template for the debug tool volume.                 |

**Table 4-10 (Cont.) Debug Tool Configuration Parameters**

| Parameter                                    | Description                                                                                                                                                                                |
|----------------------------------------------|--------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| extraContainersVolumesTpl.name               | Indicates the name of the volume for debug tool logs storage.                                                                                                                              |
| extraContainersVolumesTpl.emptyDir.medium    | Indicates the location where emptyDir volume is stored.                                                                                                                                    |
| extraContainersVolumesTpl.emptyDir.sizeLimit | Indicates the emptyDir volume size.                                                                                                                                                        |
| command                                      | String array used for container command.                                                                                                                                                   |
| image                                        | Docker image name                                                                                                                                                                          |
| imagePullPolicy                              | Image Pull Policy                                                                                                                                                                          |
| name                                         | Name of the container                                                                                                                                                                      |
| resources                                    | Compute Resources required by this container                                                                                                                                               |
| resources.limits                             | Limits describes the maximum amount of compute resources allowed                                                                                                                           |
| resources.requests                           | Requests describes the minimum amount of compute resources required                                                                                                                        |
| resources.limits.cpu                         | CPU limits                                                                                                                                                                                 |
| resources.limits.memory                      | Memory limits                                                                                                                                                                              |
| resources.limits.ephemeral-storage           | Ephemeral Storage limits                                                                                                                                                                   |
| resources.requests.cpu                       | CPU requests                                                                                                                                                                               |
| resources.requests.memory                    | Memory requests                                                                                                                                                                            |
| resources.requests.ephemeral-storage         | Ephemeral Storage requests                                                                                                                                                                 |
| securityContext                              | Security options the container should run with.                                                                                                                                            |
| securityContext.allowPrivilegeEscalation     | AllowPrivilegeEscalation controls whether a process can gain more privileges than its parent process. This directly controls if the no_new_privs flag will be set on the container process |
| securityContext.capabilities                 | The capabilities to add or drop when running containers. Defaults to the default set of capabilities granted by the container runtime.                                                     |
| securityContext.capabilities.drop            | Removed capabilities                                                                                                                                                                       |
| securityContext.capabilities.add             | Added capabilities                                                                                                                                                                         |
| securityContext.runAsUser                    | The UID to run the entry point of the container process.                                                                                                                                   |
| volumeMounts.mountPath                       | Indicates the path for volume mount.                                                                                                                                                       |
| volumeMounts.name                            | Indicates the name of the directory for debug tool logs storage.                                                                                                                           |