Oracle® Communications

Cloud Native Core, Service Communication Proxy Troubleshooting Guide

Release 24.2.6
F99449-07
October 2025

Oracle Communications Cloud Native Core, Service Communication Proxy Troubleshooting Guide, Release 24.2.6

F99449-07

Contents

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

Table Acronyms

Acronym	Meaning
ASM	Aspen Service Mesh
CNE	Oracle Communications Cloud Native Core, Cloud Native Environment
NRF	Oracle Communications Cloud Native Core, Network Repository Function
OCI	Oracle Cloud Infrastructure
OHC	Oracle Help Center
OSDC	Oracle Software Delivery Cloud
SCP	Oracle Communications Cloud Native Core, Service Communication Proxy
SCPC	Service Communication Proxy Control Plane
SVC	Services

What's New in This Guide

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

Release 24.2.6 - F99449-07, October 2025

Updated the release number to 24.2.6 throughout the document.

Release 24.2.5 - F99449-06, July 2025

Updated the release number to 24.2.5 throughout the document.

Release 24.2.4 - F99449-05, April 2025

Updated the release number to 24.2.4 throughout the document.

Release 24.2.3 - F99449-04, January 2025

Updated the release number to 24.2.3 throughout the document.

Release 24.2.2 - F99449-03, November 2024

Updated the release number to 24.2.2 throughout the document.

Release 24.2.1 - F99449-02, September 2024

Updated the release number to 24.2.1 throughout the document.

Release 24.2.0 - F99449-01, July 2024

- Added the following alerts for the OAuth2.0 feature:
  - SCPNrfProxyOauthQueuesUtilizationAboveCriticalThreshold
  - SCPNrfProxyOauthQueuesUtilizationAboveMajorThreshold
  - SCPNrfProxyOauthQueuesUtilizationAboveMinorThreshold
- Added the following alerts for the Support for TLS 1.3 feature:
  - SCPWorkerSSLCertificateOnCriticalExpiry
  - SCPWorkerSSLCertificateOnMajorExpiry
  - SCPWorkerSSLCertificateOnMinorExpiry
  - SCPIngressConnectionEstablishmentFailure
  - SCPEgressConnectionEstablishmentFailure
- Removed the following alerts for the Support for TLS 1.3 feature:
  - SCPWorkerHTTPSConnectionFailure
  - SCPWorkerSSLCertificateExpire

1 Introduction

This document provides information about Oracle Communications Cloud Native Core, Service Communication Proxy troubleshooting scenarios.

Overview

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

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

This guide provides information about resolving problems you might experience while installing, configuring, and upgrading SCP. This document also provides information about tools available to help you collect and analyze diagnostic data.

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

1.1 References

- *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Service Communication Proxy User Guide*
- *Oracle Communications Cloud Native Core, Cloud Native Environment Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Network Repository Function User Guide*
- *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*
- *External reference: Drools User Guide*

2 Troubleshooting Service Communication Proxy

This section provides information to troubleshoot common errors that occur while installing and upgrading Service Communication Proxy (SCP).

> **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.

2.1 Generic Checklist

The following sections provide a generic checklist for troubleshooting tips.

Deployment Related Tips

Perform the following checks before the deployment:

- Are OCSCP deployment, pods, and services created, running, and available?
  To check this, run the following command:

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

  Inspect the output and check the following columns:

  - READY, STATUS, and RESTARTS
  - PORT(S)
- Is the correct image used and the correct environment variables set in the deployment?
  To check this, run the following command:

  ```
  # kubectl -n <namespace> get deployment <deployment-name> -o yaml
  ```
- Check if the microservices can access each other through REST interface.
  To check this, run the following command:

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

Example:

```
kubectl exec -it ocscp-scpc-subscription-6bf9b7d69f-qvcnn -n ocscp curl
http://ocscp-scpc-notification:8082/ocscp/scpc-notification/v2/
compositecustomobjects


kubectl exec -it ocscp-scpc-notification-dd5c74869-cswkb -n ocscp curl
http://ocscp-scpc-configuration:8081/ocscp/scpc-configuration/v1/
systemoptions/sse
```

> **Note**
>
> These commands are in their simplest form and display the logs only if scpc-notification and scpc-configuration pods are deployed.

The list of URIs for all the microservices:

- scp-worker: `http://ocscp-scp-worker:8000/hostNFMapper`
- scpc-configuration: `http://ocscp-scpc-configuration:8081/ocscp/scpc-configuration/v1/systemoptions/sse` Or any other configuration URI from SWAGGER-UI
- scpc-notification: `http://ocscp-scpc-notification:8082/ocscp/scpc-notification/v2/compositecustomobjects`
- scpc-subscription: `http://ocscp-scpc-subscription:8080/ocscp/scpc-subscription/v1/appstate`

Application Related Tips

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

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

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

Example:

```
# kubectl -n scp-svc logs -f $(kubectl -n scp-svc get pods -o name|cut -d'/' -
f 2|grep nfr)
```

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

2.2 Helm Install Failure

This section describes Helm installation failure scenarios.

2.2.1 Incorrect Image Name in the ocscp-custom-values Files

Problem

`helm install` might fail if incorrect image name is provided in the `ocscp_values.yaml` file.

Error Code or Error Message

When you run `kubectl get pods -n <ocscp_namespace>`, the status of the pods might be ImagePullBackOff or ErrImagePull.

Solution

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

1. Edit the `ocscp_values.yaml` file and provide release specific image name and tags.
2. Run the `helm install` command.
3. Run the `kubectl get pods -n <ocscp_namespace>` command to verify if the status of all the pods is **Running**.

2.2.2 Docker Registry is Incorrectly Configured

Problem

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

Error Code or Error Message

When you run `kubectl get pods -n <ocscp_namespace>`, the status of the pods might be ImagePullBackOff or ErrImagePull.

Solution

Configure docker registry on all primary and secondary nodes.

For information about docker registry configuration, see *Oracle Communications Cloud Native Core, Cloud Native Environment Installation, Upgrade, and Fault Recovery Guide*.

2.2.3 Continuous Restart of Pods

Problem

`helm install` might fail if MySQL primary and secondary hosts may not be configured properly in ocscp-custom-values.yaml.

Error Code/Error Message

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

Solution

MySQL servers may not be configured properly. For more information about the MySQL configuration, see *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

2.3 Custom Value File Parse Failure

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

Problem

Unable to parse the `ocscp_values-x.x.x.yaml` file while running Helm install.

Error Code or Error Message

Error: failed to parse ocscp_values-x.x.x.yaml: error converting YAML to JSON: yaml

Symptom

When parsing the `ocscp-custom-values-x.x.x.yaml` file, if the above mentioned error is received, it indicates that the file is not parsed because of the following reasons:

- The tree structure may not have been followed
- There may be a tab spaces in the file

Solution

Download the `ocscp_csar_23_2_0_0_0.zip` folder from MOS and complete the steps as described in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

2.4 Curl HTTP2 Not Supported

Problem

curl http2 is not supported on the system.

Error Code or Error Message

Unsupported protocol error is thrown or connection is established with HTTP/1.1 200 OK

Symptom

If unsupported protocol error is thrown or connection is established with http1.1, it is an indication that curl http2 support may not be present on your machine.

Solution

> **Note**
>
> You must check the software platform policies before executing the following procedure.

Following is the procedure to install curl with HTTP2 support:

1. Run the following command to ensure that Git is installed:

```
$ sudo yum install git -y
```

2. Run the following commands to install nghttp2:

```
$ git clone https://github.com/tatsuhiro-t/nghttp2.git
$ cd nghttp2


$ autoreconf -i
$ automake
$ autoconf


$ ./configure
$ make
$ sudo make install


$ echo '/usr/local/lib' > /etc/ld.so.conf.d/custom-libs.conf


$ ldconfig
```

3. Run the following commands to install the latest Curl:

```
$ wget http://curl.haxx.se/download/curl-7.46.0.tar.bz2  (NOTE: Check for
latest version during Installation)
$ tar -xvjf curl-7.46.0.tar.bz2
$ cd curl-7.46.0


$ ./configure --with-nghttp2=/usr/local --with-ssl


$ make


$ sudo make install


$ sudo ldconfig
```

4. Run the following command to ensure that HTTP2 is added in features:

```
$ curl --http2-prior-knowledge -v "<http://10.75.204.35:32270/nscp-
disc/v1/nf-instances?requester-nf-type=AMF&target-nf-type=SMF>"
```

2.5 SCP DB goes into the Deadlock State

Problem

MySQL locks get struck.

Error Code/Error Message

ERROR 1213 (40001): Deadlock found when trying to get lock; try restarting transaction.

Symptom

Unable to access MySQL.

Solution

Following is the procedure to remove the deadlock:

1. Run the following command on each SQL node:

   ```
   SELECT
   CONCAT('KILL ', id, ';')
   FROM INFORMATION_SCHEMA.PROCESSLIST
   WHERE `User` = <DbUsername>
   AND `db` = <DbName>;
   ```

   This command retrieves the list of commands to kill each connections.

   Example:

   ```
   select
    CONCAT('KILL ', id, ';')
    FROM INFORMATION_SCHEMA.PROCESSLIST
    where `User` = 'scpuser'
         AND `db` = 'ocscpdb';
   +-------------------------+
   | CONCAT('KILL ', id, ';') |
   +-------------------------+
   | KILL 204491;            |
   | KILL 200332;            |
   | KILL 202845;            |
   +-------------------------+
   3 rows in set (0.00 sec)
   ```

2. Run the kill command on each SQL node.

2.6 Helm Test Error Scenarios

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

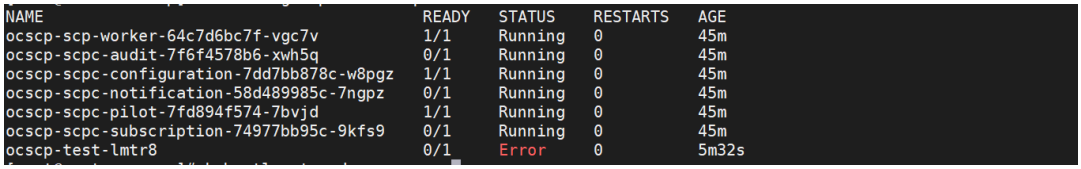
> **Note**
>
> You must clean up any residual job from the SCP deployment before performing the following procedure.

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

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

2. Check for the Helm Test pod that is in the error state:

Figure 2-1 Helm test pod



3. Run the following command to retrieve the logs:

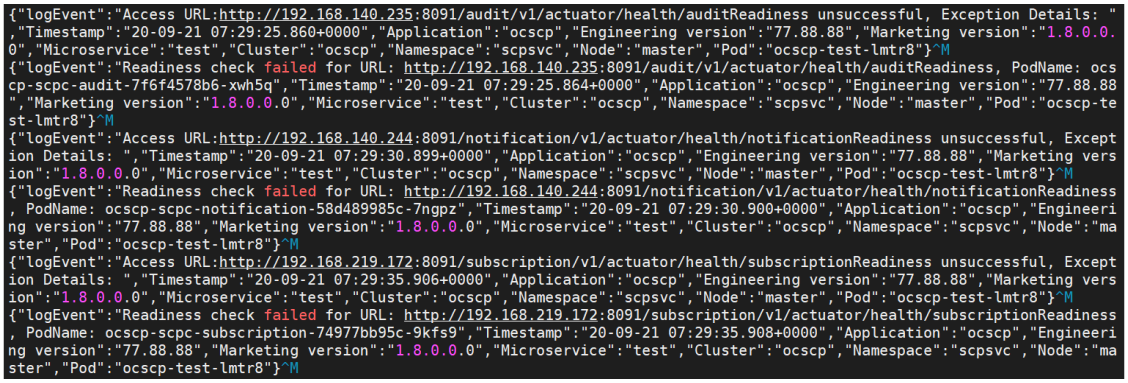
```
kubectl logs <podname> -n <namespace>
```

Readiness Probe Failure

Helm install might fail due to the readiness probe URL failure.

In case the following error appears, check for readiness probe URL correctness in the particular microservice helm charts under charts folder:

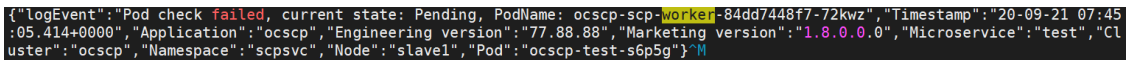
Figure 2-2 Readiness Probe Failure



Low Resources

Helm install might fail due to low resource and following error may appear:

Figure 2-3 Low Resource



In this case, check the CPU and memory availability in the Kubernetes cluster.

2.7 Using Debug Tool

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

- tcpdump

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

2.7.1 Prerequisites to Use the Debug Tool

This section describes the prerequisites for using the debug tool.

> **Note**
>
> - For CNE 23.2.0 and later versions, follow Step 1.
> - For CNE versions prior to 23.2.0, follow Step 2.
> - The debug tool requires security context with the following permissions:
>
>
> ```
> securityContext:
>         allowPrivilegeEscalation: true
>         capabilities:
>           drop:
>           - ALL
>           add:
>           - NET_RAW
>           - NET_ADMIN
>         runAsUser: <user-id>
>       volumeMounts:
>       - mountPath: /tmp/tools
>         name: debug-tools-dir
> ```
>
> For OpenShift environment, security context constraint must exist to allow above permissions to enable debug tool deployment.

Ensure that you have admin privileges.

1. If you are using CNE 23.2.0 and later versions, add a namespace to an empty resource by running the following command to verify if the current disallow-capabilities cluster policy has namespace in it:

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

   Sample output:

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

```
-exclude:
    any:
    -resources:{}
```

a. 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":["ocscp"]} }]'
```

Sample output:

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

b. If it is required to remove the namespace added in the above mentioned step, run 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:{}
```

c. To add a namespace to an existing namespace list, run the following command to verify if the current disallow-capabilities cluster policy has namespaces in it:

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

Sample output:

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

d. If namespaces are already added, then patch the policy by running 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": "ocscp" }]'
```

Sample output:

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

e. If it is required to remove the namespace added in the above mentioned step, run 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'.

2. If you are using CNE versions prior to 23.2.0, create PodSecurityPolicy (PSP) in the Bastion Host by following these steps:
   a. Log in to the Bastion Host.
   b. Run the following command to create a new PSP:

> **Note**
>
> `readOnlyRootFileSystem`, `allowPrivilegeEscalation`, and `allowedCapabilities` parameters are required by the debug container. Other parameters are mandatory for PSP creation and can be customized as per the CNE environment. It is recommended to use the default values.

```
$ 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
```

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

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

d. Run the following command to associate 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: ocscp
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 information about parameters, see Debug Tool Configuration Parameters.

3. To complete the NF specific Helm configurations, do the following:
   a. Log in to the NF server.
   b. Run the following command to open the `ocscp_custom_values.yaml` file:

      ```
      $ vim <ocscp_custom_values file>
      ```

   c. In the global configuration section, add the following:

      ```
      extraContainersTpl: |
          - command:
              - /bin/sleep
              - infinity
            name: tools
            resources:
              requests:
                cpu: "1"
                memory: {{ .Values.global.debugToolContainerMemoryLimit |
      quote }}
                ephemeral-storage: "2Gi"
              limits:
                cpu: "1"
                memory: {{ .Values.global.debugToolContainerMemoryLimit |
      quote }}
                ephemeral-storage: "4Gi"
            securityContext:
              allowPrivilegeEscalation: true
              capabilities:
                drop:
                - ALL
                add:
                - NET_RAW
                - NET_ADMIN
              runAsUser: <user-id>
            volumeMounts:
            - mountPath: /tmp/tools
              name: debug-tools-dir
      ```

> **Note**
>
> - The Debug Tool Container has a default user ID: 7000. If the you want to override this default value, it can be done using the runAsUser field. Otherwise, the field can be skipped.
> Default value: uid=7000(debugtool) gid=7000(debugtool) groups=7000(debugtool)
> - If you want to customize the container name, replace the name field in the above mentioned `values.yaml` with the following:
>
> ```
> name: {{ printf "%s-tools-%s" (include "getprefix" .)
> (include "getsuffix" .) | trunc 63 | trimPrefix "-" |
> trimSuffix "-"  }}
> ```
>
> This ensures that the container name is prefixed and suffixed with the required values.

d. In 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, the `extraContainers` parameter 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 parameter.
> - At the service level, the `extraContainers` parameter determines whether to use the extra container configuration from the global level, or enable or disable injecting extra containers for the specific service.

2.7.2 Running the Debug Tool

Perform the following procedure to run the debug tool.

1. To enter the debug tool container, run the following command to retrieve the pod details:

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

Example:

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

Sample Output:

```
NAME                                        READY   STATUS    RESTARTS
AGE
ocscp-scp-worker-58c8469595-5d6gc           2/2     Running   0
```

```
3d12h
ocscp-scpc-audit-d8f5cdc96-4nw7c            2/2     Running   0
3d12h
ocscp-scpc-configuration-774f9bc65b-f9ttn   2/2     Running   0
3d12h
ocscp-scpc-notification-779965766-9ckr5     2/2     Running   0
3d12h
ocscp-scpc-subscription-7bf6d6c884-2r4vp    2/2     Running   0
3d12h
```

2. Run the following command to enter the debug tool container:

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

   Example:

   ```
   $ kubectl exec -it ocscp-scpc-notification-779965766-9ckr5 -c tools -n
   ocscp bash
   ```

3. Run the debug tools:

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

   Example:

   ```
   bash -4.2$ tcpdump
   ```

4. Copy the output files from container to host:

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

   Example:

   ```
   $ kubectl cp -c tools ocscp-scpc-notification-779965766-9ckr5:/tmp/
   capture.pcap -n ocscp /tmp/
   ```

2.7.3 Debug Tool Configuration Parameters

Following are the parameters used to configure the debug tool.

CNE Parameters

Table 2-1 CNE Parameters

Parameter	Description
apiVersion	Defines the version schema of this representation of an object.
kind	Indicates a string value representing the REST resource this object represents.
metadata	Indicates the metadata of Standard object.

Table 2-1 (Cont.) CNE Parameters

Parameter	Description
metadata.name	Indicates the metadata name that must be unique within a namespace.
spec	Defines the policy enforced.
spec.readOnlyRootFilesystem	Controls whether the containers run with a read-only root filesystem, that is, no writable layer.
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 allowed values correspond to the volume sources that are defined when creating a volume.

Role Creation Parameters

Table 2-2 Role Creation

Parameter	Description
apiVersion	Defines the versioned schema of this representation of an object.
kind	Indicates a string value representing the REST resource this object represents.
metadata	Indicates the metadata of Standard object.
metadata.name	Indicates the name of metadata that must be unique within a namespace.
metadata.namespace	Defines the space within which each name must be unique.
rules	Manages all the PolicyRules for this Role.
apiGroups	Indicates the name of the APIGroup that contains the resources.
rules.resources	Indicates the list of resources this rule applies to.
rules.verbs	Indicates the list of Verbs that applies to ALL the ResourceKinds and AttributeRestrictions contained in this rule.
rules.resourceNames	Indicates an optional white list of names that the rule applies to.

Table 2-3 Role Binding Creation

Parameter	Description
apiVersion	Defines the versioned schema of this representation of an object.
kind	Indicates the string value representing the REST resource this object represents.
metadata	Indicates the metadata of Standard object.
metadata.name	Indicates the name that must be unique within a namespace.
metadata.namespace	Defines the space within which each name must be unique.
roleRef	References a Role in the current namespace or a ClusterRole in the global namespace.
roleRef.apiGroup	Indicates the group for the resource being referenced.
roleRef.kind	Indicates the type of resource being referenced.
roleRef.name	Indicates the name of resource being referenced.
subjects	Manages references to the objects the role applies to.
subjects.kind	Indicates the type of object being referenced. Values defined by this API group are "User", "Group", and "ServiceAccount".
subjects.apiGroup	Manages the API group of the referenced subject.
subjects.name	Indicates the name of the object being referenced.

Debug Tool Configuration Parameters

Table 2-4 Debug Tool Configuration Parameters

Parameter	Description
command	Indicates the string array used for container command.
image	Indicates the docker image name.
imagePullPolicy	Indicates the Image Pull Policy.
name	Indicates the name of the container.
resources	Indicates the Compute Resources required by this container.
resources.limits	Describes the maximum amount of compute resources allowed.
resources.requests	Describes the minimum amount of compute resources required.
resources.limits.cpu	Indicates the CPU limits.
resources.limits.memory	Indicates the Memory limits.
resources.limits.ephemeral-storage	Indicates the Ephemeral Storage limits.
resources.requests.cpu	Indicates the CPU requests.
resources.requests.memory	Indicates the Memory requests.
resources.requests.ephemeral-storage	Indicates the Ephemeral Storage requests.
securityContext	Indicates the Security options the container should run with.
securityContext.allowPrivilegeEscalation	AllowPrivilegeEscalation controls whether a process can gain more privileges than its parent process. It directly controls whether the no_new_privs flag to be set on the container process.
secuirtyContext.readOnlyRootFilesystem	Indicates whether this container has a read-only root filesystem. The default value is false.

Table 2-4 (Cont.) Debug Tool Configuration Parameters

Parameter	Description
securityContext.capabilities	Indicates the capabilities to add or drop when running containers. It defaults to the default set of capabilities granted by the container runtime.
securityContext.capabilities.drop	Indicates the removed capabilities.
secuirtyContext.capabilities.add	Indicates the added capabilities.
securityContext.runAsUser	Indicates the UID to run the entrypoint of the container process.
extraContainersTpl.volumeMounts	Indicates that the parameter is used for mounting the volume.
extraContainersTpl.volumeMounts.mountPath	Indicates the path for volume mount.
extraContainersTpl.volumeMounts.name	Indicates the name of the directory for debug tool logs storage.
extraContainersVolumesTpl.name	Indicates the name of the volume for debug tool logs storage.
extraContainersVolumesTpl.emptyDir.medium	Indicates where the `emptyDir` volume is stored.
extraContainersVolumesTpl.emptyDir.sizeLimit	Indicates the `emptyDir` volume size.

2.7.4 Tools Tested in Debug Container

The following tables describe the list of debug tools that are tested.

Table 2-5 tcpdump

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

Table 2-5 (Cont.) tcpdump

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

Table 2-6 ip

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

Table 2-6 (Cont.) ip

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

Table 2-7 netstat

Options Tested	Description	Output	Capabilities
-a	Show both listening and non-listening sockets. For TCP, this means established connections.	`netstat -a` Active Internet connections (servers and established)Proto Recv-Q Send-Q Local Address Foreign Address Statetcp 0 0 0.0.0.0:tproxy 0.0.0.0:* LISTENtcp 0 0 0.0.0.0:websm 0.0.0.0:* LISTENtcp 0 0 cncc-core-ingress:websm 10-178-254-194.ku:47292 TIME_WAITtcp 0 0 cncc-core-ingress:46519 kubernetes.defaul:https ESTABLISHEDtcp 0 0 cncc-core-ingress:websm 10-178-254-194.ku:47240 TIME_WAITtcp 0 0 cncc-core-ingress:websm 10-178-254-194.ku:47347 TIME_WAITudp 0 0 localhost:59351 localhost:ambit-lm ESTABLISHEDActive UNIX domain sockets (servers and established)Proto RefCnt Flags Type State I-Node Pathunix 2 [ ] STREAM CONNECTED 576064861	-
-l	Show only listening sockets.	`netstat -l` Active Internet connections (only servers)Proto Recv-Q Send-Q Local Address Foreign Address Statetcp 0 0 0.0.0.0:tproxy 0.0.0.0:* LISTENtcp 0 0 0.0.0.0:websm 0.0.0.0:* LISTENActive UNIX domain sockets (only servers)Proto RefCnt Flags Type State I-Node Path	-
-s	Display summary statistics for each protocol.	`netstat -s` Ip:4070 total packets received0 forwarded0 incoming packets discarded4070 incoming packets delivered4315 requests sent outIcmp:0 ICMP messages received0 input ICMP message failed.ICMP input histogram:2 ICMP messages sent0 ICMP messages failedICMP output histogram:destination unreachable: 2	-
-i	Display a table of all network interfaces.	`netstat -i` Kernel Interface tableIface MTU RX-OK RX-ERR RX-DRP RX-OVR TX-OK TX-ERR TX-DRP TX-OVR Flgeth0 1440 4131 0 0 0 4355 0 0 0 BMRUlo 65536 0 0 0 0 0 0 0 0 LRU	-

Table 2-8 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`	-
--http2	Use the specified HTTP/2	`curl --http2 -v http://cncc-iam-ingress-gateway.cncc.svc.cluster.local:30085/cncc/auth/admin`	-

Table 2-9 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

Table 2-10 nmap

Options Tested	Description	Output	Capabilities

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

Table 2-10 (Cont.) nmap

Options Tested	Description	Output	Capabilities

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

Table 2-10 (Cont.) nmap

Options Tested	Description	Output	Capabilities

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

Table 2-11 dig

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

2.8 Logs

This chapter explains the process to retrieve the logs and status that can be used for effective troubleshooting. SCP provides various sources of information that may be helpful in the troubleshooting process.

Log Levels

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

Supported Log Levels

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

- DEBUG: A log level used for events considered to be useful during software debugging when more granular information is required.
- INFO: The standard log level indicating that something happened, the application entered a certain state, and so on.
- WARN: Indicates that something unexpected happened in the application, a problem, or a situation that might disturb one of the processes. But that doesn't mean that the application has failed. The WARN level should be used in situations that are unexpected, but the code can continue the work.
- ERROR: The log level that should be used when the application reaches an issue preventing one or more functionalities from properly functioning.

Configuring Log Levels

To view logging configurations and update logging levels, use the Logging Config option on the Cloud Native Configuration Console. For more information, see "Configuring Global Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy User Guide*.

2.8.1 Collecting Logs

Perform this procedure to collect logs from pods or containers.

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

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

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

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

   Example:

   ```
   $ kubectl logs ocscp-scp-worker-xxxxx -n ocscp
   ```

3. Run the following command to store the log in a file:

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

   Example:

   ```
   $ kubectl logs ocscp-scp-worker-xxxxx -n ocscp > logs.txt
   ```

4. (Optional) Run the following command 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 ocscp-scp-worker-xxxxx -n ocscp -f --tail 100 > logs.txt
   ```

2.8.2 Understanding Logs

This section explains the logs required to look into, to handle different SCP debugging issues.

The log level attributes of SCP services are as follows:

- SCPC-Subscription
- SCPC-Notification
- SCP-Worker

Sample Logs

Sample log statement for SCPC-Subscription:


```
{"instant":
{"epochSecond":1614521111,"nanoOfSecond":908545000},"thread":"main","level":"I
NFO","loggerName":"com.oracle.cgbu.cne.scp.soothsayer.subscription.processor.S
ubscriptionDataConsumer","message":"{logMsg=Subscription consumer cycle
completed.Thread Will now sleep for given time, cycle=232,
threadSleepTimeInMs=100}","endOfBatch":false,"loggerFqcn":"org.apache.logging.
log4j.spi.AbstractLogger","threadId":30,"threadPriority":5,"messageTimestamp":
"21-06-07 12:38:56.784+0000","application":"ocscp","microservice":"scpc-
subscription","engVersion":"1.12.0","mktgVersion":"1.12.0.0.0","vendor":"oracl
e","namespace":"scpsvc","node":"master","pod":"ocscp-scpc-
subscription-7f5b7c8ccd-
z89wb","subsystem":"subscription","instanceType":"prod","processId":"1"}
```


Sample log statement for SCPC-Notification:


```
{"instant":
{"epochSecond":1623069485,"nanoOfSecond":496630558},"thread":"main","level":"I
NFO","loggerName":"com.oracle.cgbu.cne.scp.soothsayer.Process","message":"{log
Msg=Successfully processed notification, nfInstanceId=6faf1bbc-6e4a-2828-a507-
a14ef8e1bc5b,
nfType=NRF}","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.Abs
tractLogger","threadId":1,"threadPriority":5,"messageTimestamp":"21-06-07
12:38:05.496+0000","application":"ocscp","microservice":"scpc-
notification","engVersion":"1.12.0","mktgVersion":"1.12.0.0.0","vendor":"oracl
e","namespace":"scpsvc","node":"master","pod":"ocscp-scpc-
notification-76597b5b7-
wfmxb","subsystem":"notification","instanceType":"prod","processId":"1"}
```

Sample log statement for SCP-Worker


```
{"instant":\{"epochSecond":1623069702,"nanoOfSecond":672444454},"thread":"scp-
upstream-
worker-7","level":"WARN","loggerName":"com.oracle.cgbu.cne.scp.router.routelay
er.MsgRouteChain","message":"MsgRouteChain::sendRsp(): SCP generated
Response, Response Code = 503, body = {\"title\":\"Service
Unavailable\",\"status\":\"503\",\"detail\":\"Service Unavailable:: Service
Unavailable\"}, error category = Destination webclient Connection Failure,
ingress request host = ocscp-scp-worker.scpsvc.svc.cluster.local:8000,
ingress request path = /nnrf-nfm/v1/subscriptions, ingress 3gpp-sbi-target-
apiRoot = http://nrf1svc.scpsvc.svc.cluster.local:8080, egress request Uri =
http://nrf2svc.scpsvc.svc.cluster.local:8080/nnrf-nfm/v1/subscriptions,
egress request Destination =
nrf2svc.scpsvc.svc.cluster.local","endOfBatch":false,"loggerFqcn":"org.apache.
logging.log4j.spi.AbstractLogger","threadId":139,"threadPriority":5,"messageId
":"c49b3b48-afc1-4058-83e8-b719ee181ed8","messageTimestamp":"21-06-07
12:41:42.672+0000","application":"ocscp","microservice":"scp-
worker","engVersion":"17.17.17","mktgVersion":"1.12.0.0.0","vendor":"oracle","
namespace":"scpsvc","node":"master","pod":"ocscp-scp-
worker-9567767dc-7bqg9","subsystem":"router","instanceType":"prod","processId"
:"1"}
```


The following table describes different log attributes:

Table 2-12 Log Attribute Details

Log Attribute	Details	Sample Value	Data Type
instant	Epoch time **Note**: It is a group of two values such as epochSecond and nanoOfSecond.	{"epochSecond":1614521244,"nanoOfSecond":775702000}	Object
thread	Logging Thread Name	"pool-4-thread-1"	String
level	Log Level of the log printed	"ERROR"	String
loggerName	Class or Module which printed the log	"com.oracle.cgbu.cne.scp.soothsayer.audit.process.AuditMaster"	String
message	Message related to the log providing brief details **Note**: WARN log level indicates that SCP connection with NRF is established.	{logMsg=NRF health check did not complete successfully. Next health check will start in given interval, timeIntervalInSec=2}	String
endOfBatch	Log4j2 Internal	false	boolean
loggerFqcn	Log4j2 Internal	org.apache.logging.log4j.spi.AbstractLogger	String
threadId	Thread Id generated internally by Log4j2	31	Integer
threadPriority	Thread Priority set internally by Log4j2	5	Integer
messageTimestamp	Timestamp when log was printed	"21-06-07 12:41:15.277+0000"	String
application	Application name	"ocscp"	String
scpFqdn	SCP FQDN	ocscp-scp-worker.ocscp-thrust5-06.svc.thrust5	String
microservice	Name of the microservice	"scpc-audit"	String

Table 2-12 (Cont.) Log Attribute Details

Log Attribute	Details	Sample Value	Data Type
engVersion	Engineering version of the software	"1.12.0"	String
mktgVersion	Marketing version of the software	"1.12.0.0.0"	String
vendor	Vendor of the software	"oracle"	String
namespace	Namespace where application is deployed	"scpsvc"	String
node	Node where the pod resides	"master"	String
pod	Pod name of deployed application	"ocscp-scpc-audit-6c5ddb4c54-hf8kr"	String
subsystem	Subsystem inside microservice group	"audit"	String
instanceType	Instance type	"prod"	String
processId	Process ID internally assigned	1	Integer

2.9 Verifying the Availability of Multus Container Network Interface

Perform the following procedure to verify whether the Multus Container Network Interface (CNI) feature is enabled after SCP installation is complete.

> **Note**
>
> Ensure that Multus Container Network Interface configuration is completed as described in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

1. To verify whether the pod contains signaling network, go to Kubernetes cluster and run the following command:

   ```
   kubectl describe pod <pod-name> -n <namespace>
   ```

2. Check whether the pod output contains the "net1" interface.
   Sample pod output:

   ```
   k8s.v1.cni.cncf.io/network-status:
                   [{
                       "name": "",
                       "ips": [
                           "192.168.219.111"
                       ],
                       "default": true,
                       "dns": {}
                   },{
                       "name": "scpsvc/macvlan-siga",
                       "interface": "net1",
                       "ips": [
                           "192.68.3.78"
                       ],
                       "mac": "b6:41:f9:a6:c8:8e",
   ```

```
        "dns": {}
    }]
k8s.v1.cni.cncf.io/networks: [{ "name": "macvlan-siga"}]
```

2.10 Upgrade or Rollback Failure

When Service Communication Proxy (SCP) upgrade or rollback fails, perform the following procedure.

1. Check the pre or post upgrade or rollback hook logs in Kibana as applicable. Users can filter upgrade or rollback logs using the following filters:
   - For upgrade: lifeCycleEvent=9001 or 9011
   - For rollback: lifeCycleEvent=9002

   Sample Log:

   ```
   {"time_stamp":"2021-11-22
   10:28:11.820+0000","thread":"main","level":"INFO","logger":"com.oracle.cgbu
   .cne.scp.soothsayer.hooks.releases.ReleaseHelmHook_1_14_1","message":"{logM
   sg=Starting Pre-Upgrade hook Execution, lifeCycleEvent=9001 | Upgrade,
   sourceRelease=101400,
   targetRelease=101500}","loc":"com.oracle.cgbu.cne.scp.soothsayer.common.uti
   ls.EventSpecificLogger.submit(EventSpecificLogger.java:94)"
   ```

2. Check the pod logs in Kibana to analyze the cause of failure.
3. After detecting the cause of failure, do the following:
   - For upgrade failure:
     - If the cause of upgrade failure is database or network connectivity issue, contact your system administrator. When the issue is resolved, rerun the upgrade command.
     - If the cause of failure is not related to database or network connectivity issue and is observed during the preupgrade phase, do not perform rollback because SCP 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.
4. If the issue persists, contact My Oracle Support.

2.11 Error Messages for Mediation Rule Configuration

This section allows you to troubleshoot and resolve problems in Mediation Rules using Drools Rule Language (DRL).

The following types of errors are handled:

- Custom Errors
- Errors specific to Drool library

The custom errors generated by application are as follows:

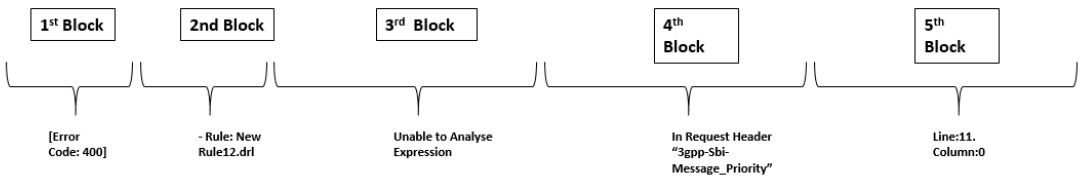
- Value not valid for type: This error is shown in cases where a field in the request body contains a value that is not expected.
- Fields are required and missing for rules with state: This error is shown in cases where a required field file is missing in the request.
- Rule name already exists in the database: When cloning a rule and the new name exists in the database, this error appears.
- Rule cannot be deleted on status: An APPLIED rule cannot be deleted. In the case of a delete request with APPLIED status, this exception will appear.
- Request body ruleName does not match with param ruleName: If the value of the rule name in the URL does not match the value in the "name" field in the request body, this exception will appear.
- Rule was not found: This error is shown when a get or delete request has a rule name that doesn't exist in the database.
- The State field is not valid at the creation of the rule; it must be SAVE state: The rule is not going to be created if the value of state in the request body is different from SAVE state.
- State transition is not valid for status: For more information, refer "Mediation Rule API Migration" section in the *Oracle Communications Cloud Native Core, Service Communication Proxy User Guide*.
- The APPLIED Status is not valid at the creation of the rule, it must be in the DRAFT status.

> **Note**
>
> Once a rule is generated, ruleName, Format, and Status fields cannot be changed. Some fields are dependent on Status fields to be able to modify. For more information, refer "Mediation Rule API Migration" section in the *Oracle Communications Cloud Native Core, Service Communication Proxy User Guide*.

For DRL errors, the Drools provide standardized messages in the following format:

- **1st Block:** Indicates the error code.
- **2nd Block:** Indicates the DRL source line and column where the problem occurred.
- **3rd Block:** Indicates the description of the problem.
- **4th Block:** Indicates the Components in the DRL source where the issue occurred, such as rules, functions, and queries.
- **5th Block:** Indicates the DRL source pattern where the issue occurred.

Figure 2-4 Error Message Format



The following standardized error messages are supported by Drools:

- **101: no viable alternative**

  Indicates that the parser reached a decision point but was unable to find an alternative.

  Example rule with incorrect spelling

```
//incorrect spelling
//incorrect syntax
package com.oracle.cgbu.ocmediation.nfmediation;
import com.oracle.cgbu.ocmediation.factdetails.Request;

dialect "mvel"

rule "ruleTest"
when
Â req : Request(headers.has("header") == true) //special character
then
 req.headers.add("newHeader","132465")
end
```

  > **Note**
  >
  > The parser reported an invalid token due to a special character.

- **102: mismatched input**

  Indicates that the parser expects a specific input that is not present at the current input point.

  Example rule with an incomplete rule statement

```
//incomplete rule statement
// incorrect syntax
package com.oracle.cgbu.ocmediation.nfmediation;
import com.oracle.cgbu.ocmediation.factdetails.Request;

dialect "mvel"

//Must have a rule name

when
 req : Request(headers.has("header1") == true)
then
 req.headers.add("TEST","132465")
end
```

  > **Note**
  >
  > The parser encountered an incomplete construct due to a missing rule name.

- **103: failed predicate**

Indicates that a validating semantic predicate is evaluated as false. These semantic predicates are commonly used in DRL files to identify component terms such as declare, rule, exists, not, and others.

Example rule with an invalid keyword

```
package com.oracle.cgbu.ocmediation.nfmediation;
import com.oracle.cgbu.ocmediation.factdetails.Request;

dialect "mvel"

fdsfdsfds

rule "ruleTest"
when
 req : Request(headers.has("header") == true)
then
 req.headers.add("newHeader","132465")
end
```

> **Note**
>
> This text line is not a DRL keyword construct, hence the parser cannot validate the rest of the DRL file.

- **105: did not match anything**

  Indicates the parser reached a grammar sub-rule that must match an alternative at least once but did not match anything.

  Example rule with invalid text in an empty condition

```
package com.oracle.cgbu.ocmediation.nfmediation;
import com.oracle.cgbu.ocmediation.factdetails.Request;

dialect "mvel"

rule "ruleTest"
when
 None // Must remove `None` if condition is empty
then
 req.headers.add("TEST","132465")
end
```

> **Note**
>
> The condition should be empty, but the word `None` is used. This error is resolved by removing `None`, which is not a valid DRL keyword.

For more information on the list of errors specific to the Drool library, see *Drools User Guide*.

2.12 Errors Observed on Grafana and OCI Dashboards

This section provides information to troubleshoot errors observed on Grafana and OCI dashboards. These errors occur when the number of records fetched by a query, which is an expression that uses application metrics and dimensions, exceeds the configured limit, due to which charts or widgets are invisible on dashboards.

The following sample error messages are displayed on the:

- Grafana dashboard: `execution: execution: query processing would load too many samples into memory in query execution`.
- OCI dashboard: `Query cannot result in more than 2000 streams`

To resolve this issue, perform the following tasks to view the data on these dashboards:

1. On the Grafana or OCI dashboard, minimize the query interval to check if the data appears.
   a. To debug an issue, check the error logs and select the interval on the dashboard based on the timestamp of the observed error logs to minimize the search results.

   This task fetches only records within that query interval and reduces the number of records on the dashboard.
2. You can add more filters using the metric dimensions (as per the traffic being run) to the query to minimize the search results. For more information about metric dimensions, see *Oracle Communications Cloud Native Core, Service Communication Proxy User Guide*. To query a particular metric, these dimensions act as filter keys as defined in Table 2-13 and Table 2-14.
3. Ensure that the metric used in the query is pegged by the SCP application.

The following tables describe examples of queries on Grafana and OCI dashboards:

Table 2-13 Examples of Query on Grafana Dashboard

Metric Used	Default Query on Dashboard	Query after Applying Additional Filters
ocscp_metric_http_rx_req_total	sum(irate(ocscp_metric_http_rx_req_total{namespace="scpsvc"}[2m])) by (pod)	sum(irate(ocscp_metric_http_rx_req_total{namespace="scpsvc",**ocscp_nf_service_type="nausf-auth"**}[2m])) by (pod)
ocscp_metric_http_tx_req_total	sum(irate(ocscp_metric_http_tx_req_total{namespace="ocscp"}[5m])) by (ocscp_nf_type,ocscp_producer_service_instance_id)	sum(irate(ocscp_metric_http_rx_req_total{namespace="ocscp",,**ocscp_nf_service_type="nausf-auth"**}[5m])) by (ocscp_nf_type,ocscp_producer_service_instance_id)

Table 2-14 Examples of Query on OCI Dashboard

Metric Used	Default Query on Dashboard	Query after Applying Additional Filters
ocscp_metric_http_rx_req_total	ocscp_metric_http_rx_req_total[10m]{k8Namespace="ocscp-2"}.rate().groupBy(podname).sum()	• ocscp_metric_http_rx_req_total[10m]{k8Namespace="ocscp-2",**ocscp_nf_service_type="nausf-auth"**}.rate().groupBy(podname).sum() • ocscp_metric_http_rx_req_total[10m]{k8Namespace="ocscp-2",**ocscp_nf_service_type="nausf-auth", ocscp_consumer_info="smf#smf8svc.ocscp-2.svc.cluster.local#NA**"}.rate().groupBy(podname).sum()
ocscp_metric_http_tx_res_total	ocscp_metric_http_tx_res_total[10m]{k8Namespace="ocscp-2"}.rate().groupBy(ocscp_consumer_info).sum()	• ocscp_metric_http_tx_res_total[10m]{k8Namespace="ocscp-2",**ocscp_nf_service_type="nausf-auth"**}.rate().groupBy(ocscp_consumer_info).sum() • ocscp_metric_http_tx_res_total[10m]{k8Namespace="ocscp-2",**ocscp_nf_service_type="nausf-auth", ocscp_consumer_info="smf#smf8svc.ocscp-2.svc.cluster.local#NA**"}.rate().groupBy(ocscp_consumer_info).sum()

3 Alerts

This section provides information about the supported alerts and how to configure the alerts.

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

You can configure alerts in Prometheus and ScpAlertrules.yaml file.

The following table provides information about Service Communication Proxy (SCP) alerts.

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

> **Note**
>
> - `kubectl` commands might vary based on the platform deployment. Replace `kubectl` with Kubernetes environment-specific command line tool to configure Kubernetes resources through kube-api server. The instructions provided in this document are as per the Oracle Communications Cloud Native Environment (OCCNE) version of kube-api server.
> - The alert file can be customized as required by the deployment environment. For example, namespace can be added as a filtered criteria to the alert expression to filter alerts only for a specific namespace.

3.1 System level alerts

This section lists the system level alerts.

3.1.1 SCPNotificationPodMemoryUsage

Table 3-1 SCPNotificationPodMemoryUsage

Field	Description
Severity	Major

Table 3-1 (Cont.) SCPNotificationPodMemoryUsage

Field	Description
Conditions	sum(container_memory_usage_bytes{image!="",pod=~".*scpc-notification.+"}) by (pod,namespace, instance) > 3006477107
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3001
Description	Notify Notification service Pod memory usage if it is above threshold Threshold value is 85% of allocated (4GB) memory: **3.4 GB**
Recommended Actions	**Cause**: When high notification rate or very large NF profile size is present in notifications. **Diagnostic Information**: Monitor the notification metric: ocscp_nrf_notifications_requests_nf_total. Notification usage reduces after some time when it crosses 2.5 GB or 3 GB. **Recovery**: This alert is cleared automatically when the scpc-notification pod memory usage reduces below the defined threshold. Reduce the notification rate. These notifications are generated by NRF and can be controlled through NRF. For any assistance, contact My Oracle Support.

3.1.2 SCPWorkerPodMemoryUsage

Table 3-2 SCPWorkerPodMemoryUsage

Field	Description
Severity	major
Conditions	sum(container_memory_usage_bytes{image!="",pod=~".*scp-worker.+"}) by (pod,namespace, instance) > 6012954214
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7004
Description	Notify Worker per Pod memory usage is above threshold Threshold value is 85% of allocated (8GB) memory: **7.3 GB**
Recommended Actions	**Cause**: When there is high traffic rate, alternate routing, more number of routing rules and rules size, and due to network or producer NF latency. **Diagnostic Information**: Monitor traffic rate, alerts, and latency on the KPI Dashboard. Check the traffic rates of the following metrics if they are too high: • ocscp_metric_http_rx_req_total • ocscp_metric_http_tx_req_total • ocscp_metric_http_rx_res_total • ocscp_metric_http_tx_res_total Check the upstream response time by using the following command and ensure whether upstream is taking too long to respond: ocscp_metric_upstream_service_time_total. Check the following platform metric for current memory usage by the scp-worker pod: container_memory_usage_bytes. **Recovery**: This alert is cleared automatically when the scp-worker pod memory usage reduces below the defined threshold. Reduce the traffic rate and improve the latency. For any assistance, contact My Oracle Support.

3.1.3 SCPInstanceDown

Table 3-3 SCPInstanceDown

Field	Description
Severity	Critical
Conditions	kube_pod_status_ready{pod =~ '.*scp-worker.*|.*scpc-notification.*|.*scpc-subscription.*|.*scpc-configuration.*|.*scpc-audit.*|,|.*scpc-alternate-resolution.*|,condition =~ 'true'} !=1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7006
Description	Notify that if any pod in ocscp release is down. Provides information like pod name, instance id and app name.
Recommended Actions	**Cause**: When the following issues occur: • The Control plane pods, such as configuration, subscription, notification, audit, and alternate-resolution are down due to connection failure with DB. • Pod restarts due to kubernetes liveliness or readiness probe failures. • Application restarts or starts failure. **Diagnostic Information**: • Check if DB services are active by running the following command: `kubectl describe pod <podname> -n <namespace>` • Check kubernetes events for probe failures in the platform logs. • Check if any exception is reported in the SCP application logs. **Recovery**: This alert is cleared automatically when the inactive pod is active. Recover DB services if down. Collect the application logs and contact My Oracle Support for any assistance.

3.2 Application level alerts

This section lists the application level alerts.

3.2.1 SCPCcaFeatureEnabledWithoutHttps

Table 3-4 SCPCcaFeatureEnabledWithoutHttps

Field	Description
Severity	Info
Condition	ocscp_worker_cca_validation_feature_enabled_without_https > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.9022
Description	An alert is raised when the CCA validation feature is enabled without enabling HTTPS.

Table 3-4 (Cont.) SCPCcaFeatureEnabledWithoutHttps

Field	Description
Recommended Actions	**Cause**: CCA validation feature is enabled without enabling HTTPS. **Diagnostic Information**: Deploy HTTPS SCP deployment. **Recovery**: The alert is cleared automatically if either the CCA feature is disabled or deployment is changed to HTTPS. For any assistance, contact My Oracle Support.

3.2.2 SCPIngressTrafficRateAboveMinorThreshold

Table 3-5 SCPIngressTrafficRateAboveMinorThreshold

Field	Description
Severity	minor
Condition	sum(rate(ocscp_metric_http_rx_req_total{app_kubernetes_io_name="scp-worker"}[2m]))by (kubernetes_namespace,ocscp_locality,kubernetes_pod_name)>= 1200 to 1400
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7001
Description	Notify that Traffic rate is from 1200 to 1400 mps (user configure minor threshold value) with Locality and current value of traffic rate.
Recommended Actions	**Cause**: When the Consumer NF sends more traffic than expected. **Diagnostic Information**: Monitor the ingress traffic to pod using the KPI Dashboard. Refer to the rate of ocscp_metric_http_rx_req_total metric on the Grafana GUI. **Recovery**: This alert is cleared automatically when the ingress traffic reduces below the minor threshold or exceeds the major threshold. If this alert is not cleared, then check the Consumer NF for an uneven distribution of traffic per connection or for any other issue. For any assistance, contact My Oracle Support.

3.2.3 SCPIngressTrafficRateAboveMajorThreshold

Table 3-6 SCPIngressTrafficRateAboveMajorThreshold

Field	Description
Severity	major
Conditions	sum(rate(ocscp_metric_http_rx_req_total{app_kubernetes_io_name="scp-worker"}[2m]))by (kubernetes_namespace,ocscp_locality,kubernetes_pod_name) >= 1400 to 1600
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7001
Description	Notify that Traffic rate is from 1400 to 1600 mps (user configure major threshold value) with Locality and current value of traffic rate.

Table 3-6 (Cont.) SCPIngressTrafficRateAboveMajorThreshold

Field	Description
Recommended Actions	**Cause**: When the Consumer NF sends more traffic than expected. **Diagnostic Information**: Monitor the ingress traffic to pod using the KPI Dashboard. Refer to the rate of ocscp_metric_http_rx_req_total metric on the Grafana GUI. **Recovery**: This alert is cleared automatically when the ingress traffic reduces below the major threshold or exceeds the critical threshold. If this alert is not cleared, then check the Consumer NF for an uneven distribution of traffic per connection or for any other issue. If this alert continues for a long duration, then reduce the ingress traffic from consumer to pod. For any assistance, contact My Oracle Support.

3.2.4 SCPIngressTrafficRateAboveCriticalThreshold

Table 3-7 SCPIngressTrafficRateAboveCriticalThreshold

Field	Description
Severity	critical
Conditions	sum(rate(ocscp_metric_http_rx_req_total{app_kubernetes_io_name="scp-worker"} [2m]))by(kubernetes_namespace,ocscp_locality,kubernetes_pod_name) >= 1600
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7001
Description	Notify that Traffic rate is above 1600mps (user configure critical threshold value) with Locality and current value of traffic rate.
Recommended Actions	**Cause**: When the Consumer NF sends more traffic than expected. **Diagnostic Information**: Monitor the ingress traffic to pod using the KPI Dashboard. Refer to the rate of the ocscp_metric_http_rx_req_total metric on the Grafana GUI. **Recovery**: This alert is cleared automatically when the ingress traffic reduces below the critical threshold. If this alert is not cleared, then check the Consumer NF for an uneven distribution of traffic per connection or for any other issue. If this alert continues for a long duration, then reduce the ingress traffic from consumer to pod. For any assistance, contact My Oracle Support.

3.2.5 SCPRoutingFailedForProducer

Table 3-8 SCPRoutingFailedForProducer

Field	Description
Severity	Info

Table 3-8 (Cont.) SCPRoutingFailedForProducer

Field	Description
Conditions	increase(ocscp_metric_routing_attempt_fail_total{app_kubernetes_io_name="scp-worker"}[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7005
Description	Notify that Routing failed for producer. Provides detail such as NFService Type, NFType, Locality, producer FQDN and value.
Recommended Actions	**Cause**: When routing fails to select a producer NF due to unavailability of routing rules for an NF service or producer. **Diagnostic Information**: • Check whether the routing rules are configured for the NF for which routing failed. • Check the notification logs for any error while processing the notification of the NF for which routing failed. Then, run the following command to get the notification logs: `kubectl logs <podname> -n <namespace>` • Check whether the NF is reachable or not by using one of the following steps: – Run the ping command from primary/secondary nodes using IP of service. Example of a ping command: `ping <IPAddress>`. – Run the ping command from inside the pod, if FQDN of service is used. – Instead of using the ping command, you can collect tcpdump for ensuring the connectivity. tcpdump must be run on the debug container for scp-worker microservice. Example of a tcpdump: `tcpdump -w capture.pcap -i <pod interface>`. **Recovery**: This alert is cleared automatically when the routing is complete for a producer NF or no more traffic is received in the next Promethues scrape interval. Check if the NF is deregistered. Register the NF to create routing rules if rules do not exist. For any assistance, contact My Oracle Support.

3.2.6 SCPAuditErrorResponse

Table 3-9 SCPAuditErrorResponse

Field	Description
Severity	Info
Conditions	ocscp_audit_error_response > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.4001
Description	Alert is raised when Audit module receives a 3xx, 4xx, or 5xx error from NRF. This alert is labeled with specific nftype, nrfRegionOrSetId, and auditmethod. **Note**: Alert is cleared on the next audit cycle.

Table 3-9 (Cont.) SCPAuditErrorResponse

Field	Description
Recommended Actions	**Cause**: When the configured NRF sends error responses, down, or not reachable. **Diagnostic Information**: • Check if NRF is up and reachable. To check the NRF status, see *Oracle Communications Cloud Native Core, Network Repository Function User Guide*. • Check if the NF is reachable or not using one of the following steps: – Run the ping command from primary/secondary nodes using IP of NRF. Example of a ping command: `ping <IpAddress>`. – Run the ping command from inside the pod if FQDN of NRF is used. – Instead of using ping, you can collect tcpdump for ensuring connectivity. Example of a tcpdump: `tcpdump -w capture.pcap -i <pod interface>`. • Monitor audit and worker service logs: `kubectl logs <pod name> -n <namespace>` • Check jaeger traces for scp-worker. **Recovery**: The alert is cleared automatically during the next audit cycle and when no more errors are received. Collect audit and worker service logs and contact My Oracle Support for any assistance.

3.2.7 SCPAuditEmptyNFArrayResponse

Table 3-10 SCPAuditEmptyNFArrayResponse

Field	Description
Severity	Critical
Conditions	ocscp_audit_2xx_empty_nf_array_rx_total > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.4002
Description	Alert is generated when Audit module receives a 2xx response with empty NFInstance array from NRF. Alert is labeled with specific nftype, nrfRegionOrSetId, and auditmethod. Alert is cleared if Audit receives a success response with non-empty NFInstance array or on next audit cycle if topology source is changed to LOCAL.
Recommended Actions	**Cause**: When NRF does not have any NF registered or due to any error condition on NRF. **Diagnostic Information**: Check if NRF contains any registered NF and validate as required. For more information, refer to NRF documents. **Recovery**: This alert is cleared automatically if Audit receives a success response with non-empty NFInstance array or during the next audit cycle when the topology source is changed to LOCAL. Register a NF with NRF or change the topology source to LOCAL. For any assistance, contact My Oracle Support.

3.2.8 DuplicateLocalityFoundInForeignNF

Table 3-11 DuplicateLocalityFoundInForeignNF

Field	Description
Severity	Major
Conditions	ocscp_notification_duplicate_foreign_location > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3002
Description	Alert is raised when an unknown NF or SCP is registered with duplicate locality from the present region.
Recommended Actions	**Cause**: When SCP discovers a duplicate locality of an NF from an unknown region. **Diagnostic Information**: Check logs for NF notification received by running the following command: `kubectl -n <namespace> logs <pod name>`. Check the following metric to get the NFInstanceId information for which this alert is raised: ocscp_notification_duplicate_foreign_location (nfInstanceId). From the metric, get the NF Instance ID, Locality, and serving_scope. Check the NF Profile of the corresponding NF in the unknown region as identified by the serving_scope. Check and correct the locality in the NF Profile to ensure it aligns with localities of that unknown region that should be different from locality of SCP which reported this alert. **Recovery**: This alert is cleared automatically if an unknown NF or SCP is deregistered or registers update with the correct locality. Re-register NF with correct locality information. Collect logs for notification and audit service. For any assistance, contact My Oracle Support.

3.2.9 ForeignNFLocalityNotServed

Table 3-12 ForeignNFLocalityNotServed

Field	Description
Severity	Critical
Conditions	ocscp_notification_foreign_nf_locality_unserved > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3003
Description	Alert is raised when a Foreign Producer NF's locality is not served by any SCP.

Table 3-12 (Cont.) ForeignNFLocalityNotServed

Field	Description
Recommended Actions	**Cause**: When SCP discovers an unknown Producer NF's without any locality served by an SCP. **Diagnostic Information**: Check logs for received NF notification by running the following command:`kubectl get pods -n <namespace>`. **Note**: Use the complete name of notification pod in the following command:`kubectl logs <pod> -n <namespace> .` Check the following metric to get the NFInstanceId information for which this alert is raised: ocscp_notification_foreign_nf_locality_unserved (nfInstanceId). **Recovery**: This alert is cleared automatically if the unknown NF is deregistered or registers update received with locality served by SCP. Re-register NF with correct locality information. For any assistance, contact My Oracle Support.

3.2.10 UnknownLocalityFoundInForeignNF

Table 3-13 UnknownLocalityFoundInForeignNF

Field	Description
Severity	critical
Conditions	ocscp_notification_foreign_nf_locality_absent > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3004
Description	Alert will be raised when a Foreign Producer NF's locality is unknown.
Recommended Actions	**Cause**: When SCP discovers an unknown Producer NF's without locality information. **Diagnostic Information**: Check logs for the received NF notification by running the following command: `kubectl get pods -n <namespace>`. Use the complete name of notification pod in the following command:`kubectl logs <pod> -n <namespace> -f --tail=0` Check the following metric to get the NFInstanceId information for which this alert is raised: ocscp_notification_foreign_nf_locality_absent(nfInstanceId). **Recovery**: This alert is cleared automatically if unknown NF is deregistered or registers update received with locality known to SCP. Re-register NF with correct locality information. For any assistance, contact My Oracle Support.

3.2.11 SCPUpstreamResponseTimeout

Table 3-14 SCPUpstreamResponseTimeout

Field	Description
Severity	info

Table 3-14 (Cont.) SCPUpstreamResponseTimeout

Field	Description
Conditions	idelta(ocscp_metric_upstream_response_timeout_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7011
Description	Alert is raised when Upstream connection to a producer NF fails
Recommended Actions	**Cause**: When a producer NF is down, not reachable, or latency is high. **Diagnostic Information**: Check whether the producer NF is up and network connectivity to the producer NF is established by using one of the following steps: • Run the ping command from primary/secondary nodes by using IP of producer NF. Example of a ping command: `ping <IPAddress>`. • Run the ping command from inside the pod, if FQDN of producer NF is used. • Instead of using the ping command, you can collect tcpdump for ensuring the connectivity. tcpdump must be run on the debug container for scp-worker microservice. Example of a tcpdump: `tcpdump -w capture.pcap -i <pod interface>`. Check the upstream response time by using the following metric and determine if upstream is taking too long to respond: ocscp_metric_upstream_service_time (producer FQDN) **Recovery**: This alert is cleared automatically in the next scrape interval if the system does not observe any error. For any assistance, contact My Oracle Support.

3.2.12 SCPSingleNfInstanceAvailableForNFType

Table 3-15 SCPSingleNfInstanceAvailableForNFType

Field	Description
Severity	Major
Conditions	ocscp_no_nf_instance == 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3005
Description	Alert is raised when there is a single NFInstance available with SCP for an NFType.
Recommended Actions	**Cause**: When the `preventiveAuditOnLastNFInstanceDeletion` attribute is set to true, SCP has single NFInstance available for an NFType. **Diagnostic Information**: Check all SCP NRFs for specific NFType in the alert if only one NFInstance is available. For information about registered NFs, see *Oracle Communications Cloud Native Core, Network Repository Function User Guide*. Check the number of NFs of a particular type by using API or CNC console of SCP. For information about procedures to check the NFs available with SCP, see " Configuring Service Communication Proxy using the CNC Console" in *Oracle Communications Cloud Native Core, Service Communication Proxy User Guide*. **Recovery**: This alert is cleared automatically in the next scrape interval if more than one NFInstance is available for a specified NFType in the alert. For any assistance, contact My Oracle Support.

3.2.13 SCPNoNfInstanceForNFType

Table 3-16 SCPNoNfInstanceForNFType

Field	Description
Severity	Critical
Conditions	ocscp_no_nf_instance == 1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3006
Description	Alert is raised when there is a no NFInstance available with SCP for a NFType
Recommended Actions	**Cause**: When the `preventiveAuditOnLastNFInstanceDeletion` flag is set to true, SCP has no NFInstance available for a NFTyp. **Diagnostic Information**: Check all SCP NRFs for specific NFType in the alert if no NFInstance is available. For information about registered NFs, see *Oracle Communications Cloud Native Core, Network Repository Function User Guide*. Check the number of NFs of a particular type by using API or CNC console of SCP. For information about procedures to check the NFs available with SCP, see " Configuring Service Communication Proxy using the CNC Console" in *Oracle Communications Cloud Native Core, Service Communication Proxy User Guide*. **Recovery**: This alert is cleared automatically in the next scrape interval if at least one NFInstance is available for a specified NFType in the alert. For any assistance, contact My Oracle Support.

3.2.14 SCPIngressTrafficRateExceededConfiguredLimit

Table 3-17 SCPIngressTrafficRateExceededConfiguredLimit

Alert Parameters	Value
Description	Ingress traffic rate exceeds configured rate limit for consumer fqdn: {{$labels.ocscp_consumer_fqdn}}
Summary	'Ingress traffic rate exceeds configured rate limit for consumer fqdn: ocscpconsumerfqdn = {{$labels.ocscp_consumer_host}},consumernfinstanceid = {{$labels.ocscp_consumer_nf_instance_id}}, consumernftype = {{$labels.ocscp_consumer_nf_type}}, configuredingressrate = {{$labels.ocscp_configured_ingress_rate}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}},scp_fqdn: ' {{$labels.scp_fqdn}} ',timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} and value = {{ $value }} '
Severity	Critical
Condition	This alert is raised when the ingress traffic rate exceeds the configured rate for consumer FQDN. increase(ocscp_metric_ingress_rate_limiting_throttle_req_total[2m]) > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7012
Metric Used	ocscp_metric_ingress_rate_limiting_throttle_req_total

Table 3-17 (Cont.) SCPIngressTrafficRateExceededConfiguredLimit

Alert Parameters	Value
Recommended Actions	**Cause:** When the ingress traffic rate exceeds the configured rate limit for the consumer FQDN. **Diagnostic Information:** • Check the ingress traffic rate from the consumer FQDN. • To check the ingress rate, refer to the following metrics: cscp_metric_http_rx_req_total • Check the ingress rate limit configuration as described in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. **Recovery:** This alert is cleared when no more requests get suppressed due to ingress rate limiting in the next scrape interval. For any assistance, contact My Oracle Support.

3.2.15 SCPIngressTrafficRoutedWithoutRateLimitTreatment

Table 3-18 SCPIngressTrafficRoutedWithoutRateLimitTreatment

Alert Parameters	Value
Description	Ingress traffic routed without rate limit treatment
Summary	'Ingress traffic routed without rate limit treatment: consumernftype = {{$labels.ocscp_consumer_nf_type}},consumernfinstanceid = {{$labels.ocscp_consumer_nf_instance_id}}, consumerfqdn = {{$labels.ocscp_consumer_host}}, cause = {{$labels.ocscp_cause}} ,namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}},scp_fqdn: ' {{$labels.scp_fqdn}} ', timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} and value = {{ $value }} '
Severity	Major
Condition	This alert is raised when the ingress traffic routes without rate limiting treatment. increase(ocscp_metric_ingress_rate_limiting_not_applied_req_total[2m]) > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7013
Metric Used	ocscp_metric_ingress_rate_limiting_not_applied_req_total
Recommended Actions	**Cause:** When the ingress traffic routes without rate limiting treatment. **Diagnostic Information:** • Check the ingress rate limiting configurations for the untreated FQDNs that can be obtained from the following metric: ocscp_metric_ingress_rate_limiting_not_applied_req_total(ocscp_consumer_fqdn) • Check the ingress rate limit configuration as described in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. **Recovery:** This alert is cleared when no more requests get routed without ingress rate limiting treatment in the next scrape interval. For any assistance, contact My Oracle Support.

3.2.16 SCPEgressTrafficRateExceededConfiguredLimit

Table 3-19 SCPEgressTrafficRateExceededConfiguredLimit

Field	Description
Summary	'Egress traffic rate exceeds configured rate limit: producernftype = {{$labels.ocscp_nf_type}}, producernfservicetype = {{$labels.ocscp_nf_service_type}}, producernfinstanceid = {{$labels.ocscp_nf_instance_id}}, producerfqdn = {{$labels.ocscp_producer_host}}, consumernftype = {{$labels.ocscp_consumer_nf_type}},consumernfinstanceid = {{$labels.ocscp_consumer_nf_instance_id}}, consumerfqdn = {{$labels.ocscp_consumer_host}}, configuredegressrate = {{$labels.ocscp_configured_egress_rate}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}},scp_fqdn: ' {{$labels.scp_fqdn}} ', timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} and value = {{ $value }} '
Severity	Critical
Conditions	idelta(ocscp_metric_egress_rate_limiting_throttle_req_total{app_kubernetes_io_name="scp-worker"}[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7014
Description	Alert is raised when the egress traffic rate exceed the configured rate.
Recommended Actions	**Cause**: When the egress traffic rate exceeds the configured rate. **Diagnostic Information**: Check the egress traffic rate by using the following metric: ocscp_metric_http_tx_req_total. Check the egress rate limit configuration as described in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. **Recovery**: This alert is cleared when no more requests get suppressed due to egress rate limiting in the next scrape interval. For any assistance, contact My Oracle Support.

3.2.17 SCPEgressTrafficRoutedWithoutRateLimitTreatment

Table 3-20 SCPEgressTrafficRoutedWithoutRateLimitTreatment

Field	Description
Severity	Major
Conditions	idelta(ocscp_metric_egress_rate_limiting_not_applied_req_total{app_kubernetes_io_name="scp-worker"}[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7015
Description	Alert is raised when egress traffic routes without rate limiting.

Table 3-20 (Cont.) SCPEgressTrafficRoutedWithoutRateLimitTreatment

Field	Description
Recommended Actions	**Cause**: When the egress traffic routes without rate limiting treatment. **Diagnostic Information**: Check the egress rate limiting configurations for the untreated producer FQDN. Obtain the producer FQDN by using the following metric: ocscp_metric_egress_rate_limiting_not_applied_req_total(ocscp_producer_fqdn) Check the egress rate limit configuration as described in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. **Recovery**: This alert is cleared when no more requests get routed without egress rate limiting treatment in the next scrape interval. For any assistance, contact My Oracle Support.

3.2.18 SCPNotificatoinRejectTopologySourceLocal

Table 3-21 SCPNotificatoinRejectTopologySourceLocal

Field	Description
Severity	Info
Conditions	increase(ocscp_notifications_rejected_topologysource_local_total[15m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3007
Description	Alert is raised when SCP rejects a notification from NRF due to topology source set to LOCAL for NF Type.
Recommended Actions	**Cause**: When NF Topology Source Info is set to LOCAL. **Diagnostic Information**: Check the topology source information of an NF Type. For information about the topology source APIs, see *Oracle Communications Cloud Native Core, Service Communication Proxy User Guide*. **Recovery**: This alert is cleared automatically after 15 minutes when NF Topology Source Info is set to NRF from LOCAL. For any assistance, contact My Oracle Support.

3.2.19 SCPNotificationProcessingFailureForNF

Table 3-22 SCPNotificationProcessingFailureForNF

Field	Description
Severity	Major
Conditions	increase(ocscp_failure_processed_nf_notification_total[15m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.3008
Description	Alerts is raised when Notification processing has failed on SCP.

Table 3-22 (Cont.) SCPNotificationProcessingFailureForNF

Field	Description
Recommended Actions	**Cause**: When Notification processing has failed on SCP. **Diagnostic Information**: Check notification pod logs for any errors by running the following command: `kubectl logs <notification pod name> -n <scp namespace>` . To get the list of pods, run the following command: `kubectl get pod -n <scp namespace>` Sample logs: `{"instant": {"epochSecond":1620272241,"nanoOfSecond":609935406},"thread":"runQueueThreadPool1","level":"ERROR","loggerName":"com.oracle.cgbu.cne.scp.soothsayer.Process","message":"{logMsg=Notified profile IP endpoints already present in stored profile, logMsgCode=DUPLICATE_IPENDPOINT_OR_FQDN_FOUND_IN_STORED_PROFILE, rootCause=Notified Profile contains an ipEndPoint {\"ipv4Address\":\"10.75.203.74\",\"transport\":\"TCP\",\"port\":32673} which is already present in stored Profile with nfInstanceId 93ED74AA-A29C-4450-9D7A-9278CAF6266D for serviceInstanceId audmcl08nv08-udmueauthn-589d6d5bcc-4dslh}","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.AbstractLogger","threadId":34,"threadPriority":5,"messageTimestamp":"21-05-06 03:37:21.609+0000","application":"ocscp-soothsayer","microservice":"ocscp-scpc-notification","engVersion":"24.2.6","mktgVersion":"24.2.6.0.0","vendor":"oracle","namespace":"scpsvc","node":"slave1","pod":"ocscp-scpc-notification-547f699c96-m7nc8","subsystem":"notification","instanceType":"prod","processId":"1"}` `{"instant": {"epochSecond":1620272241,"nanoOfSecond":734873890},"thread":"runQueueThreadPool1","level":"WARN","loggerName":"com.oracle.cgbu.cne.scp.soothsayer.scheduler.RunQueueConsumer","message":"{logMsg=Profile Processing failed, nfInstanceId=93ED74AA-A29C-4450-9D7A-9278CAF6266D}","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.AbstractLogger","threadId":34,"threadPriority":5,"messageTimestamp":"21-05-06 03:37:21.734+0000","application":"ocscp-soothsayer","microservice":"ocscp-scpc-`

Table 3-22 (Cont.) SCPNotificationProcessingFailureForNF

Field	Description
	`notification","engVersion":"24.2.6","mktgVersion":"24.2.6.0.0","vendor":"oracle","namespace":"scpsvc","node":"slave1","pod":"ocscp-scpc-notification-547f699c96-m7nc8","subsystem":"notification","instanceType":"prod","processId":"1"}` For information about the topology source APIs, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. **Recovery**: This alert is cleared automatically after 15 minutes. For any assistance, contact My Oracle Support.

3.2.20 SCPSubscriptionFailureForNFType

Table 3-23 SCPSubscriptionFailureForNFType

Field	Description
Severity	Critical
Conditions	increase(ocscp_subscription_nf_failure_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.2001
Description	Alerts is raised when SCP subscription to NRF has failed. This alert is labeled with specific nftype, nrfRegionOrSetId, and auditmethod.

Table 3-23 (Cont.) SCPSubscriptionFailureForNFType

Field	Description
Recommended Actions	**Cause**: When the subscription fails for an NF Type with NRF. **Diagnostic Information**: Check whether NRF is up. To check the NRF status, see the *Oracle Communications Cloud Native Core, Network Repository Function User Guide*. Check whether the NRF is reachable or not by using one of the following steps: • Run the ping command from primary or secondary nodes using IP of NRF. Example of a ping command: `ping <IPAddress>`. • Run the ping command from inside the pod if FQDN of NRF is used. • Instead of using ping, you can collect tcpdump for ensuring the connectivity. Example of a tcpdump: `tcpdump -w capture.pcap -i <pod interface>`. If NRF is up, check scp-worker logs to find any error response from NRF. If there are error responses, monitor NRF logs: kubectl logs <pod name> -n <namespace>. Sample logs: `{"instant": {"epochSecond":1620275506,"nanoOfSecond":134773910},"thread":"pool-8-thread-1","level":"ERROR","loggerName":"com.oracle.cgbu.cne.scp.soothsayer.subscription.processor.SubscriptionDataConsumer","message":"{logMsg=Exception occurred while handling action for subscriber data, action=TO_BE_RENEWED, stackTrace=com.oracle.cgbu.cne.scp.soothsayer.subscription.operations.NrfSubscriptionClient.triggerPatchRequest(NrfSubscriptionClient.java:177)\ncom.oracle.cgbu.cne.scp.soothsayer.subscription.processor.SubscriptionDataConsumer.handleAction(SubscriptionDataConsumer.java:322)\ncom.oracle.cgbu.cne.scp.soothsayer.subscription.processor.SubscriptionDataConsumer.consumeSQ(SubscriptionDataConsumer.java:140)\ncom.oracle.cgbu.cne.scp.soothsayer.subscription.processor.SubscriptionDataConsumer.run(SubscriptionDataConsumer.java:84)\njava.base/java.util.concurrent.ThreadPoolExecutor.runWorker(ThreadPoolExecutor.java:1130)\njava.base/java.util.concurrent.ThreadPoolExecutor$Worker.run(ThreadPoolExecutor.java:630)\njava.base/java.lang.Thread.run(Thread.java:832)}","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.AbstractLogger","threadId":33,"threadPriority":5,"messageTimestamp":"21-05-06 04:31:46.134+0000","application":"ocscp-soothsayer","microservice":"ocscp-scpc-subscription","engVersion":"1.15.0","mktgVersion":"1.15.0.0.0","vendor":"oracle","namespace":"scpsvc","node":"master","pod":"ocscp-scpc-`

Table 3-23 (Cont.) SCPSubscriptionFailureForNFType

Field	Description
	`subscription-55cfb57cc6-2qp2g","subsystem":"subscription","instanceType":"prod","processId":"1"}` **Recovery**: This alert is cleared automatically when NRF is up and running or errors are corrected for received error responses. For any assistance, contact My Oracle Support.

3.2.21 SCPReSubscriptionFailureForNFType

Table 3-24 SCPReSubscriptionFailureForNFType

Field	Description
Severity	Critical
Conditions	increase(ocscp_patch_subscription_nf_failure_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.2002
Description	Alerts is raised when SCP re-subscription to NRF has failed. This alert is labeled with specific nftype, nrfRegionOrSetId, and auditmethod.
Recommended Actions	**Cause**: When the re-subscription fails for an NF Type with NRF. **Diagnostic Information**: Check whether NRF is up. To check the NRF status, see the *Oracle Communications Cloud Native Core, Network Repository Function User Guide*. Check whether the NRF is reachable or not by using one of the following steps: • Run the ping command from primary/secondary nodes by using IP of NRF. Example of a ping command: `ping <IPAddress>`. • Run the ping command from inside the pod, if FQDN of NRF is used. • Instead of using ping, you can collect tcpdump for ensuring the connectivity. Example of a tcpdump: `tcpdump -w capture.pcap -i <pod interface>`. Check scp-worker logs to find any error response from NRF. If there are error responses, monitor NRF logs: kubectl logs <pod name> -n <namespace>. **Recovery**: This alert is cleared automatically when NRF is up and running or errors are corrected for received error responses. For any assistance, contact My Oracle Support.

3.2.22 SCPNrfRegistrationFailureForRegionOrSetId

Table 3-25 SCPNrfRegistrationFailureForRegionOrSetId

Field	Description
Severity	Major
Conditions	increase(ocscp_nrf_registration_failure_total[2m]) > 0

Table 3-25 (Cont.) SCPNrfRegistrationFailureForRegionOrSetId

Field	Description
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.2003
Description	Alerts is raised when SCP registration fails. This alert is labeled with specific nftype, nrfRegionOrSetId, and auditmethod.
Recommended Actions	**Cause**: When the registration fails for an NF Type with NRF. **Diagnostic Information**: Check whether NRF is up. To check the NRF status, see the *Oracle Communications Cloud Native Core, Network Repository Function User Guide*. Check whether the NRF is reachable or not by using one of the following steps: • Run the ping command from primary or secondary nodes by using IP of NRF. Example of a ping command: `ping <IPAddress>`. • Run the ping command from inside the pod if FQDN of NRF is used. • Instead of using ping, you can collect tcpdump for ensuring the connectivity. Example of a tcpdump: `tcpdump -w capture.pcap -i <pod interface>`. Check scp-worker logs to find any error response from NRF. If there are error responses, monitor NRF logs: kubectl logs <pod name> -n <namespace>. Sample logs: `{"instant": {"epochSecond":1620638888,"nanoOfSecond":78728229},"thread":"registration-0","level":"ERROR","loggerName": "com.oracle.cgbu.cne.scp.soothsayer.subscription.processor.NrfRegistrationProcessor","message":"{logMsg=Registration will be retried after configured interval, configuredIntervalInSec=6}","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.AbstractLogger", "threadId":33,"threadPriority":5,"messageTimestamp":"21-05-10 09:28:08.078+0000","application":"ocscp-soothsayer","microservice":"ocscp-scpc-subscription","engVersion":"24.2.6","mktgVersion":"24.2.6.0.0","vendor":"oracle","namespace":"scpsvc","node":"slave1","pod":"ocscp-scpc-subscription-66c68b9db6-6g582","subsystem":"subscription","instanceType":"prod","processId":"1"}` **Recovery**: This alert is cleared automatically when NRF is up and running or errors are corrected for received error responses. For any assistance, contact My Oracle Support.

3.2.23 SCPNrfHeartbeatFailureForRegionOrSetId

Table 3-26 SCPNrfHeartbeatFailureForRegionOrSetId

Field	Description
Severity	Major
Conditions	increase(ocscp_nrf_heartbeat_failures_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.2004
Description	Alerts is raised when SCP Heartbeat fails.
Recommended Actions	**Cause**: When the Heartbeat fails for an NF Type with NRF. **Diagnostic Information**: Check whether NRF is up. To check the NRF status, see the *Oracle Communications Cloud Native Core, Network Repository Function User Guide*. Check whether the NRF is reachable or not by using one of the following steps: • Run the ping command from primaryor secondary nodes by using IP of NRF. Example of a ping command: `ping <IPAddress>`. • Run the ping command from inside the pod if FQDN of NRF is used. • Instead of using ping, you can collect tcpdump for ensuring the connectivity. Example of a tcpdump: `tcpdump -w capture.pcap -i <pod interface>`. Check scp-worker logs to find any error response from NRF. If there are error responses, monitor NRF logs: kubectl logs <pod name> -n <namespace>. **Recovery**: This alert is cleared automatically when NRF is up and running or errors are corrected for received error responses. For any assistance, contact My Oracle Support.

3.2.24 SCPDBOperationFailure

Table 3-27 SCPDBOperationFailure

Field	Description
Severity	Warning
Conditions	increase(ocscp_db_operation_failure_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.2005
Description	Alert is raised for any DB operation failures.
Recommended Actions	**Cause**: When the SCP DB operation fails. **Diagnostic Information**: Check whether the DB service is up. Check the status/age of the mysql pod by using the following command: `kubectl get pods -n <namespace>`. Where, <namespace> is the namespace in which mysql pod is deployed. **Recovery**: This alert is cleared automatically when the DB service is up and running. For any assistance, contact My Oracle Support.

3.2.25 SCPGeneratedErrorsResponseForNFService

Table 3-28 SCPGeneratedErrorsResponseForNFService

Field	Description
Severity	Info
Conditions	increase(ocscp_metric_scp_generated_response_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7016
Description	Alert is raised for NF type for which SCP generated response is triggered.
Recommended Actions	**Cause**: When the error response is generated for NF Service Type by SCP. **Diagnostic Information**: Monitor scp-worker logs to determine the reason for error responses generated by SCP. Check for error reason in the logs: kubectl logs <pod name> -n <namespace>. **Recovery**: This alert is cleared automatically when the cause for error response at SCP worker is corrected and configured. For any assistance, contact My Oracle Support.

3.2.26 SCPCircuitBreakingAppliedForNF

Table 3-29 SCPCircuitBreakingAppliedForNF

Field	Description
Severity	Info
Conditions	ocscp_circuit_breaking_applied > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.7017
Description	Alert is raised for NF when circuit breaking is applied.
Recommended Actions	**Cause**: When Circuit Breaking applies for producer NF FQDN based on the configured http2MaxRequests value. **Diagnostic Information**: Monitor scp-worker logs for number of error responses when outstanding requests exceed the configured http2MaxRequests value: kubectl logs <pod name> -n <namespace>. Check the latency to upstream producer from SCP. Use the following metric to check the same: ocscp_metric_upstream_service_time_total(ocscp_producer_host or ocscp_nf_end_point). **Recovery**: This alert is cleared automatically when the configuration for http2MaxRequests for circuit breaking is configured beyond the traffic at worker or lower the traffic than the value configured for circuit breaking. For any assistance, contact My Oracle Support.

3.2.27 SCPUpgradeStarted

Table 3-30 SCPUpgradeStarted

Field	Description
Severity	Info
Conditions	When the SCP upgrade process for an SCP microservice starts.
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.6001
Description	Alert is raised when the SCP upgrade process for an SCP microservice starts.
Recommended Actions	**Cause**: When SCP upgrade is performed for a particular microservice. **Diagnostic Information**: Not applicable. **Recovery**: This alert is cleared automatically in 5 minutes when the `customAlertExpiryEnabled` parameter is set to `false` in the `ocscp_values.yaml` file. Otherwise, it is cleared after a specific duration as specified in the `customAlertExpiryDuration` parameter when the `customAlertExpiryEnabled` value is `true`. For any assistance, contact My Oracle Support.

3.2.28 SCPUpgradeFailed

Table 3-31 SCPUpgradeFailed

Field	Description
Severity	Critical
Conditions	When any SCP microservice upgrade fails during the upgrade process.
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.6002
Description	Alert is raised when any SCP microservice upgrade fails.
Recommended Actions	**Cause**: When any SCP microservice upgrade fails during the upgrade process. **Diagnostic Information**: Monitor new hook-jobs that might have failed after multiple attempts. Also, monitor any failed log. Run the following command to check the pod of hook-job: `kubectl get pods -n <namespace>`. Run the following command to check the logs: `kubectl logs <pod name> -n <namespace>`. **Recovery**: This alert is cleared automatically in 5 minutes when the `customAlertExpiryEnabled` parameter is set to `false` in the `ocscp_values.yaml` file. Otherwise, it is cleared after a specific duration as specified in the `customAlertExpiryDuration` parameter when the `customAlertExpiryEnabled` value is `true`. For any assistance, contact My Oracle Support.

3.2.29 SCPUpgradeSuccessful

Table 3-32 SCPUpgradeSuccessful

Field	Description
Severity	Info
Conditions	When any SCP microservice upgrade is completed.
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.6003
Description	Alert is raised when any SCP microservice upgrade is completed.
Recommended Actions	**Cause**: When any SCP microservice upgrade is completed. **Diagnostic Information**: Not applicable. Run the following command to check the pod of hook-job: `kubectl get pods -n <namespace>`. Run the following command to check the logs: `kubectl logs <pod name> -n <namespace>`. **Recovery**: This alert is cleared automatically in 5 minutes when the `customAlertExpiryEnabled` parameter is set to `false` in the `ocscp_values.yaml` file. Otherwise, it is cleared after a specific duration as specified in the `customAlertExpiryDuration` parameter when the `customAlertExpiryEnabled` value is `true`. For any assistance, contact My Oracle Support.

3.2.30 SCPRollbackStarted

Table 3-33 SCPRollbackStarted

Field	Description
Severity	Info
Conditions	When the rollback process for an SCP microservice starts.
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.6004
Description	Alert is raised when the rollback process for an SCP microservice starts.
Recommended Actions	**Cause**: When the rollback process for an SCP microservice starts. **Diagnostic Information**: Not applicable. **Recovery**: This alert is cleared automatically in 5 minutes when the `customAlertExpiryEnabled` parameter is set to `false` in the `ocscp_values.yaml` file. Otherwise, it is cleared after a specific duration as specified in the `customAlertExpiryDuration` parameter when the `customAlertExpiryEnabled` value is `true`. For any assistance, contact My Oracle Support.

3.2.31 SCPRollbackFailed

Table 3-34 SCPRollbackFailed

Field	Description
Severity	Critical

Table 3-34 (Cont.) SCPRollbackFailed

Field	Description
Conditions	When any SCP microservice rollback fails during the rollback process.
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.6005
Description	Alert is raised when any SCP microservice rollback fails.

Recommended Actions	**Cause**: When any SCP microservice rollback fails during the rollback process.
	Diagnostic Information: Monitor new hook-jobs that might have failed after multiple attempts. Also, monitor any failed log.
	Run the following command to check the pod of hook-job:
	`kubectl get pods -n <namespace>`
	Run the following command to check the logs:
	`kubectl logs <pod name> -n <namespace>`
	Recovery: This alert is cleared automatically in 5 minutes when the `customAlertExpiryEnabled` parameter is set to `false` in the `ocscp_values.yaml` file. Otherwise, it is cleared after a specific duration as specified in the `customAlertExpiryDuration` parameter when the `customAlertExpiryEnabled` value is `true`.
	For any assistance, contact My Oracle Support.

3.2.32 SCPRollbackSuccessful

Table 3-35 SCPRollbackSuccessful

Field	Description
Severity	Info
Conditions	When any SCP microservice rollback is completed.
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.6006
Description	Alert is raised when any SCP microservice rollback is completed.
Recommended Actions	**Cause**: When any SCP microservice rollback is completed. **Diagnostic Information**: Not applicable. **Recovery**: This alert is cleared automatically in 5 minutes when the `customAlertExpiryEnabled` parameter is set to `false` in the `ocscp_values.yaml` file. Otherwise, it is cleared after a specific duration as specified in the `customAlertExpiryDuration` parameter when the `customAlertExpiryEnabled` value is `true`. For any assistance, contact My Oracle Support.

3.2.33 ScpWorkerPodCpuUtilizationAboveWarnThreshold

Table 3-36 ScpWorkerPodCpuUtilizationAboveWarnThreshold

Field	Details
Description	CPU utilization of SCP worker at warn level
Summary	CPU utilization of SCP worker at warn level. namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} and value = {{ $value }}
Severity	Warning
Condition	This alert is raised when CPU utilization of SCP-Worker reaches the WARN level. ocscp_worker_pod_overload_control_cpu_utilization_warn > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7018
Metric Used	ocscp_worker_pod_overload_control_cpu_utilization_warn
Recommended Action	**Cause**: When CPU utilization of scp-worker reaches the WARN level. **Diagnostic Information**: 1. Get the configured threshold level values using Pod Overload Control Policy REST APIs. For more information about Pod Overload Control Policy, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. 2. Check the CPU threshold level status from the scp-worker pod logs under the warn level. **Recovery**: • Reduce the incoming service request rate. • This alert is automatically cleared when CPU utilization is reduced to below WARN threshold level. For any assistance, contact My Oracle Support.

3.2.34 ScpWorkerPodCpuUtilizationAboveMinorThreshold

Table 3-37 ScpWorkerPodCpuUtilizationAboveMinorThreshold

Field	Details
Description	CPU utilization of SCP worker at minor level
Summary	CPU utilization of SCP worker at minor level. namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} and value = {{ $value }}
Severity	Minor
Condition	This alert is raised when CPU utilization of scp-worker reaches the MINOR level. ocscp_worker_pod_overload_control_cpu_utilization_minor > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7019
Metric Used	ocscp_worker_pod_overload_control_cpu_utilization_minor

Table 3-37 (Cont.) ScpWorkerPodCpuUtilizationAboveMinorThreshold

Field	Details
Recommended Action	**Cause**: When CPU utilization of SCP-Worker reaches the MINOR level. **Diagnostic Information**: 1. Get the configured threshold level values using Pod Overload Control Policy REST APIs. For more information about Pod Overload Control Policy, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. 2. Check the CPU threshold level status from the scp-worker pod logs under the warn level. **Recovery**: • Reduce the incoming service request rate. • This alert is automatically cleared when CPU utilization is reduced to below MINOR threshold level. For any assistance, contact My Oracle Support.

3.2.35 ScpWorkerPodCpuUtilizationAboveMajorThreshold

Table 3-38 ScpWorkerPodCpuUtilizationAboveMajorThreshold

Field	Details
Description	CPU utilization of SCP worker at major level
Summary	CPU utilization of SCP worker at major level. namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} and value = {{ $value }}
Severity	Major
Condition	This alert is raised when CPU utilization of scp-worker reaches the MAJOR level. ocscp_worker_pod_overload_control_cpu_utilization_major > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7020
Metric Used	ocscp_worker_pod_overload_control_cpu_utilization_major
Recommended Action	**Cause**: When CPU utilization of SCP-Worker reaches the MAJOR level. **Diagnostic Information**: 1. Get the configured threshold level values using Pod Overload Control Policy REST APIs. For more information about Pod Overload Control Policy, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. 2. Check the CPU threshold level status from the scp-worker pod logs under the warn level. **Recovery**: • Reduce the incoming service request rate. • This alert is automatically cleared when CPU utilization is reduced to below MAJOR threshold level. For any assistance, contact My Oracle Support.

3.2.36 ScpWorkerPodCpuUtilizationAboveCriticalThreshold

Table 3-39 ScpWorkerPodCpuUtilizationAboveCriticalThreshold

Field	Details
Description	CPU utilization of SCP worker at critical level
Summary	CPU utilization of SCP worker at critical level. namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} and value = {{ $value }}
Severity	Critical
Condition	This alert is raised when CPU utilization of scp-worker reaches the CRITICAL level. ocscp_worker_pod_overload_control_cpu_utilization_critical > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7021
Metric Used	ocscp_worker_pod_overload_control_cpu_utilization_critical
Recommended Action	**Cause**: When CPU utilization of scp-worker reaches the CRITICAL level. **Diagnostic Information**: 1. Get the configured threshold level values using Pod Overload Control Policy REST APIs. For more information about Pod Overload Control Policy, see *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. 2. Check the CPU threshold level status from the scp-worker pod logs under the warn level. **Recovery**: • Reduce the incoming service request rate. • This alert is automatically cleared when CPU utilization is reduced to below CRITICAL threshold level. For any assistance, contact My Oracle Support.

3.2.37 SCPUnhealthyPeerSCPDetected

Table 3-40 SCPUnhealthyPeerSCPDetected

Field	Details
Description	Next hop SCP is marked unhealthy
Summary	'Next hop SCP is marked unhealthy. peerScpFqdn: {{labels.peerScpName}}, scpFqdn: {{labels.scpFqdn}} , namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} and value = {{ $value }} '
Severity	Info
Condition	This alert is raised when the peer SCP is marked as unhealthy. ocscp_peer_scp_unhealthy > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7022
Metric Used	ocscp_peer_scp_unhealthy

Table 3-40 (Cont.) SCPUnhealthyPeerSCPDetected

Field	Details
Recommended Action	**Cause**: The peer SCP is marked as unhealthy because of consecutive failure responses. **Diagnostic Information**: 1. Check transport failures and routing errors on the peer SCP. 2. Run the ping command from master or slave nodes using IP of Service. Sample command: `ping <IPAddress>`. 3. If FQDN of service is used, run the ping command from inside the pod. 4. If the pod does not support the ping command, get the debug container of SCP pod. 5. If you do not want to use the ping command, collect `tcpdump` to establish the connection. Sample command: `tcpdump -w capture.pcap -i <pod interface>` **Recovery**: This alert is automatically cleared after the degradation time is over. Degradation time = Number of consecutive degradations multiplied by configured base ejection. For any assistance, contact My Oracle Support.

3.2.38 SCPDnsSrvQueryFailure

Table 3-41 SCPDnsSrvQueryFailure

Field	Details
Description	DNS SRV Query failed with cause {{$labels.cause}}
Summary	'DNS SRV Query failed with cause {{$labels.cause}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when the DNS server lookup for SRV fails due to network, servfail, or timed-out errors. ocscp_alternate_resolution_dnssrv_rx_error_res == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.8001
Metric Used	ocscp_alternate_resolution_dnssrv_rx_error_res
Recommended Action	**Cause**: When the dnsSRVAlternateRouting flag is set to true, if the DNS SRV lookup fails due to network, servfail, or timed-out errors. **Diagnostic Information**: Check the DNS SRV server status and re-establish the status to normal. **Recovery**: This alert is automatically cleared when SCP performs a successful DNS SRV query. For any assistance, contact My Oracle Support.

3.2.39 SCPProducerOverloadThrottled

Table 3-42 SCPProducerOverloadThrottled

Field	Details
Description	Producer is in Throttled Overload state
Summary	'Producer is in Throttled Overload state. producerFqdn: {{$labels.producerFqdn}}, scpfqdn: {{$labels.scp_fqdn}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}
Severity	Info
Condition	This alert is raised when the producer NF is in the throttled congestion state. ocscp_load_manager_peer_load_throttled_threshold == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.7023
Metric Used	ocscp_load_manager_peer_load_throttled_threshold
Recommended Action	**Cause**: When the load of producer NF is higher than the throttled threshold configured for the service. **Diagnostic Information**: 1. Check and configure the throttled threshold for each service using Routing Options REST APIs configurations as described in "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. 2. Check the load for each service using the NF Profile REST APIs as described in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. **Recovery**: This alert clears automatically when the NF profile is deregistered or changed with load less than the throttled abatement threshold. For any assistance, contact My Oracle Support.

3.2.40 SCPProducerOverloadAlternateRouted

Table 3-43 SCPProducerOverloadAlternateRouted

Field	Details
Description	Producer is in Alternate Route Overload state
Summary	'Producer is in Alternate Route Overload state. producerFqdn: {{$labels.producerFqdn}}, scpfqdn: {{$labels.scp_fqdn}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}
Severity	Info
Condition	This alert is raised when the producer NF is in the alternate routing congestion state. ocscp_load_manager_peer_load_alternateRoute_threshold == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.7024

Table 3-43 (Cont.) SCPProducerOverloadAlternateRouted

Field	Details
Metric Used	ocscp_load_manager_peer_load_alternateRoute_threshold
Recommended Action	**Cause**: When the load of producer NF is higher than alternate routing threshold configured for the service. **Diagnostic Information**: 1. Check and configure the alternate routing threshold for each service using Routing Options REST APIs configurations as described in "Configuring Routing Options" in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. 2. Check the load for each service using the NF Profile REST APIs as described in *Oracle Communications Cloud Native Core, Service Communication Proxy REST Specification Guide*. **Recovery**: This alert clears automatically when the NF profile is deregistered or changed with load less than the alternate routing abatement threshold. For any assistance, contact My Oracle Support.

3.2.41 SCPSeppNotConfigured

Table 3-44 SCPSeppNotConfigured

Field	Details
Description	SEPP is not configured for PLMN
Summary	'SEPP is not configured for PLMN'Summary: 'SEPP is not configured for PLMN. plmnid: {{$labels.plmn_id}}, scpfqdn: {{$labels.scp_fqdn}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when Security Edge Protection Proxy (SEPP) is not configured. ocscp_metric_sepp_not_configured_current == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.7025
Metric Used	cscp_metric_sepp_not_configured_current
Recommended Action	**Cause**: When SEPP routing related rules are not configured at SCP for selected PLMN in the inter-PLMN routing. **Diagnostic Information**: 1. Check whether SEPP profile is registered or SEPP related configuration is made at SCP. 2. Verify the routing rules created for selected inter-PLMN in the PLMN_SEPP_MAPPING table. **Recovery**: This alert clears automatically when the SEPP related routing rules are created at SCP for selected PLMN in the inter-PLMN routing. For any assistance, contact My Oracle Support.

3.2.42 SCPSeppRoutingFailed

Table 3-45 SCPSeppRoutingFailed

Field	Details
Description	Routing towards SEPP failed
Summary	Routing towards SEPP failed. sepp_fqdn: {{$labels.ocscp_sepp_fqdn}}, scpfqdn: {{$labels.scp_fqdn}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when routing towards SEPP fails. ocscp_metric_sepp_routing_attempt_fail_current == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.7026
Metric Used	ocscp_metric_sepp_routing_attempt_fail_current
Recommended Action	**Cause**: Inter-PLMN routing failed for the selected SEPP instances. **Diagnostic Information**: 1. Check whether selected SEPP is up and healthy. 2. Check if selected SEPP is reachable. 3. Use the ping command from primary or secondary nodes using IP of SEPP. Sample ping command is `ping <IpAddress>` 4. If FQDN of SEPP is used, try ping command from inside the pod. 5. Alternative to ping, collect tcpdump for ensuring proper connectivity. Sample command: `tcpdump -w capture.pcap -i <pod interface>` 6. Check scp-worker logs for any error response from selected SEPP. If there are error responses, monitor selected SEPP logs by running this command: `kubectl logs <pod name> -n <namespace>` **Recovery**: This alert clears automatically when routing is successful for selected SEPP. For any assistance, contact My Oracle Support.

3.2.43 SCPGlobalEgressRLRemoteParticipantConnectivityFailure

Table 3-46 SCPGlobalEgressRLRemoteParticipantConnectivityFailure

Field	Details
Description	'SCP Global Egress RL Remote Participant Connectivity Failure for participant
Summary	'SCP Global Egress RL Remote Participant Connectivity Failure for participant: {{$labels.scp_remote_coh_cluster_name}}, scp_fqdn: {{$labels.scp_fqdn}}, scp_local_coh_cluster_name: {{$labels.scp_local_coh_cluster_name}}, scp_remote_coh_cluster_fqdn: {{$labels.scp_remote_coh_cluster_fqdn }}, scp_remote_coh_cluster_port: {{$labels.scp_remote_coh_cluster_port }}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Major

Table 3-46 (Cont.) SCPGlobalEgressRLRemoteParticipantConnectivityFailure

Field	Details
Condition	This alert is raised when the remote participant SCP connection is not established or goes down. ocscp_global_egress_rl_remote_participant_connectivity_failure == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9001
Metric Used	ocscp_global_egress_rl_bucketkey_not_rate_controlled_total
Recommended Action	**Cause**: When the remote participant SCP connection is not established or down. **Diagnostic Information**: 1. Check whether the FQDN or IP port are properly configured with remote SCP's scp-cache microservice. 2. Check whether clusterName and NFInstanceID are configured for the remote SCP. 3. Check whether communication path is active between the two SCP's scp-cache microservice. 4. Run the following command to monitor scp-cache logs: `kubectl logs <pod name> -n <namespace>` **Recovery**: This alert clears automatically if the connection is established with the remote participant SCP. For any assistance, contact My Oracle Support.

3.2.44 SCPGlobalEgressRLRemoteParticipantWithDuplicateNFInstanceId

Table 3-47 SCPGlobalEgressRLRemoteParticipantWithDuplicateNFInstanceId

Field	Details
Description	SCP global egress RL remote participant configured with duplicate NF InstanceId for participant.
Summary	'SCP Global Egress RL Remote Participant Configured With Duplicate NFInstanceId for participant: {{$labels.scp_remote_coh_cluster_name}}, scp_fqdn: {{$labels.scp_fqdn}}, scp_nf_instance_id: {{$labels.scp_nf_instance_id}}, scp_local_coh_cluster_name: {{$labels.scp_local_coh_cluster_name}}, scp_remote_coh_cluster_fqdn: {{$labels.scp_remote_coh_cluster_fqdn }}, scp_remote_coh_cluster_port: {{$labels.scp_remote_coh_cluster_port }}, namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }} {{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when a duplicate remote coherence participant is found. ocscp_global_egress_rl_remote_participant_is_duplicate == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9002
Metric Used	ocscp_global_egress_rl_remote_participant_is_duplicate

Table 3-47 (Cont.) SCPGlobalEgressRLRemoteParticipantWithDuplicateNFInstanceId

Field	Details
Recommended Action	**Cause**: Duplicate configuration of remote coherence participants with local SCP. **Diagnostic Information**: 1. Ensure Global Rate Limit feature is enabled. 2. Check whether the clusterName and NFInstanceID of local SCPs and remote SCPs are not duplicates. 3. Check whether the clusterName and NFInstanceID of the first remote SCP and the second remote SCP are not duplicates. 4. Configure unique values for clusterName and NFInstanceID for local SCPs as well as remote SCPs. **Recovery**: • This alert is cleared automatically if no duplicate configurations between local and remote SCPs are found. For any assistance, contact My Oracle Support.

3.2.45 SCPMediationConnectivityFailure

Table 3-48 SCPMediationConnectivityFailure

Field	Details
Description	'SCP Mediation Connectivity Failed, scp_fqdn
Summary	'SCP Mediation Connectivity Failed, scp_fqdn: {{$labels.scp_fqdn}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when Mediation connection is not established or request to Mediation is not successful. ocscp_mediation_http_not_reachable == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9002
Metric Used	ocscp_mediation_http_not_reachable

Table 3-48 (Cont.) SCPMediationConnectivityFailure

Field	Details

Recommended Action	**Cause**: The remote Mediation connection is not established or request to Mediation is not successful.
	Diagnostic Information:
	1. Check the errors of the ocscp_mediation_http_not_reachable metric on the Grafana dashboard.
	2. Run the following command to check the status of the mediation pod:
	`kubectl get pods -n <namespace>`
	Recovery:
	1. If the mediation pod is not in the ready state, run the following command to check the scp-mediation logs:
	`kubectl logs <pod name> -n <namespace>`
	2. If the mediation pod is absent, then redeploy or upgrade SCP with `mediationService` set to `true`.
	This alert clears automatically when the connection is established with Mediation when Mediation is invoked from any of the trigger points.
	For any assistance, contact My Oracle Support.

3.2.46 SCPNotificationQueuesUtilizationAboveMinorThreshold

Table 3-49 SCPNotificationQueuesUtilizationAboveMinorThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Notification Queues Utilization Above Minor Threshold'
Summary	'SCP Notification Queues Utilization Above Minor Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when the queues in the notification service are utilized above 65% of the maximum size (user configure minor threshold value). ocscp_notification_queue_alert{severity="MINOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.3009
Metric Used	ocscp_notification_queue_alert

Table 3-49 (Cont.) SCPNotificationQueuesUtilizationAboveMinorThreshold

Field	Details
Recommended Action	**Cause**: The Notification module is getting more traffic than expected. **Diagnostic Information**: 1. Monitor Notification traffic to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_notification_queue_utilization **Recovery**: This alert clears automatically when notification traffic goes below minor threshold or exceeds major threshold. For any assistance, contact My Oracle Support.

3.2.47 SCPNotificationQueuesUtilizationAboveMajorThreshold

Table 3-50 SCPNotificationQueuesUtilizationAboveMajorThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Notification Queues Utilization Above Major Threshold'
Summary	'SCP Notification Queues Utilization Above Major Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when the queues in the notification service is utilized above 75% of the maximum size (user configure major threshold value). ocscp_notification_queue_alert{severity="MAJOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.3010
Metric Used	ocscp_notification_queue_alert
Recommended Action	**Cause**: The Notification module is getting more traffic than expected. **Diagnostic Information**: 1. Monitor Notification traffic to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_notification_queue_utilization **Recovery**: This alert clears automatically when notification traffic goes below major threshold or above critical major threshold. For any assistance, contact My Oracle Support.

3.2.48 SCPNotificationQueuesUtilizationAboveCriticalThreshold

Table 3-51 SCPNotificationQueuesUtilizationAboveCriticalThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Notification Queues Utilization Above Critical Threshold'
Summary	'SCP Notification Queues Utilization Above Critical Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when the queues in notification service are utilized above 85% of the maximum size (user configure critical threshold value). ocscp_notification_queue_alert{severity="CRITICAL"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.3011
Metric Used	ocscp_notification_queue_alert
Recommended Action	**Cause**: The Notification module is getting more traffic than expected. **Diagnostic Information**: 1. Monitor Notification traffic to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_notification_queue_utilization **Recovery**: This alert clears automatically when notification traffic goes below critical threshold. For any assistance, contact My Oracle Support.

3.2.49 SCPNrfProxyQueuesUtilizationAboveMinorThreshold

Table 3-52 SCPNrfProxyQueuesUtilizationAboveMinorThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Nrfproxy Queues Utilization Above Minor Threshold'
Summary	'SCP Nrfproxy Queues Utilization Above Minor Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when the task queues in scp-nrfproxy service are utilized above 65% of the maximum size (user configure minor threshold value). ocscp_nrfproxy_queue_alert{severity="MINOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9010
Metric Used	ocscp_nrfproxy_queue_alert

Table 3-52 (Cont.) SCPNrfProxyQueuesUtilizationAboveMinorThreshold

Field	Details
Recommended Action	**Cause**: NrfProxy task queues are getting filled and the traffic is more than expected. **Diagnostic Information**: 1. Monitor traffic towards nrfProxy to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_nrfproxy_queue_utilization **Recovery**: This alert clears automatically when the traffic goes below minor threshold or above major threshold. For any assistance, contact My Oracle Support.

3.2.50 SCPNrfProxyQueuesUtilizationAboveMajorThreshold

Table 3-53 SCPNrfProxyQueuesUtilizationAboveMajorThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Nrfproxy Queues Utilization Above Major Threshold'
Summary	'SCP Nrfproxy Queues Utilization Above Major Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when the task queues in scp-nrfproxy service are utilized above 75% of the maximum size (user configure major threshold value). ocscp_nrfproxy_queue_alert{severity="MAJOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9011
Metric Used	ocscp_nrfproxy_queue_alert
Recommended Action	**Cause**: NrfProxy task queues are getting filled and the traffic is more than expected. **Diagnostic Information**: 1. Monitor traffic towards nrfProxy to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_nrfproxy_queue_utilization **Recovery**: This alert clears automatically when the traffic goes below major threshold or above critical threshold. For any assistance, contact My Oracle Support.

3.2.51 SCPNrfProxyQueuesUtilizationAboveCriticalThreshold

Table 3-54 SCPNrfProxyQueuesUtilizationAboveCriticalThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Nrfproxy Queues Utilization Above Critical Threshold'
Summary	'SCP Nrfproxy Queues Utilization Above Critical Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when the task queues in scp-nrfproxy service are utilized above 85% of the maximum size (user configure critical threshold value). ocscp_nrfproxy_queue_alert{severity="CRITICAL"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9012
Metric Used	ocscp_nrfproxy_queue_alert
Recommended Action	**Cause**: NrfProxy task queues are getting filled and the traffic is more than expected. **Diagnostic Information**: 1. Monitor traffic towards nrfProxy to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_nrfproxy_queue_utilization **Recovery**: This alert clears automatically when the traffic goes below critical threshold. For any assistance, contact My Oracle Support.

3.2.52 SCPWorkerQueuesUtilizationAboveMinorThreshold

Table 3-55 SCPWorkerQueuesUtilizationAboveMinorThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Worker Queues Utilization Above Minor Threshold'
Summary	'SCP Worker Queues Utilization Above Minor Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when task queues in scp-worker service are utilized above 65% of the maximum size (user configure minor threshold value). ocscp_worker_queue_alert{severity="MINOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9007
Metric Used	ocscp_worker_queue_alert

Table 3-55 (Cont.) SCPWorkerQueuesUtilizationAboveMinorThreshold

Field	Details
Recommended Action	**Cause**: Worker task queues are getting filled and the traffic is more than expected. **Diagnostic Information**: 1. Monitor traffic towards nrfProxy to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_worker_queue_utilization **Recovery**: This alert clears automatically when the traffic goes below minor threshold or above major threshold. For any assistance, contact My Oracle Support.

3.2.53 SCPWorkerQueuesUtilizationAboveMajorThreshold

Table 3-56 SCPWorkerQueuesUtilizationAboveMajorThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Worker Queues Utilization Above Major Threshold'
Summary	'SCP Worker Queues Utilization Above Major Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when task queues in scp-worker service are utilized above 75% of the maximum size (user configure major threshold value). ocscp_worker_queue_alert{severity="MAJOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9008
Metric Used	ocscp_worker_queue_alert
Recommended Action	**Cause**: Worker task queues are getting filled and the traffic is more than expected. **Diagnostic Information**: 1. Monitor traffic towards nrfProxy to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_worker_queue_utilization **Recovery**: This alert clears automatically when the traffic goes below major threshold or goes above critical threshold. For any assistance, contact My Oracle Support.

3.2.54 SCPWorkerQueuesUtilizationAboveCriticalThreshold

Table 3-57 SCPWorkerQueuesUtilizationAboveCriticalThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Worker Queues Utilization Above Critical Threshold'
Summary	'SCP Worker Queues Utilization Above Critical Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when task queues in scp-worker service are utilized above 85% of the maximum size (user configure critical threshold value). ocscp_worker_queue_alert{severity="CRITICAL"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9009
Metric Used	ocscp_worker_queue_alert
Recommended Action	**Cause**: Worker task queues are getting filled and the traffic is more than expected. **Diagnostic Information**: 1. Monitor traffic towards nrfProxy to pod using the KPI dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_worker_queue_utilization **Recovery**: This alert clears automatically when the traffic goes below critical threshold. For any assistance, contact My Oracle Support.

3.2.55 SCPCacheQueuesUtilizationAboveMinorThreshold

Table 3-58 SCPCacheQueuesUtilizationAboveMinorThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Cache Queues Utilization Above Minor Threshold'
Summary	'SCP Cache Queues Utilization Above Minor Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when the task queues in the scp-cache service are utilized above 65% of their maximum size (the user-configured minor threshold value). ocscp_cache_queue_alert{severity="MINOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.13002
Metric Used	ocscp_cache_queue_utilization

Table 3-58 (Cont.) SCPCacheQueuesUtilizationAboveMinorThreshold

Field	Details
Recommended Action	**Cause**: When the cache task queues are getting filled, and traffic is higher than expected. **Diagnostic Information**: 1. Monitor traffic towards the cache pod using the KPI Dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_cache_queue_utilization. **Recovery**: The alert is cleared automatically when minor threshold or goes above major threshold. For any assistance, contact My Oracle Support.

3.2.56 SCPCacheQueuesUtilizationAboveMajorThreshold

Table 3-59 SCPCacheQueuesUtilizationAboveMajorThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Cache Queues Utilization Above Major Threshold'
Summary	SCP Cache Queues Utilization Above Major Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when the task queues in the scp-cache service are utilized above 75% of their maximum size (the user-configured major threshold value). ocscp_cache_queue_alert{severity="MAJOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.13001
Metric Used	ocscp_cache_queue_utilization
Recommended Action	**Cause**: When the cache task queues are getting filled, and traffic is higher than expected. **Diagnostic Information**: 1. Monitor traffic towards the cache pod using the KPI Dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_cache_queue_utilization. **Recovery**: The alert is cleared automatically when traffic falls below a major threshold or goes above a critical threshold. For any assistance, contact My Oracle Support.

3.2.57 SCPCacheQueuesUtilizationAboveCriticalThreshold

Table 3-60 SCPCacheQueuesUtilizationAboveCriticalThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Cache Queues Utilization Above Critical Threshold'
Summary	'SCP Cache Queues Utilization Above Critical Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when the task queues in the scp-cache service are utilized above 85% of their maximum size (the user-configured critical threshold value). ocscp_cache_queue_alert{severity="CRITICAL"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.13000
Metric Used	ocscp_cache_queue_utilization
Recommended Action	**Cause**: When the cache task queues are getting filled, and traffic is higher than expected. **Diagnostic Information**: 1. Monitor traffic towards the cache pod using the KPI Dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_cache_queue_utilization. **Recovery**: The alert is cleared automatically when traffic falls below a critical threshold. For any assistance, contact My Oracle Support.

3.2.58 SCPLoadManagerQueuesUtilizationAboveMinorThreshold

Table 3-61 SCPLoadManagerQueuesUtilizationAboveMinorThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Load Manage Queues Utilization Above Minor Threshold'
Summary	'SCP Load Manager Queues Utilization Above Minor Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when the task queues in the scp-load-manager service are utilized above 65% of their maximum size (the user-configured minor threshold value). ocscp_load_manager_queue_alert{severity="MINOR"}
OID	1.3.6.1.4.1.323.5.3.35.1.2.11002
Metric Used	ocscp_load_manager_queue_utilization

Table 3-61 (Cont.) SCPLoadManagerQueuesUtilizationAboveMinorThreshold

Field	Details
Recommended Action	**Cause**: When the cache task queues are getting filled, and traffic is higher than expected. **Diagnostic Information**: 1. Monitor traffic towards the cache pod using the KPI Dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_load_manager_queue_utilization. **Recovery**: The alert is cleared automatically when traffic falls below a minor threshold. For any assistance, contact My Oracle Support.

3.2.59 SCPLoadManagerQueuesUtilizationAboveMajorThreshold

Table 3-62 SCPLoadManagerQueuesUtilizationAboveMajorThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Load Manage Queues Utilization Above Major Threshold'
Summary	'SCP Load Manager Queues Utilization Above Major Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when the task queues in the scp-load-manager service are utilized above 75% of their maximum size (the user-configured major threshold value). ocscp_load_manager_queue_alert{severity="MAJOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.11001
Metric Used	ocscp_load_manager_queue_utilization
Recommended Action	**Cause**: When the cache task queues are getting filled, and traffic is higher than expected. **Diagnostic Information**: 1. Monitor traffic towards the cache pod using the KPI Dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_load_manager_queue_utilization. **Recovery**: The alert is cleared automatically when traffic falls below a major threshold. For any assistance, contact My Oracle Support.

3.2.60 SCPLoadManagerQueuesUtilizationAboveCriticalThreshold

Table 3-63 SCPLoadManagerQueuesUtilizationAboveCriticalThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}: SCP Load Manage Queues Utilization Above Critical Threshold'
Summary	'SCP Load Manager Queues Utilization Above Critical Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when the task queues in the scp-load-manager service are utilized above 85% of their maximum size (the user-configured critical threshold value). ocscp_load_manager_queue_alert{severity="CRITICAL"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.11000
Metric Used	ocscp_load_manager_queue_utilization
Recommended Action	**Cause**: When the cache task queues are getting filled, and traffic is higher than expected. **Diagnostic Information**: 1. Monitor traffic towards the cache pod using the KPI Dashboard. 2. Refer to rate of the following metric on the Grafana dashboard: ocscp_load_manager_queue_utilization. **Recovery**: The alert is cleared automatically when traffic falls below a critical threshold. For any assistance, contact My Oracle Support.

3.2.61 SCPProducerNfSetUnhealthy

Table 3-64 SCPProducerNfSetUnhealthy

Field	Details
Description	All producer NFs in NF set are marked unhealthy
Summary	'All producer NFs in NF set are marked unhealthy. nfSet: {{$labels.ocscp_nf_setid}}, scpFqdn: {{$labels.scp_fqdn}} , namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Info
Condition	This alert is raised when all producer NFs in an NF Set are marked unhealthy. ocscp_metric_nf_set_unhealthy > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7027
Metric Used	ocscp_metric_nf_set_unhealthy

Table 3-64 (Cont.) SCPProducerNfSetUnhealthy

Field	Details
Recommended Action	**Cause**: All the producer NFs are marked unhealthy because of consecutive failure responses. **Diagnostic Information**: 1. Check transport failures and routing errors on producer NFs. 2. Run the ping command from primary or secondary nodes using IP of Service. Sample command: `ping <IPAddress>`. 3. If FQDN of service is used, run the ping command from inside the pod. 4. If the pod does not support the ping command, get the debug container of SCP pod. 5. If you do not want to use the ping command, collect `tcpdump` to establish the connection. Sample command: `tcpdump -w capture.pcap -i <pod interface>` **Recovery**: This alert is automatically cleared after the degradation time is over. Degradation time = Number of consecutive degradations multiplied by configured base ejection. For any assistance, contact My Oracle Support.

3.2.62 SCPPeerSeppUnhealthy

Table 3-65 SCPPeerSeppUnhealthy

Field	Details
Description	Peer Sepp is marked unhealthy
Summary	'Peer Sepp is marked unhealthy. seppFqdn: {{$labels.ocscp_sepp_fqdn}}, namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Info
Condition	This alert is raised when peer SEPP is marked unhealthy. ocscp_sepp_unhealthy > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7028
Metric Used	ocscp_sepp_unhealthy

Table 3-65 (Cont.) SCPPeerSeppUnhealthy

Field	Details
Recommended Action	**Cause**: The peer SEPP is marked unhealthy because of consecutive failure responses. **Diagnostic Information**: 1. Check transport failures and routing errors on peer SCP. 2. Run the ping command from primary or secondary nodes using IP of Service. Sample command: `ping <IPAddress>`. 3. If FQDN of service is used, run the ping command from inside the pod. 4. If the pod does not support the ping command, get the debug container of SCP pod. 5. If you do not want to use the ping command, collect `tcpdump` to establish the connection. Sample command: `tcpdump -w capture.pcap -i <pod interface>` **Recovery**: This alert is automatically cleared after the degradation time is over. Degradation time = Number of consecutive degradations multiplied by configured base ejection. For any assistance, contact My Oracle Support.

3.2.63 SCPMicroServiceUnreachable

Table 3-66 SCPMicroServiceUnreachable

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.namespace}}, podname: {{$labels.pod}}: SCP communication between the micro-services indicated by source and destination has failed'
Summary	Summary: 'SCP communication between the micro-services indicated by source and destination has failed: {{$labels.instance}}, namespace: {{$labels.namespace}}, source:{{$labels.source}}, destination: {{$labels.destination}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when the communication between SCP microservices indicated by source and destination has failed. ocscp_metric_svc_unreachable==1
OID	1.3.6.1.4.1.323.5.3.35.1.2.7029
Metric Used	ocscp_metric_svc_unreachable
Recommended Action	**Cause**: Communication between SCP microservices has failed. **Diagnostic Information**: Verify whether endpoints of all the services are in Running and Ready state. If not, restart the services. **Recovery**: This alert clears automatically when the required services are in Running and Ready state. For any assistance, contact My Oracle Support.

3.2.64 SCPTrafficFeedSendFailed

Table 3-67 SCPTrafficFeedSendFailed

Field	Details
Description	'Sending messages to Traffic Feed failed. Cause : {{$labels.ocscp_cause}}'
Summary	'Sending messages to Traffic Feed failed, cause: {{$labels.ocscp_cause}}, scp_fqdn: {{$labels.scp_fqdn}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when sending messages to traffic feed fails. increase(ocscp_metric_trafficfeed_attempted_total{app_kubernetes_io_name="scp-worker"}[1h]) > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9003
Metric Used	ocscp_metric_trafficfeed_attempted_total
Recommended Action	**Cause**: Sending of message to traffic feed failed. **Diagnostic Information**: • Check failure reason. • Check if the traffic feed OCNADD configuration is correct. • Check if the OCNADD server is reachable and available. **Recovery**: This alert clears automatically after 24 hrs if sending messages to traffic feed stops failing. For any assistance, contact My Oracle Support.

3.2.65 SCPTrafficFeedKafkaClusterUnhealthy

Table 3-68 SCPTrafficFeedKafkaClusterUnhealthy

Field	Details
Description	'Kafka cluster is marked unhealthy, Cause : {{$labels.ocscp_cause}}'
Summary	'Kafka cluster is marked unhealthy, cause: {{$labels.ocscp_cause}}, scp_fqdn: {{$labels.ocscp_fqdn}}, namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when the Kafka cluster is unhealthy. ocscp_metric_trafficfeed_cluster_unhealthy == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9026
Metric Used	ocscp_metric_trafficfeed_cluster_unhealthy
Recommended Action	**Cause**: The Kafka cluster is unhealthy. **Diagnostic Information**: • Check and diagnose OCNADD Kafka cluster. **Recovery**: This alert clears when the Kafka cluster recovers from the failure condition. For any assistance, contact My Oracle Support.

3.2.66 SCPTrafficFeedPartitionUnhealthy

Table 3-69 SCPTrafficFeedPartitionUnhealthy

Field	Details
Description	'Kafka partition {{$labels.kafka_partition_id}} is marked unhealthy, Cause : {{$labels.ocscp_cause}}'
Summary	'Kafka cluster is marked unhealthy, cause: {{$labels.ocscp_cause}}, partition_id: {{$labels.kafka_partition_id}}, topic: {{$labels.topic}}, scp_fqdn: {{$labels.ocscp_fqdn}}, namespace: {{$labels.namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when the Kafka partition is unhealthy. ocscp_metric_trafficfeed_partition_unhealthy == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9025
Metric Used	ocscp_metric_trafficfeed_partition_unhealthy
Recommended Action	**Cause**: The Kafka partition is unhealthy. **Diagnostic Information**: • Check and diagnose OCNADD Kafka cluster. **Recovery**: This alert clears when the Kafka partition recovers from the failure condition. For any assistance, contact My Oracle Support.

3.2.67 SCPServiceMeshFailure

Table 3-70 SCPServiceMeshFailure

Field	Details
Description	'SCP servicemesh failure encountered'
Summary	'SCP servicemesh failure encountered for nfservicetype: {{$labels.ocscp_nf_service_type}}, nftype: {{$labels.ocscp_nf_type}}, nfinstanceid: {{$labels.ocscp_nf_instance_id}}, serviceinstanceid: {{$labels.ocscp_service_instance_id}}, producerfqdn: {{$labels.ocscp_producer_host}}, responsecode: {{$labels.ocscp_response_code}} serverheader:{{$labels.ocscp_server_header}}, namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'SeverityInfo
Severity	Info
Condition	This alert is raised when service mesh failure occurs. increase(ocscp_metric_sidecarproxy_failures_total[2m]) > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.7030
Metric Used	ocscp_metric_sidecarproxy_failures_total

Table 3-70 (Cont.) SCPServiceMeshFailure

Field	Details
Recommended Action	**Cause**: Service mesh failure observed at SCP. **Diagnostic Information**: • Service mesh is unable to reach peer NF due to connection failures or host is not known to service mesh or some other errors at service mesh. • Verify the service mesh logs to detect error details. • Check sidecar status of peer NF. **Recovery**: This alert clears automatically after 2 minutes if there is no service mesh failure observed by SCP with the same dimensions. For any assistance, contact My Oracle Support.

3.2.68 SCPHealthCheckFailedForPeerSCP

Table 3-71 SCPHealthCheckFailedForPeerSCP

Field	Details
Description	'SCP HealthCheck failed for peer SCP'
Summary	'SCP HealthCheck failed for peer SCP. namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Info
Condition	This alert is raised when peer SCP or inter-SCP becomes unhealthy due to health check status and outlier detection. ocscp_interscp_health_check_status_failed == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9023
Metric Used	ocscp_interscp_health_check_status_failed
Recommended Action	**Cause**: When peer SCP is unhealthy to recieve any SBI message requests due to health check and outlier detection. **Diagnostic Information**: • Monitor if the overall average load of peer SCP is greater than the configured threshold value. • Check outlier detection. **Recovery**: This alert clears automatically if SCP-C decides SCP-P is healthy or available based on the current and previous status of outlier detection and health check. For any assistance, contact My Oracle Support.

3.2.69 SCPHealthCheckFailed

Table 3-72 SCPHealthCheckFailed

Field	Details
Description	'SCP HealthCheck failed'
Summary	'SCP HealthCheck failed. namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'

Table 3-72 (Cont.) SCPHealthCheckFailed

Field	Details
Severity	Info
Condition	This alert is raised when SCP is unhealthy because the overall average load of SCP is greater than the configured threshold. ocscp_health_check_status_failed == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.9024
Metric Used	ocscp_health_check_status_failed
Recommended Action	**Cause**: When SCP is unhealthy to receive any SBI message requests due to the overall average load. **Diagnostic Information**: Monitor if the overall average load of SCP is greater than the configured threshold value. **Recovery**: This alert clears automatically when the overall average load of SCP is less than the configured threshold value. For any assistance, contact My Oracle Support.

3.2.70 ScpWorkerPodPendingTransUtilizationAboveMinorThreshold

Table 3-73 ScpWorkerPodPendingTransUtilizationAboveMinorThreshold

Field	Details
Description	Worker Pending Transaction lead to minor level
Summary	'Worker Pending Transaction lead to minor level.namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when pending transaction utilization of SCP-Worker reaches MINOR level. ocscp_worker_pod_overload_control_pendingTrans_utilization_minor > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9014
Metric Used	ocscp_worker_pod_overload_control_pendingTrans_utilization_minor
Recommended Action	**Cause**: When pending transactions utilization of SCP-Worker reaches MINOR level. **Diagnostic Information**: • Monitor pending transactions usage while processing traffic. The threshold value of minor pending transaction utilization can be checked from database. • Check MINOR level logs of SCP-Worker for pending transaction status. **Recovery**: This alert clears automatically when pending transaction utilization is below MINOR threshold level. For any assistance, contact My Oracle Support.

3.2.71 ScpWorkerPodPendingTransUtilizationAboveMajorThreshold

Table 3-74 ScpWorkerPodPendingTransUtilizationAboveMajorThreshold

Field	Details
Description	Worker Pending Transaction lead to major level

Table 3-74 (Cont.) ScpWorkerPodPendingTransUtilizationAboveMajorThreshold

Field	Details
Summary	Worker Pending Transaction lead to major level. namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when pending transaction utilization of SCP-Worker reaches MAJOR level. ocscp_worker_pod_overload_control_pendingTrans_utilization_major > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9015
Metric Used	ocscp_worker_pod_overload_control_pendingTrans_utilization_major
Recommended Action	**Cause**: When pending transactions utilization of SCP-Worker reaches MAJOR level. **Diagnostic Information**: • Monitor pending transactions usage while processing traffic. The threshold value of major pending transaction utilization can be checked from database. • Check MAJOR level logs of SCP-Worker for pending transaction status. **Recovery**: This alert clears automatically when pending transaction utilization is below MAJOR threshold level. For any assistance, contact My Oracle Support.

3.2.72 ScpWorkerPodPendingTransUtilizationAboveCriticalThreshold

Table 3-75 ScpWorkerPodPendingTransUtilizationAboveCriticalThreshold

Field	Details
Description	Worker Pending Transaction lead to critical level
Summary	'Worker Pending Transaction lead to critical level. namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when pending transaction utilization of SCP-Worker reaches CRITICAL level. ocscp_worker_pod_overload_control_pendingTrans_utilization_critical > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9016
Metric Used	ocscp_worker_pod_overload_control_pendingTrans_utilization_critical
Recommended Action	**Cause**: When pending transactions utilization of SCP-Worker reaches CRITICAL level. **Diagnostic Information**: • Monitor pending transactions usage while processing traffic. The threshold value of critical pending transaction utilization can be checked from database. • Check CRITICAL level logs of SCP-Worker for pending transaction status. **Recovery**: This alert clears automatically when pending transaction utilization is below CRITICAL threshold level. For any assistance, contact My Oracle Support.

3.2.73 ScpWorkerPodPendingTransUtilizationAboveWarnThreshold

Table 3-76 ScpWorkerPodPendingTransUtilizationAboveWarnThreshold

Field	Details
Description	Worker Pending Transaction lead to Warn level
Summary	'Worker Pending Transaction lead to warn level. namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Warn
Condition	This alert is raised when pending transaction utilization of SCP-Worker reaches WARN level. ocscp_worker_pod_overload_control_pendingTrans_utilization_warn > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9017
Metric Used	ocscp_worker_pod_overload_control_pendingTrans_utilization_warn
Recommended Action	**Cause**: When pending transactions utilization of SCP-Worker reaches WARN level. **Diagnostic Information**: • Monitor pending transactions usage while processing traffic. The threshold value of warn pending transaction utilization can be checked from database. • Check WARN level logs of SCP-Worker for pending transaction status. **Recovery**: This alert clears automatically when pending transaction utilization is below WARN threshold level. For any assistance, contact My Oracle Support.

3.2.74 ScpWorkerPodResourceUtilizationAboveMinorThreshold

Table 3-77 ScpWorkerPodResourceUtilizationAboveMinorThreshold

Field	Details
Description	Worker overload control lead to minor level
Summary	'Worker overload control lead to minor level.namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when overload control resource utilization of SCP-Worker reaches MINOR level. ocscp_worker_pod_overload_control_resource_utilization_minor > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9018
Metric Used	ocscp_worker_pod_overload_control_resource_utilization_minor
Recommended Action	**Cause**: When overload control resource utilization of SCP-Worker reaches MINOR level. **Diagnostic Information**: • Monitor pending transactions usage while processing traffic. The threshold value of minor overload control resource utilization can be checked from database. • Check MINOR level logs of SCP-Worker for overload control resource utilization status. **Recovery**: This alert clears automatically when overload control resource utilization is below MINOR threshold level. For any assistance, contact My Oracle Support.

3.2.75 ScpWorkerPodResourceUtilizationAboveMajorThreshold

Table 3-78 ScpWorkerPodResourceUtilizationAboveMajorThreshold

Field	Details
Description	Worker overload control lead to major level
Summary	'Worker overload control lead to major level. namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when overload control resource utilization of SCP-Worker reaches MAJOR level. ocscp_worker_pod_overload_control_resource_utilization_major > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9019
Metric Used	ocscp_worker_pod_overload_control_resource_utilization_major
Recommended Action	**Cause**: When overload control resource utilization of SCP-Worker reaches MAJOR level. **Diagnostic Information**: • Monitor pending transactions usage while processing traffic. The threshold value of major overload control resource utilization can be checked from database. • Check MAJOR level logs of SCP-Worker for overload control resource utilization status. **Recovery**: This alert clears automatically when overload control resource utilization is below MAJOR threshold level. For any assistance, contact My Oracle Support.

3.2.76 ScpWorkerPodResourceUtilizationAboveWarnThreshold

Table 3-79 ScpWorkerPodResourceUtilizationAboveWarnThreshold

Field	Details
Description	'Worker overload control lead to Warn level'
Summary	'Worker overload control lead to warn level. namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Warning
Condition	This alert is raised when overload control resource utilization of SCP-Worker reaches WARN level. ocscp_worker_pod_overload_control_resource_utilization_warn > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9021
Metric Used	ocscp_worker_pod_overload_control_resource_utilization_warn

Table 3-79 (Cont.) ScpWorkerPodResourceUtilizationAboveWarnThreshold

Field	Details
Recommended Action	**Cause**: When overload control resource utilization of SCP-Worker reaches WARN level. **Diagnostic Information**: • Monitor pending transactions usage while processing traffic. The threshold value of warn overload control resource utilization can be checked from database. • Check WARN level logs of SCP-Worker for overload control resource utilization status. **Recovery**: This alert clears automatically when overload control resource utilization is below WARN threshold level. For any assistance, contact My Oracle Support.

3.2.77 ScpWorkerPodResourceUtilizationAboveCriticalThreshold

Table 3-80 ScpWorkerPodResourceUtilizationAboveCriticalThreshold

Field	Details
Description	Worker overload control lead to critical level
Summary	'Worker overload control lead to critical level. namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when overload control resource utilization of SCP-Worker reaches CRITICAL level. ocscp_worker_pod_overload_control_resource_utilization_critical > 0
OID	1.3.6.1.4.1.323.5.3.35.1.2.9020
Metric Used	ocscp_worker_pod_overload_control_resource_utilization_critical
Recommended Action	**Cause**: When overload control resource utilization of SCP-Worker reaches CRITICAL level. **Diagnostic Information**: • Monitor pending transactions usage while processing traffic. The threshold value of critical overload control resource utilization can be checked from database. • Check CRITICAL level logs of SCP-Worker for overload control resource utilization status. **Recovery**: This alert clears automatically when overload control resource utilization is below CRITICAL threshold level. For any assistance, contact My Oracle Support.

3.2.78 SCPDNSSRVNRFMigrationTaskFailure

Table 3-81 SCPDNSSRVNRFMigrationTaskFailure

Field	Description
Severity	critical
Condition	ocscp_configuration_dnssrv_nrf_migration_task_failure == 1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15001

Table 3-81 (Cont.) SCPDNSSRVNRFMigrationTaskFailure

Field	Description
Description	An alert is raised to notify that migration from static to DNS has failed.
Recommended Actions	**Cause**: • If migration to DNS task was waiting for DNS SRV data and wait time elapsed. • If any migration task fails due to no acknowledgement from other microservices. • Wait time elapsed in task completion response for any migration task. **Diagnostic Information**: Monitor that all the DNS SRV configurations are proper and that all SCP pods are up and running in the proper state. **Recovery**: • DNS SRV data, wait time elapsed: Once data is received from DNS, the alert will be cleared. • No acknowledgement: Keep retrying for success acknowledgement and clear or remove the alert on receiving success acknowledgement. • Wait time elapsed for completion response: Resend the event and wait. Repeat this until a completion response is received, and on receiving a completion response, clear the alert. In all the raised conditions, the alert will also be cleared on receiving the new migration task. For any assistance, contact My Oracle Support.

3.2.79 SCPDNSSRVNRFNonMigrationTaskFailure

Table 3-82 SCPDNSSRVNRFNonMigrationTaskFailure

Field	Description
Severity	critical
Condition	ocscp_configuration_dnssrv_nrf_non_migration_task_failure == 1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15003
Description	An alert is raised to notify that the non-migrated task has failed.
Recommended Actions	**Cause**: • If a non-migrated task fails due to no acknowledgement from other microservices. • Wait time elapsed in task completion response. **Diagnostic Information**: Monitor that all the DNS SRV configurations are proper and that all SCP pods are up and running in the proper state. **Recovery**: • No acknowledgement: Keep retrying for success acknowledgement and clear or remove the alert on receiving success acknowledgement. • Wait time elapsed: When a new task is submitted for the same SPN, this alert is immediately cleared; if subsequent task processing fails, again, the alert will be raised. For any assistance, contact My Oracle Support.

3.2.80 SCPDNSSRVNRFDuplicateTargetDetected

Table 3-83 SCPDNSSRVNRFDuplicateTargetDetected

Field	Description
Severity	critical
Condition	ocscp_configuration_dnssrv_nrf_duplicate_target_detected == 1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15002
Description	An alert is raised to notify that a duplicate target NRF has been detected in the DNS SRV records.
Recommended Actions	**Cause**: This alert is raised when a duplicate target FQDN is received from the DNS SRV for different NRF SRV FQDN(s). In this case, the target FQDNs received against the first NRF SRV FQDN in the scpc-configuration service from the scpc-alternate-resolution service shall be processed, but the target FQNDS received against the subsequent NRF SRV FQDN will be ignored, and this alert shall be raised. **Diagnostic Information**: Monitor that all the DNS SRV configurations are proper and that all SCP pods are up and running in the proper state. **Recovery**: • If for the same NRF SRV FQDN receives non-overlapping target FQDNs, then the alert will be cleared. For any assistance, contact My Oracle Support.

3.2.81 SCPHighResponseTimeFromProducer

Table 3-84 SCPHighResponseTimeFromProducer

Field	Description
Severity	Info
Condition	(sum(rate(ocscp_metric_upstream_service_time_total{ocscp_upstream_service_time="15000ms"}[2m])) by (kubernetes_namespace) + sum(rate(ocscp_metric_upstream_service_time_total{ocscp_upstream_service_time=">15000ms"}[2m])) by (kubernetes_namespace)) > 200
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15004
Description	It notifies when the traffic exceeds 200 messages per second and the response delay from the producer takes more than 10 seconds.
Recommended Actions	**Cause**: More than 200 messages per second have an upstream response time above 10 seconds. **Diagnostic Information**: Monitor metric metricocscp_metric_upstream_service_time_total with ocscp_upstream_service_time="15000ms" and ocscp_upstream_service_time=">15000ms". **Recovery**: An alert is cleared automatically when the number of responses with a response delay of more than 10 seconds falls below 200 messages per second. If Alert is not getting cleared, then check for any producer NFs or specific service request types that are taking more than 10 seconds to respond and take corrective actions if needed. Note that immediate action may not be needed, as this alter is informational. However, having too many requests with a long response delay may cause performance degradation at SCP. For any assistance, contact My Oracle Support.

3.2.82 SCPCGroupVersionDetectionFailed

Table 3-85 SCPCGroupVersionDetectionFailed

Field	Description
Severity	critical
Condition	ocscp_worker_cgroup_version_detection_failed == 1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15005
Description	Notify that cgroup version detection has failed.
Recommended Actions	**Cause**: SCP is unable to detect the cgroup version from the underlying kernel with the command "stat -fc %T /sys/fs/cgroup/." The possible expected value is either tmpfs or cgroup2fs. **Diagnostic Information**: • Option 1: Check worker error level logs for failure of cgroup version detection. • Option 2: – Login to the worker pod and run the command "stat -fc %T /sys/fs/cgroup." – Verify whether it outputs either tmpfs or cgroup2fs. **Recovery**: • Make sure the cgroup has either tmpfs or cgroup2fs. • Do same version upgrade on SCP.. For any assistance, contact My Oracle Support.

3.2.83 SCPCPUUsageFileReadFailed

Table 3-86 SCPCPUUsageFileReadFailed

Field	Description
Severity	critical
Condition	ocscp_worker_cpu_usage_file_read_failed == 1
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15006
Description	Notify that the CPU usage file read operation failed within the detected cgroup version.

Table 3-86 (Cont.) SCPCPUUsageFileReadFailed

Field	Description
Recommended Actions	**Cause**: SCP encountered a failure in performing a read operation for the CPU usage file within the detected cgroup version. The file path is determined based on the detected cgroup version. **Diagnostic Information**: • Option 1: Check worker warn level logs for failure in performing a read operation for the CPU usage file. • Option 2: – Log in to the worker pod and run the command "stat -fc %T /sys/fs/cgroup" to confirm whether it outputs either 'tmpfs' or 'cgroup2fs'. – Depending on the detected cgroup version, inspect the file located at /sys/fs/cgroup/cpu/cpuacct.usage for 'tmpfs' or /sys/fs/cgroup/cpu.stat for 'cgroup2fs'. – Ensure that the file exists at the specified path and possesses the required permissions for read operations. **Recovery**: • Verify that the files are appropriate according to the cgroup version and possess the necessary permissions for read operations. • Perform the same version upgrade on the SCP. For any assistance, contact My Oracle Support.

3.2.84 SCPIgnoreUnknownService

Table 3-87 SCPIgnoreUnknownService

Field	Description
Severity	Info
Condition	increase(ocscp_ignore_unknown_service_total[24h]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15000
Description	An alert is raised to notify that SCP ignored an unknown service in the NF profile.
Recommended Actions	**Cause**: SCP has received the NF profile with an unknown service and processed the profile by ignoring this unknown service. **Diagnostic Information**: Check the received NF profile for the unknown services. **Recovery**: If the unknown services are not present in the NF profile in the next scrapping interval, then the alert will be cleared. For any assistance, contact My Oracle Support.

3.2.85 SCPWorkerSSLCertificateOnCriticalExpiry

Table 3-88 SCPWorkerSSLCertificateOnCriticalExpiry

Field	Description
Severity	Critical
Condition	ocscp_metric_ssl_certificate_expire_total == 1
OID used for SNMP Traps	11.3.6.1.4.1.323.5.3.35.1.2.15010

Table 3-88 (Cont.) SCPWorkerSSLCertificateOnCriticalExpiry

Field	Description
Description	An alert is raised whenever the SCP SSL certificate is about to expire, based on the configured threshold values.
Recommended Actions	**Cause**: The SSL certificate expiration is approaching the configured critical expiry time. **Diagnostic Information**: • Whenever this alert is raised, it indicates that the SSL certificates configured for SCP are about to expire. • Verify the certificate expiry date. **Recovery**: The SCP SSL secret needs to be updated with renewed SSL certificates. For any assistance, contact My Oracle Support.

3.2.86 SCPWorkerSSLCertificateOnMajorExpiry

Table 3-89 SCPWorkerSSLCertificateOnMajorExpiry

Field	Description
Severity	major
Condition	ocscp_metric_ssl_certificate_expire_total == 2
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15011
Description	An alert is raised whenever the SCP SSL certificate is about to expire, based on the configured threshold values.
Recommended Actions	**Cause**: The SSL certificate expiration is approaching the configured major expiry time. **Diagnostic Information**: • Whenever this alert is raised, it indicates that the SSL certificates configured for SCP are about to expire. • Verify the certificate expiry date. **Recovery**: The SCP SSL secret needs to be updated with renewed SSL certificates. For any assistance, contact My Oracle Support.

3.2.87 SCPWorkerSSLCertificateOnMinorExpiry

Table 3-90 SCPWorkerSSLCertificateOnMinorExpiry

Field	Description
Severity	minor
Condition	ocscp_metric_ssl_certificate_expire_total == 3
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15012
Description	An alert is raised whenever the SCP SSL certificate is about to expire, based on the configured threshold values.

Table 3-90 (Cont.) SCPWorkerSSLCertificateOnMinorExpiry

Field	Description
Recommended Actions	**Cause**: The SSL certificate expiration is approaching the configured minor expiry time. **Diagnostic Information**: • Whenever this alert is raised, it indicates that the SSL certificates configured for SCP are about to expire. • Verify the certificate expiry date. **Recovery**: The SCP SSL secret needs to be updated with renewed SSL certificates. For any assistance, contact My Oracle Support.

3.2.88 SCPIngressConnectionEstablishmentFailure

Table 3-91 SCPIngressConnectionEstablishmentFailure

Field	Description
Severity	info
Condition	increase(ocscp_worker_https_ingress_connection_failure_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15007
Description	An alert is raised whenever any ingress HTTPS connection is failed.
Recommended Actions	**Cause**: Whenever an Ingress HTTPS connection establishment fails. **Diagnostic Information**: This alert may be raised if any connection establishment fails or if a handshake fails. **Recovery**: The alert will be cleared if there are no ingress HTTPS connection failures in the next scrape interval (2 minutes). For any assistance, contact My Oracle Support.

3.2.89 SCPEgressConnectionEstablishmentFailure

Table 3-92 SCPEgressConnectionEstablishmentFailure

Field	Description
Severity	info
Condition	increase(ocscp_worker_https_egress_connection_failure_total[2m]) > 0
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.35.1.2.15008
Description	An alert is raised whenever any egress HTTPS connection is failed.
Recommended Actions	**Cause**: Whenever an egress HTTPS connection establishment fails to send the request to producer NFs. **Diagnostic Information**: This alert may be raised if any connection establishment fails or if a handshake fails. **Recovery**: The alert will be cleared if there are no engress HTTPS connection failures in the next scrape interval (2 minutes). For any assistance, contact My Oracle Support.

3.2.90 SCPNrfProxyOauthQueuesUtilizationAboveCriticalThreshold

Table 3-93 SCPNrfProxyOauthQueuesUtilizationAboveCriticalThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.namespace}}, podname: {{$labels.pod}}: SCP NrfProxy Oauth Queues Utilization Above Critical Threshold'
Summary	'SCP NrfProxy Oauth Queues Utilization Above Critical Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Critical
Condition	This alert is raised when SCP NrfProxy Oauth Queues Utilization is above the critical threshold. ocscp_nrfproxy_oauth_queue_alert{severity="CRITICAL"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.14000
Metric Used	ocscp_nrfproxy_oauth_queue_alert
Recommended Action	**Cause**: When NrfProxy Task queues are filled, the traffic exceeds the limit. **Diagnostic Information**: • Monitor the traffic toward nrfProxy to pod using the Grafana dashboard. • Refer to the rate of ocscp_nrfproxy_oauth_queue_alert metric on the Grafana dashboard. **Recovery**: This alert clears automatically when the traffic decreases below the critical threshold. For any assistance, contact My Oracle Support.

3.2.91 SCPNrfProxyOauthQueuesUtilizationAboveMajorThreshold

Table 3-94 SCPNrfProxyOauthQueuesUtilizationAboveMajorThreshold

Field	Details
Description	'instancename: {{$labels.instance}}, namespace: {{$labels.namespace}}, podname: {{$labels.pod}}: SCP NrfProxyOauth Queues Utilization Above Major Threshold'
Summary	'SCP NrfProxy Oauth Queues Utilization Above Major Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Major
Condition	This alert is raised when SCP NrfProxy Oauth Queues Utilization is above the major threshold. ocscp_nrfproxy_oauth_queue_alert{severity="MAJOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.14001
Metric Used	ocscp_nrfproxy_oauth_queue_alert
Recommended Action	**Cause**: When NrfProxy Task queues are filled, the traffic exceeds the limit. **Diagnostic Information**: • Monitor the traffic toward nrfProxy to pod using the Grafana dashboard. • Refer to the rate of ocscp_nrfproxy_oauth_queue_alert metric on the Grafana dashboard. **Recovery**: This alert clears automatically when the traffic decreases below the Major threshold. For any assistance, contact My Oracle Support.

3.2.92 SCPNrfProxyOauthQueuesUtilizationAboveMinorThreshold

Table 3-95 SCPNrfProxyOauthQueuesUtilizationAboveMinorThreshold

Field	Details
Description	description: 'instancename: {{$labels.instance}}, namespace: {{$labels.namespace}}, podname: {{$labels.pod}}: SCP NrfProxyOauth Queues Utilization Above Minor Threshold'
Summary	'SCP NrfProxy Oauth Queues Utilization Above Minor Threshold, instancename: {{$labels.instance}}, namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}'
Severity	Minor
Condition	This alert is raised when SCP NrfProxy Oauth Queues Utilization is above the minor threshold. ocscp_nrfproxy_oauth_queue_alert{severity="MINOR"} == 1
OID	1.3.6.1.4.1.323.5.3.35.1.2.14002
Metric Used	ocscp_nrfproxy_oauth_queue_alert
Recommended Action	**Cause**: When NrfProxy Task queues are filled, the traffic exceeds the limit. **Diagnostic Information**: • Monitor the traffic toward nrfProxy to pod using the Grafana dashboard. • Refer to the rate of ocscp_nrfproxy_oauth_queue_alert metric on the Grafana dashboard. **Recovery**: This alert clears automatically when the traffic decreases below the Major threshold. For any assistance, contact My Oracle Support.

3.3 Configuring Alerts

This section lists the configuring alerts.

3.3.1 Applying Alerts Rule to CNE without Prometheus Operator

SCP Helm Chart Release Name: _NAME_

Prometheus NameSpace: _Namespace _

Perform the following procedure to configure Service Communication Proxy alerts in Prometheus.

1. Run the following command to check the name of the config map used by Prometheus:

   ```
   $kubectl get configmap -n <_Namespace_>
   ```

   Example:

   ```
   $kubectl get configmap -n prometheus-alert2
   NAME                                  DATA   AGE
   lisa-prometheus-alert2-alertmanager   1      146d
   lisa-prometheus-alert2-server         4      146d
   ```

2. Take a backup of the current config map of Prometheus. This command saves the configmap in the provided file. In the following command, the configmap is stored in the */tmp/tempConfig.yaml* file:

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

   Example:

   ```
   $ kubectl get configmaps lisa-prometheus-alert2-server -o yaml -n
   prometheus-alert2 > /tmp/tempConfig.yaml
   ```

3. Check and delete the "alertsscp" rule if it has already configured in the prometheus config map. If configured, this step removes the " alertsscp " rule. This is an optional step if configuring the alerts for the first time.

   ```
   $ sed -i '/etc\/config\/alertsscp/d' /tmp/tempConfig.yaml
   ```

4. Add the "alertsscp" rule in the configmap dump file under the ' rule_files ' tag.

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

5. Update the configmap using below command. Ensure to use the same configmap name that was used to take a backup of the prometheus configmap.

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

   Example:

   ```
   $ kubectl replace configmap lisa-prometheus-alert2-server -f /tmp/
   tempConfig.yaml
   ```

6. Run the following command to patch the configmap with a new "alertsscp" rule:

   > **Note**
   >
   > The patch file provided is the `ocscp_csar_23_2_0_0_0.zip` folder provided with SCP, that is, `SCPAlertrules.yaml`.

   ```
   $ kubectl patch  configmap _NAME_-server -n _Namespace_ --type merge --
   patch "$(cat ~/SCPAlertrules.yaml)"
   ```

   Example:

   ```
   $ kubectl replace configmap lisa-prometheus-alert2-server -f /tmp/
   tempConfig.yaml
   ```

> **Note**
>
> Prometheus takes about 20 seconds to apply the updated Config map.

3.3.2 Applying Alerts Rule to CNE with Prometheus Operator

Perform the following procedure to apply alerts rule to Cloud Native Environment (CNE) with Prometheus Operator (CNE 1.9.0 and later).

- Run the following command to apply SCP alerts file to create Prometheus rules Custom Resource Definition (CRD):

  ```
  kubectl apply -f <file_name> -n <scp namespace>
  ```

  Where,

  - `<file_name>` is the SCP alerts file.
  - `<scp namespace>` is the SCP namespace.

  Example:

  ```
  kubectl apply -f ocscp_alerting_rules_promha_24.2.6.yaml -n scpsvc
  ```

  Sample file delivered with SCP package:

  ```
  ocscp_alerting_rules_promha_24.2.6.yaml
  ```

3.3.3 Configuring Service Communication Proxy Alert using the SCPAlertrules.yaml file

> **Note**
>
> Default NameSpace is **scpsvc** for Service Communication Proxy. You can update the NameSpace as per the deployment.

To access the `scpAlertsrules_<scp release number>.yaml` file from the `Scripts` folder of `ocscp_csar_23_2_0_0_0.zip`, download the SCP package from My Oracle Support as described in "Downloading the SCP Package " in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

Alerts Details

Description and summary for alerts are added by the Prometheus alert manager.

Alerts are supported for three different resources/routing crosses threshold.

- SCPIngress Traffic Rate Above Threshold
  - Has three threshold level Minor (above 1400 mps to 2000mps), Major (1600 to 1800 mps), Critical (above 1800 mps). These values are configurable.

  - In the description, information is presented similar to: "Ingress Traffic Rate at Locality: <Locality of scp> is above <threshold level (minor/major/critical> threshold (i.e. <value of threshold>)"
  - In Summary: "Namespace: <Namespace of scp deployment that Locality>, Pod: <SCP-worker Pod name>: Current Ingress Traffic Rate is <Current rate of Ingress traffic > mps which is above 70 Percent of Max MPS(<upper limit of ingress traffic rate per pod>)"

> **Note**
>
> Ingress traffic rate is per scp-worker pod in a namespace at particular SCP-Locality. Currently, 2000mps is the upper limit for per scp-worker pod.

- SCP Routing Failed For Service
  - It alerts for which NF Service Type and NF Type at particular locality, Routing failed
  - Description: "Routing failed for service"
  - Summary: "Routing failed for service: NFService Type = <Message NF Service Type>, NFType = <Message NF Type>, Locality = <SCP Locality where Routing Failed> and value = <Accumulated failure till now, of such message for NFType and NFService Type>"

> **Note**
>
> The value field currently does not provide the number of failures in particular time interval, instead it provides the total number of Routing failures.

- SCP Pod Memory Usage: Type of alert is SCPWorkerPodMemoryUsage.
  - Pod memory usage for SCP Pods (Soothsayer and Worker) deployed at a particular node instance is provided.
  - The Soothsayer pod threshold is 8 GB
  - The Worker pod threshold is 4 GB
  - Summary: Instance: "<Node Instance name>, NameSpace: <Namespace of SCP deployment>, Pod: <(Soothsayer/Worker) Pod name>: <Soothsayer/Worker> Pod High Memory usage detected"
  - Summary: "Instance: "<Node Instance name>, Namespace: <Namespace of SCP deployment>, Pod: <(Soothsayer/Worker) Pod name>: Memory usage is above <threshold value>G (current value is: <current value of memory usage>)"

3.3.4 Configuring Alert Manager for SNMP Notifier

Grouping of alerts is based on:

- podname
- alertname
- severity
- namespace
- nfServiceType

- nfServiceInstanceId

User needs to add subroutes for SCP alerts in AlertManager config map as below:

1. Take a backup of the current config map of Alertmanager by running the following command:

   ```
   kubectl get configmaps <NAME-alertmanager> -oyaml -n <Namespace> > /tmp/
   bkupAlertManagerConfig.yaml
   ```

   Example:

   ```
   kubectl get configmaps occne-prometheus-alertmanager -oyaml -n occne-infra
   > /tmp/bkupAlertManagerConfig.yaml
   ```

2. Edit Configmap to add subroute for SCP Trap OID:

   ```
   kubectl edit configmaps <NAME-alertmanager> -n <Namespace>
   ```

   Example:

   ```
   kubectl edit configmaps occne-prometheus-alertmanager -n occne-infra
   ```

3. Add the subroute under 'route' in configmap:

   ```
   routes:
         - receiver: default-receiver
           group_interval: 1m
           group_wait: 10s
           repeat_interval: 9y
           group_by: [podname, alertname, severity, namespace, nfservicetype,
   nfserviceinstanceid, servingscope, nftype]
           match_re:
             oid: ^1.3.6.1.4.1.323.5.3.35.(.*)
   ```

MIB Files for SCP

There are two MIB files which are used to generate the traps. The user need to update these files along with the Alert file in order to fetch the traps in their environment.

- ocscp_mib_tc_24.2.6.mib: This is considered as SCP top level mib file, where the Objects and their data types are defined.
- ocscp_mib_24.2.6.mib: This file fetches the Objects from the top level mib file and based on the Alert notification, these objects can be selected for display.

> **Note**
>
> MIB files are packaged with `ocscp_csar_23_2_0_0_0.zip`. You can download the file from MOS as described in *Oracle Communications Cloud Native Core, Service Communication Proxy Installation, Upgrade, and Fault Recovery Guide*.

3.4 Configuring SCP Alerts for OCI

To configure SCP alerts for OCI, OCI supports metric expressions written in MQL (Metric Query Language) and therefore requires `ocscp_oci_alertrules_24.2.6.zip` file for configuring alerts in OCI observability platform. For more information, see *Oracle Communications Cloud Native Core, OCI Adaptor Deployment Guide*.