Oracle® Communications

Cloud Native Core, Certificate Management Troubleshooting Guide

Release 25.2.200
G48070-01
March 2026

Oracle Communications Cloud Native Core, Certificate Management Troubleshooting Guide, Release 25.2.200

G48070-01

Contents

Preface

Documentation Accessibility

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

Access to Oracle Support

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

Diversity and Inclusion

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

Conventions

The following text conventions are used in this document:

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

My Oracle Support

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

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

1. Select **2** for New Service Request.
2. Select **3** for Hardware, Networking and Solaris Operating System Support.
3. Select one of the following options:
   - 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**.

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

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

Acronyms

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

Table Acronyms

Acronym	Description
3GPP	3rd Generation Partnership Project
API	Application Programming Interface
CA	Certification Authority is a trusted entity that issues Secure Sockets Layer (SSL) certificates. CAs are also called issuer in this document.
CCA	Client Credentials Assertions
CMP	Certificate Management Protocol
CMP Identity Certificate	Certificate that corresponds to and certifies the CMP Identity Key. It is included in the CMPv2 requests for authentication by CA.
CMP Identity Key	Private Key used by Certificate Management to sign the CMPv2 requests and establish trust between Certificate Management and CA.
CNC	Cloud Native Core
CNC Console	Cloud Native Configuration Console
DNS	Domain Name Server
ECC	Elliptic Curve Cryptography
EE	End Entity
HNC	Heirarchical Namespace Controller
HSM	Hardware Security Module
IDP	Identity Provider
IR	Initialization Requests
OCCM	Oracle Communications Certificate Management
PKI	Public Key Infrastructure
PoP	Proof of Possession
RA	Registration Authority
RSA	Rivest-Shamir-Adleman
SAN	Subject Alternative Name
URI	Uniform Resource Indicator
URN	Uniform Resource Name

What's New in This Guide

This section lists the documentation updates for Release 25.2.2xx in *Oracle Communications Cloud Native Core, Certificate Management Troubleshooting Guide*.

Release 25.2.200 - G48070-01, March 2026

No updates have been made in this release.

1 Introduction

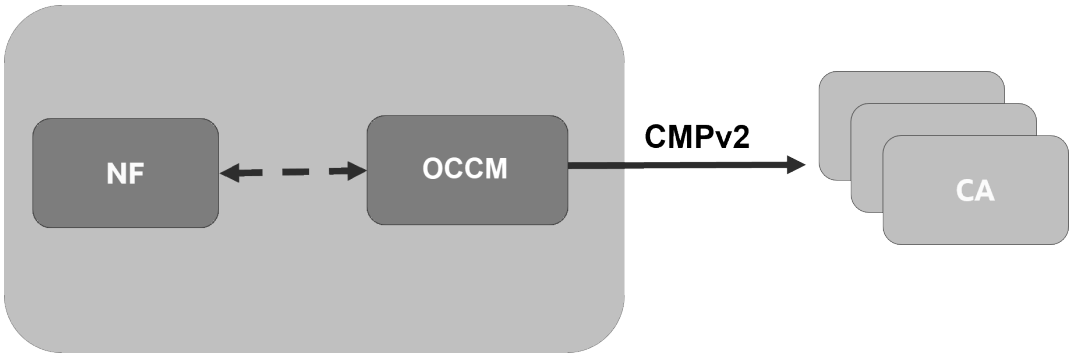
This document provides information about the Oracle Communication Cloud Native Core, Certificate Management troubleshooting scenarios.

1.1 Overview

OCCM integrates with the Certificate Authority(s) using Certificate Management Protocol Version 2 (CMPv2) and RFC4210 to facilitate these certificate management operations:

- Operator-initiated certificate creation
- Operator-initiated certificate recreation
- Automatic certificate monitoring and renewal

Figure 1-1 OCCM Integration with CA



OCCM supports transport of CMPv2 messages using HTTP-based protocol.

OCCM provides the following mechanisms to establish initial trust between OCCM and CA(s):

1. Certificate-based message signing
2. Pre-shared key or MAC based authentication

All the subsequent CMPv2 procedures are authenticated using the certificate-based mechanism in compliance with 3GPP TS 33.310.

The keys and X.509 certificates are managed using Kubernetes secrets.

1.2 Reference

Refer to the following documents for more information:

- *Oracle Communications Cloud Native Core, Certificate Management Network Impact Report*
- *Oracle Communications Cloud Native Core, Certificate Management User Guide*

- *Oracle Communications Cloud Native Core, Certificate Management REST Specification Guide*
- *Oracle Communications Cloud Native Core, Certificate Management Installation, Upgrade, and Fault Recovery Guide*

2
Logs

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

2.1 Log Levels

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

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

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

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

2.2 Collecting Logs

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

- Run the following command to get the PODs details:

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

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

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

- Store the log in a file using the following command:

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

- (Optional) You can also 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>
  ```

2.3 Understanding Logs

This section provides information on how to read logs

Log JSON Format

The following is a sample log for OCCM services:

```
{
     "instant": <epoch time in nanoseconds>,
     "thread": <threadName>,
     "level": <log_level>,
     "loggerName": <name_of_the_logging_class>,
     "message": <message>,
     "contextMap": <context_map>,
     "threadId": <threadId>,
     "threadPriority": <threadPriority>,
     "messageTimestamp": <timestamp_in_readable_format>,
     "application": occm,
     "microservice": occm,
     "cluster": <Deployment cluster name>,
     "namespace": <release namespace>,
     "node": <K8s node on which pod is running>,
     "pod": <name_of_the_pod>
}
```

The log message format is same for all the OCCM services. All logs are represented in JSON format.

The following table describes key attributes of a log message:

Table 2-1 Log Attributes

Attribute	Description	Example
instant	The Date and Time the event occurred in epoch second and nano seconds	"instant": { "epochSecond": 1590045388, "nanoOfSecond": 339789000}
thread	Name of the thread	"thread": "pool-7-thread-2"
level	Level of the log. It can be: Log level (INFO, WARN, DEBUG, ERROR)	"level": "INFO"
loggerName	Name of the class that generated the log	"loggerName": "com.oracle.cgbu.cne.occm.util.CmpUtils"
message	Information about the event	"message": "Started Application....."
contextMap	It holds information added to threadContext, if any.	"contextMap": { }

Table 2-1 (Cont.) Log Attributes

Attribute	Description	Example
threadId	Id of the thread	"threadId":"34"
threadPriority	Priority assigned to the thread	"threadPriority": 5
messageTimestamp	Time represented in human readable format and in UTC. Format is date:yyyy-MM-dd'T'HH:mm:ss.SSSZ EFK friendly and also follows Oracle Standards.	"messageTimestamp": 2023-12-04T02:51:46.458+0000
application	Application name	"application":"occm"
microservice	Micro service name	"microservice":"occm"
cluster	Deployemnt cluster name	"cluster":"occm"
namespace	Release namespace	"namespace":"occncc-thrust5-01"
node	K8s node where pod is running	"node": "thrust5-k8s-node-31"
pod	Name of the pods where the log is generated	"pod" : "occm-occm-949b9c7d4-qh5k7"

Table 2-2 OCCM Message Format

Name	Description	Example	Possible Values
UUID	UUID of the resource. Can be CertId or IssuerId	CertId= 7e2aa1e2-17ee-4601-b334-3bfd0d677d55	Any randomly generated UUID.
Name	Name of the resource.CertName or IssuerName	CertName= SCPTLS	Any string
NF	Name of the NF	NF= SCP	Any string
ErrorCode	In case of any failure, OCCM error code is logged.	ErrorCode= ERR_PROCESS_START_FAILURE	Error code from a list of OCCM error codes.
ErrorMsg	In case of any failure, error message is logged.	ErrorMsg=An error occurred during cert creation. Error running CMP command.	NA
ErrorCause	In case of any failure, error cause is logged.	ErrorCause=java.io.IOException: error=2, No such file or directory	NA

2.4 Accessing Logs

This section provides information about how to access the logs.

The OCCM application logs can be accessed in following ways:

1. Run the following command to view logs of an OCCM application pod:

```
$ kubectl logs -f -n <occm_namespace> <pod_name>
```

For example:

```
$ kubectl logs -f -n occm occm-occm-77df795fb5-wv2sb
```

2. OCCM uses cloud native supported logging framework to view the logs.
   For example: Elasticsearch, Fluentd, and Kibana (EFK) can be used with CNC Console to view the OCCM logs.

3 Using Debug Tool

Overview

The Debug Tools provides third party troubleshooting tools for debugging the runtime issues for lab and production environment. Following are the available tools:

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

Running the Debug Tool

> **Note**
>
> While testing in OCCNE environment, check the Kyverno policies and make sure to exclude the namespace in the disallow-capabilities.

To run the debug tool:

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

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

   After installation the debug-tool container will get injected into the pods, sample get pod output is here :

   ```
   [root@primary ~]# kubectl get po -n occm-ns
   NAME                          READY   STATUS    RESTARTS   AGE
   occm-occm-58db58648-d44kp   2/2     Running   0          13h
   ```

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

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

   For example:

   ```
   $ kubectl exec -it occm-occm-58db58648-d44kp -c tools -n occm-ns bash
   ```

3. Run the debug tools:

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

For 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>
```

For example:

```
$ kubectl cp -c tools -n occm-ns occm-occm-58db58648-d44kp:/tmp/
capture.pcap /tmp/
```

Enable Debug Tools

Debug tools container can be enabled/disabled for OCCM by using helm install or helm upgrade command.

Run the following command to enable/disable OCCM after updating custom-occm_values.yaml file on an installed setup:

```
$ helm upgrade <release_name> -f custom_occm_values-<version>.yaml  <helm-
repo> --version <helm_version>
```

For Example:

```
$ helm upgrade occm -f custom-occm_values_<version>.yaml  ocspf-helm-repo/
occm --version 25.1.200
```

4
Troubleshooting OCCM

This section provides information to troubleshoot common errors that occur while installing and upgrading OCCM.

4.1 Configuration Related Issues

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

4.1.1 Issuer and Certificate Related Errors

This section describes the following issuer and certificate related issues and their resolution steps:

- Unable to Edit Issuer or Certificate Configuration After Upgrade as Namespace is not Included in the Accessed Namespaces List
- Unable to Edit Certificate Configuration

Secret Name Format Error

Problem: The format in which the Kubernetes secret is provided is incorrect.

For example:

test_secret : Here, underscore is not allowed

Test-secret: Here, uppercase is not allowed

Solution: You must provide a valid string that is in compliance with kubernetes regex. It must have lower case alphanumeric characters, '-' or '.', and must start and end with an alphanumeric character. For more information, see Oracle Communications Cloud Native Core, Certificate Management Installation, Upgrade and Fault Recovery Guide.

For example:

occm-mac-secret

nrf-tls-secret

Namespace Format Error

Problem: The format in which the Kubernetes namespace is provided in the secret input is incorrect.

For example:

test_ns : Here, underscore is not allowed

Test-ns: Here, uppercase is not allowed

Solution: You must provide a valid string in compliance with Kubernetes regex. It must have lower case alphanumeric characters or '-', and must start and end with an alphanumeric character.

For example:

occm-ns

ocncc-thrust5-02

Secret Key Error

Problem: Secret data can't be found against key(s) and the configured secret key(s) are incorrect. Configured keys are not present in the Kubernetes secret or incorrect key names are provided in the input configuration.

Solution: Revisit the Kubernetes secret and provide the correct keys (filenames) in the configuration.

Input String Error

Problem: The number of characters in the string entered by the user exceeds the character limit.

Solution: The user must enter a string that does not exceed the character limit.

Incomplete TrustStore Secret Error

Problem: OCCM TrustStore secret input has missing fields. This could be because OCCM TrustStore input secret is incomplete. CA certs are missing for certificate validation.

Solution: Verify the OCCM TrustStore input secret and provide a valid one.

Invalid Secret Name

Problem: Secret name already in use as a certificate configuration already points to the same destination secret.

Solution: Provide a unique destination secret name.

Repeated Secret Error

Problem: Secret already exists on the server. In automatic life cycle management of certificate, a fresh secret is created.

Solution: Either provide a unique secret name or set the override secret flag to true. This will enable OCCM to override the existing secret.

Secret Doesn't Exist Error

Problem: Secret doesn't exist with this name. This could be because secret holding the manually created certificate (not via OCCM) doesn't exist on the server. For OCCM to start monitoring it, you must provide the corresponding input secret holding the key or certificate.

Solution: Provide details of the Kubernetes secret holding key or certificate.

Unique Secret Key Error

Problem: Secret key should be unique for same secret names. Same destination secrets may be having same keys in the Certificate configuration. Secret Key (FileNames) should be unique to avoid overriding of data.

For Example:

```
"privateKeyK8sSecretOut": {
            "namespace": "occm",
            "name": "test-secret",
            "key": "occm.pem"
    }

"certK8sSecretOut": {
        "namespace": "occm",
        "name": "test-secret",
        "key": "occm.pem"
    }
```

Solution: Provide unique destination secret (name, namespace, and key) in the configuration.

For Example:

```
"privateKeyK8sSecretOut": {
            "namespace": "occm",
            "name": "test-secret",
            "key": "nrfkey.pem"
```

```
    }

"certK8sSecretOut": {
        "namespace": "occm",
        "name": "test-secret",
        "key": "nrfcert.pem"
    }
```

CA Bundle Secret Error

Problem: CA Bundle secret doesn't exist. This could be because CA bundle check is skipped if the input configuration is not provided. If provided, it is validated whether the provided secret exists or not.

Solution: Either skip providing input CA bundle secret details or provide a valid secret.

Invalid MAC Secret Error

Problem: Invalid MAC secret has been passed because the MAC secret provided in input MAC secret is not in valid format.

For example:

12345: Here, the secret doesn't start with the prefixes.

Solution: MAC secret is expected to have following arguments.

pass: password

env: var

file: pathname

fd: number

stdin

Invalid File Format

Problem: The key or certificate files provided don't have a valid file name. This could be because the file doesn't have an extension or a period in the end.

For example:

file: This has a period at the end

abc: This doesn't have an extension

Solution: Provide a valid file name with OCCM supported extensions.

For example:

occmkey.pem

occmcert.pem

Delete Issuer Error

Problem: Issuer can not be deleted as it is in use by certificate(s).

Solution: Delete the mapped certificates first, followed by the corresponding issuer.

Issuer ID Error

Problem: Issuer ID and name do not match because the issuer edit payload doesn't have corresponding issuer ID and name.

Solution: Verify the payload and provide the name corresponding to the issuer ID or vice versa.

Issuer Already Exists Error

Problem: Issuer already exists with given name.

Solution: Provide a unique issuer name.

Incorrect UUID Error

Problem: The uuid in the request parameter does not match the uuid in the uuid in the request body attributes. This could be for the following reasons:

- The uuid in the request parameter is blank.
- The uuid does not match the uuid in the certificate configuration.

Solution: The uuid in the request parameter must match the uuid in the request body attributes. Update the uuid in the request parameter.

Unable to Trigger Recreate Request

Problem: The user is unable to trigger a new recreate request because a request has already been received for the uuid.

Solution: An older recreate request is in progress. User can trigger a new request when the previous request completes processing.

Recreation Request Rejected as the Authentication Input has expired

Problem: The user is unable to recreate the OCCM certificate because the authentication input for OCCM certificate or the certificate configured under **Initial CMP Client(OCCM) Authentication Options** has expired.

Solution: Configure a valid active certificate in the authentication input for the OCCM certificate under **Initial CMP Client(OCCM) Authentication Options.**

Recreation Request Rejected as the Authentication Input is Not Available

Problem: The user is unable to recreate the OCCM certificate because no authentication input for OCCM under **Initial CMP Client(OCCM) Authentication Options** has been given.

Solution: Configure a valid active certificate in the authentication input for the OCCM certificate under **Initial CMP Client(OCCM) Authentication Options.**

Unable to Merge Certificate and Certificate Chain

Problem: The request to merge certificate and certificate chain is rejected if the Merge Cert and Chain option is selected but the secret items for the certificate and certificate chain are different.

Solution: Configure same secret item for both certificate and certificate chain.

Namespace is not Included in the Accessed Namespaces List

Problem: The user encounters logs that say the namespace used in the issuer or certificate configuration is not a part of the accessed namespaces list when upgrading OCCM. This indicates that the namespace used in the previous deployment was not included in the `occmAccessedNamespaces` list. This can happen for the following possible reasons:

- The certificate may have failed in the earlier deployment because the namespace did not have the permissions needed to create or read secrets.
- A sub-namespace might have been used which was not added to the `occmAccessedNamespaces` list.

These warning logs indicate this namespace will not be accepted unless it is explicitly added to the `occmAccessedNamespaces` list.

Solution:

1. If a certificate fails because of namespace validation, the user can delete and create it again, and ensure that the namespace is included in the `occmAccessedNamespaces` list. Alternatively, the user can edit the certificate configuration or add the missing namespace to the list.
2. For sub-namespaces requests using sub-namespaces will be rejected unless they are already present in the `occmAccessedNamespaces` list.

Unable to get Resource Secrets in the Namespace as the Secret is Forbidden

Problem: The user may encounter the forbidden secret error for the following possible reasons:

- They don't have the permissions required to create secrets in the specified namespace.
- When using a custom ServiceAccount, all relevant namespaces must be listed in `occmAccessedNamespaces`. This error happens if the list includes a namespace not covered by the custom service account.

Solution: There are two possible solutions for this scenario:

- Delete the issuer or certificate configuration and update the namespace to one listed in `occmAccessedNamespaces`.
- Create the necessary roles and role bindings using the custom service account to enable access to the desired namespace.

Unable to Edit Issuer or Certificate Configuration After Upgrade as Namespace is not Included in the Accessed Namespaces List

Problem: If a namespace is used to create an issuer or certificate in a previous release, but that namespace is not included in the `occmAccessedNamespaces` list, the user will encounter an error when trying to edit the issuer or certificate after upgrading. In this scenario, the user will see the "Input namespace is not a part of the accessed namespace list." error message or the namespace won't be visible in the UI.

Solution: The user must ensure that the namespace is added to the `occmAccessedNamespaces` list before upgrading.

Unable to Edit Certificate Configuration

Problem: The edited configuration contains fields that can not be edited after the certificate is created. This error is seen when the user wants to edit those fields for successfully created

certificate (the current status may or may not be READY) that can not be edited after the certificate already exists.

Solution: The following fields can't be edited if the certificate already exists for the configuration:

- Name
- Cert Type
- Network Function
- Creation Mode
- Overwrite Secret
- Kubernetes secret details such as name, namespace, and key

4.1.2 CMP Related Issues

This section describes the following CMP related issues and their resolution steps:

- Server URL Error
- Issuer Configuration Error
- CMP Server Certificate Error
- Certificate Path Validation Error

Server URL Error

Problem: The issuer URL provided in the serverURL field of issuer configuration is not reachable.Could be an incorrect URL (incorrect port etc). This is causing the following errors:

CMP error: error sending server <server IP>

CMP error:transfer error

Error running CMP command

Solution: Provide a valid server URL by editing issuer configuration.

Issuer Configuration Error

Problem: Pre-shared key (MAC secret) configured in 'CMP protection for OCCM certificate' is incorrect. This is causing the following errors:

CMP error: wrong pbm value

CMP error: error validating protection

Solution: Update the issuer config with correct secret

CMP Server Certificate Error

Problem: The server certificate configured in the OCCM trust store configuration is different from that of the sender (CMP or CA server) certificate. This certificate is used for verifying signature-based protection of CMP response messages. This is causing the following errors:

CMP info: received IP

CMP info: actual name in sender DN field =<>

CMP info: does not match expected sender=<DN from server cert configured in OCCM trust store>

Solution: Update the issuer configuration with correct secret

Certificate Path Validation Error

Problem: The certificates configured in OCCM trust store are invalid or incomplete for certificate path validation of the CMP server certificate. This is causing the following errors:

CMP info: received IP

CMP error: no suitable sender cert:for msg sender name name= <CMP server DN>...

CMP error: error validating protection

Solution: Configure OCCM trust store with the corresponding CA server certificate or chain.

4.1.3 TLS Related Issues

This section describes the following TLS related issues and their resolution steps:

- Hostname validation failed
- Certificates configured in TLS TrustStore do not provide a valid trust anchor to authenticate server identity

Hostname validation failed

Error: Hostname validation failed. The TLS server certificate presented does not have the expected server URL IP. This is causing the following errors:

CMP:apps/cmp.c:2088:CMP info: will contact https://<CA server Alias> CMP DEBUG: Starting new transaction with ID=3B:C4:18:32:75:E5:E5:C2:18:B6:5A:52:E4:AD:D2:93 CMP info: sending IR CMP DEBUG: connecting to CMP server <server IP> using TLS CMP DEBUG: disconnected from CMP server CMP error: certificate verification failed:Certificate verification at depth = 0 error = 64 (IP address mismatch)

CMP:apps/cmp.c:2088:CMP info: will contact https://<CA server Alias> CMP DEBUG: Starting new transaction with ID=<Transaction ID> CMP info: sending IR CMP DEBUG: connecting to CMP server <server IP> using TLS CMP DEBUG: disconnected from CMP server CMP error: certificate verification failed:Certificate verification at depth = 0 error = 64 (IP address mismatch)

Expected IP address = <IP> Failure for: certificate

CMP error: certificate verify failed CMP error: error sending CMP error: transfer error:request sent: IR, expected response: IP

CMP error: certificate verify failed CMP error: error sending CMP error: transfer error:request sent: IR, expected response: IP

Solution: Verifiy the TLS server certificate. One possibilty could be that the certificate has DNS name instead of IP address. In that case pass DNS in the server URL of issuer.

Certificates configured in TLS TrustStore do not provide a valid trust anchor to authenticate server identity

Error: Certificates configured in TLS TrustStore do not provide a valid trust anchor to authenticate server identity. This is causing the following errors:

CMP info: sending IR CMP DEBUG: connecting to CMP CASERVER:8446 using TLS CMP DEBUG: disconnected from CMP server CMP error: certificate verification failed:Certificate verification at depth = 1 error = 19 (self-signed certificate in certificate chain) Failure for: certificate XXXXXX

CMP error: certificate verify failed CMP error: error sending CMP error: transfer error:request sent: IR, expected response: IP

Solution: Configure the TLS TrustStore configuration under issuer with valid trust anchor.

4.2 Miscellaneous Issues

This section describes the following miscellaneous issues and their resolution steps:

Stop infinite certificate request retries

Problem: How to stop infinite certificate request retries.

Solution: Delete the certificate configuration and recreate it by following the procedure mentioned in the *Oracle Communications Cloud Native Core, Certificate Management User Guide.*

Incorrect certificate expiry details

Problem: The certificate expiry details indicated in Grafana dashboard does not match with validity period of the certificate configured in corresponding Kubernetes secrets.

Solution: Check if the Kubernetes secret filled against the certificate configuration is same as what is configured in the NF for that particular interface.

Certificate is not renewed on time

Problem: Certificate is not getting renewed on time.

Solution:

1. Check if the Kubernetes secret filled against the certificate configuration is same as what is configured in the NF for that particular interface
2. Check if the certificates are manually filled in. If yes, initiate certificate recreation. Except for the initial integration with NF certificates, OCCM can only manage certificates created by it. OCCM does not support manual update of the certificates being monitored.

Failed certificate creation

Problem: Certificate creation has failed.

Solution: Certificate creation could fail due to various reasons, OCCM metrics, alert and logs can be used to identify the root cause.

Failed Certificate Renewal

Problem: Certificate renewal has failed.

Solution:

- Check CA connection alerts and metrics.
- Check if the current certificate being renewed is deleted.
- Check if the current certificate is already expired in which case OCCM creates a critical alert indicating the same and stop retrying of renewal. Operator needs to initiate certificate recreation.

Critical certificate expiry alert while integrating with NFs

Problem: Critical alert indicating certificate expiry is raised on integrating with NFs.

Solution: Check if the current certificate is already expired in which case OCCM creates a critical alert. Operator needs to initiate certificate recreation. (Holds true for both NF and OCCM certificate)

Certificate(s) not created for integrated NFs

Problem: Certificate(s) is not created for integrated NFs.

Solution:

1. Check the certificate configuration. Ensure that LCM type Automatic is selected for creation in the certificate configuration. If Manual is selected, OCCM expects that the certificate is already present.
2. Check if the Kubernetes secret filled against the certificate configuration is same as what is configured in the NF for that particular interface.

OCCM Certificate Expires When NF Certificate is About to Renew

Problem: The NF certificate is about to renew using the OCCM certificate as the signer certificate, that is, the Helm parameter `useKurOldCertMode` is set to true when the OCCM certificate expires.

Solution: OCCM will attempt retries for NF certificate renewal, waiting for the OCCM certificate to be in ready state. As soon as the OCCM certificate is ready, the NF certificate will use it and renewal will succeed. To manually create OCCM certificate and get it onboarded in OCCM, perform the same steps as in the OCCM certificate is expired scenario.

OCCM Certificate Expires, Secret is Deleted, or Certificate is Revoked

Problem: The certificate expires, secret is deleted, or certificate is revoked.

Solution: Recreate the certificate. For the procedure to recreate certificates, see Recreating Certificates in the *Oracle Communications Cloud Native Core, Certificate Management User Guide.*

No Alert When Certificate in Secret is Manually Updated or Deleted

Problem: The operator doesn't get an alert when certificate in secret is manually updated or deleted.

Solution: Verify that the monitoring service started successfully by checking that the `k8sSecretMonitoring` flag in custom values file.

Run the following command to Verify if the `occm_alerting_rules_promha.yaml` alert file has been applied or not.

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

Automatic Recreation is not Triggered

Problem: Automatic recreation is not triggered when certificate in the secret is manually updated or deleted.

Solution: Verify that the monitoring service started successfully by checking that the `k8sSecretMonitoring` flag in custom values file.

Check if the secret getting modified or deleted is input type or output type. Automatic recreation is triggered for output type secrets when there is mismatch of data in updated certificate with certificate configuration in OCCM. Or, Output certificate secret is deleted.

If output type certificate or secret is modified, then only validity update happens and no recreation happens.

For input type certificate or secret only alert is raised.

Automatic Recreation Fails when OCCM Certificate Secret is Manually Deleted

Problem: Automatic recreation is not triggered when OCCM certificate secret is deleted or modified because input secret and OCCM certificate secret provided in issuer configuration are the same.

Solution: Auto recreation is not supported if the authentication input for OCCM certificate or certificate in **Initial CMP Client(OCCM) Authentication Options** is:

1. missing
2. has an expired certificate configured
3. same secret is configured in the authentication inputs of OCCM (**Initial CMP Client(OCCM) Authentication Options**) and Other Certificate (**CMP Client Authentication Options For Other certificate**).

In this case, operator must manually configure the OCCM certificate in the secret.

Namespace is Deleted Manually Where the Secret was Ceated and Error Code 403 is logged in the Logs

Problem: This could be a due to hierarchical namespace or service account does not have the permission to read the namespace.

Solution:

1. Check the type of namespace and the behavior is fine.
2. Check permission of service account.
3. Provide access to the service account.

Expired Certificate Handling

Problem: When CMP Identity (OCCM) or End Entity(NF) certificate expiry is detected, recreation of certificate will be attempted.

In case, the recreation of certificate fails then critical alert will be raised mentioning certificate is expired. If the recreation is successful, then certificate validity is updated.

Solution: Perform the following steps if certificates are expired and recreate fails:

For End-Entity (NF) certificates:

1. Check the logs to identify the root cause. The possible cause could be CA connection failure.
2. As a resolution, recreate the certificate if CA is accessible. Alert will be cleared once recreation is successful.
3. If CA is still down then create End-Entity (NF) certificate manually and update details in secret, which is automatically monitored by OCCM.

For CMP Identity (OCCM) certificates:

1. Check the logs to identify the root cause. The possible cause could be CA connection failure. In this case the operator must manually configure the CMP Identity certificate.
2. Get the Kubernetes secret name corresponding to OCCM key and certificate location from the mapped issuer. This information is available under the CMP client authentication options for Other Certification section of the issuer.
3. Create the CMP Identity (OCCM) certificate manually and update the secret.

4. OCCM will start monitoring the certificate and alert will be automatically cleared.

Delay in Monitoring Certificates for Manual Updates

Problem: Delay in Monitoring Certificates for Manual Updates.

When 200 certificate secrets are modified or deleted then alert and certificate recreation will be attempted. The alert can be delayed by maximum of five minutes to generate the first alert and certificate recreation.

When few or one certificate secrets are modified or deleted then alert and certificate recreation will be attempted. The alert can be delayed by maximum of one minutes to generate the first alert and certificate recreation.

Following are the factors that can delay the alerts:

- Network delays
- API server load
- Watch event buffering

CMP Request Rejected as the Certificate Configured in 'CMP client options for OCCM' Certificate has Expired

Problem: The CMP request is rejected when the certificate configured in 'CMP client options for OCCM' certificate has expired. The user gets a warning in the log message. This warning is followed by the "Warning: certificate from '<cert name>' with subject 'CN=xyz' has expired. CMP error: received error:PKIStatus: rejection; PKIFailureInfo: badRequest; StatusString: "certificate verification failed"; errorCode: <errorcode>" error.

Solution: Create a new certificate with a valid expiry date and update it in the corresponding secret provided under the 'CMP client authentication options for OCCM' certificate in the issuer configuration. This certificate is used to sign CMP requests.

Rolling Back OCCM to Previous Helm Release After Editing Certificate Configuration

Problem: Rolling back OCCM to previous helm release after editing the certificate configuration.

Solution: It is not recommended to perform any configurations during the upgrade and rollback window. However, if any of the certificates are edited after the release is upgraded and a subsequently rolled back to the previous version, the edited certificates must be manually recreated after the rollback.

Unable to upgrage Helm When a Namespace from occmAccessedNamespaces is managed by HNC and the Required Role Already Exists

Problem: Helm upgrade may fail when a namespace included in occmAccessedNamespaces is managed by HNC and the required Role already exists. In this case, the user may see the "Unable to continue with update: Role "occm-occm-secret-writer-role" in namespace "ns1" exists and cannot be imported into the current release: invalid ownership metadata; label validation error: key "app.kubernetes.io/managed-by" must equal "Helm": current value is "hnc.x-k8s.io"" error.

Solution: This occurs because the existing Role was not created by Helm and has conflicting ownership metadata. To fix this:

- set `isNamespaceHncManaged` helm parameter to `true` in the custom values file. This ensures that unique Roles and RoleBindings are created for namespaces managed by HNC.

- If creating a custom ServiceAccount, prefix the namespace name to the `metadata.name` of the corresponding Role and RoleBinding.
  For example: `<namespace>-occm-secret-writer-role` or `<namespace>-occm-secret-writer-rolebinding`.

Including additional namespaces in the namespace dropdown for Issuer and Certificate Config Screens

Problem: Including additional namespaces in the namespace dropdown for Issuer and Certificate Configuration Screens

Solution: If `occmAccessedNamespaces` is left empty, the namespace dropdown in the Issuer and Certificate configuration screens will,by default, display only the namespace where occm is deployed. To display additional namespaces, add them to `occmAccessedNamespaces` and perform an upgrade.

4.3 OCCM Error Codes

The following are the types of OCCM error codes and their descriptions:

Table 4-1 Kubernetes Secret Error Codes

Error Code	Description
ERR_MISSING_Kubernetes_SECRET	When the Kubernetes secret is not found for further processing.
ERR_INCORRECT_K8s_SECRET_KEY	When the Kubernetes secret doesn't contain the configured key.
ERR_INVALID_K8S_NAMESPACE_FORMAT	When the format of Kubernetes namespace doesn't comply with the accepted values. Refer installation guide for regex information.
ERR_INVALID_K8S_SECRET_NAME_FORMAT	When the format of Kubernetes secret name doesn't comply with the accepted values. Refer installation guide for regex information.

Table 4-2 Certificate Error Codes

Error Code	Description
ERR_OCCM_CERT_NOT_READY	NF certificate creation fails with ERR_OCCM_CERT_NOT_READY, if OCCM certificate is not yet created.
ERR_MISSING_CERT_TO_BE_RENEWED	When the certificate to be renewed doesn't exist in the Kubernetes secret specified in the configuration.
ERR_RENEW_BEFORE_GREATER_THAN_OR_EQUALS_TO_CERT_ACTUAL_VALIDITY	When the renew before period configured ends up being greater than the certificate validity. The renewal is not triggered in this case.
ERR_INVALID_X509_CERT	When the certificate received from CA is not a valid X.509 certificate.
ERR_UNABLE_TO_RECREATE	When the certificate recreation is unsuccessful.

Table 4-3 Missing mandatory fields in configuration

Error Code	Description
ERR_MISSING_MANDATORY_FIELDS_IN_CMP_PROTECTION_SECRET_MAC	Either name, namespace, password key or reference key is not provided while configuring MAC secret.
ERR_MISSING_MANDATORY_FIELDS_IN_CMP_PROTECTION_SECRET_SIGNATURE	Either name, namespace, key or certificate is not provided while configuring Sign secret.
ERR_MISSING_MANDATORY_FIELDS_IN_PRIVATE_KEY_OUTPUT_LOCATION_CONFIG	Missing fields in private key secret output location in certificate configuration.
ERR_MISSING_MANDATORY_FIELDS_IN_CERT_OUTPUT_LOCATION_CONFIG	Missing fields in certificate secret output location in certificate configuration.
ERR_MISSING_MANDATORY_FIELDS_IN_CERT_CHAIN_OUTPUT_LOCATION_CONFIG	Missing fields in certificate chain secret output location in certificate configuration. Either all fields should be provided or none.
ERR_MISSING_MANDATORY_FIELDS_IN_CA_BUNDLE_SECRET	Missing fields in CA bundle input secret in certificate configuration. Either all fields should be provided or none.

Table 4-4 CMP Error Codes

Error Code	Description
ERR_CMP_COMMAND_FAILED	When CMP command execution fails. It can be during key pair generation, CSR creation or CA interaction.
ERR_CMP_COMMAND_TIMEOUT	When CMP command execution doesn't complete within the configured time.

Table 4-5 Private Key Error Codes

Error Code	Description
ERR_MISSING_KEY_ALGO	When key algorithm is not provided for private key generation.
ERR_MISSING_KEY_SIZE	When key size is not provided for RSA key.
ERR_MISSING_ECCURVE	When EcCurve is not provided for EC key.

Table 4-6 Invalid Input Error Codes

Error Code	Description
ERR_INVALID_DN	When the format of DN (recipientDN or issuerDN) doesn't comply with the accepted values. Refer to *Oracle Communications Cloud Native Core Certificate Management Installation, Upgrade, and Fault Recovery Guide* for regex information.
ERR_INVALID_IP	When the IP configured in SAN is not valid.
ERR_INVALID_DNS	When the format of DNS (configured in SAN) doesn't comply with the accepted values. Refer installation guide for regex information.
ERR_INVALID_URIIDURN	When the format of URN (configured in SAN) doesn't comply with the accepted values. Refer installation guide for regex information.

Table 4-6 (Cont.) Invalid Input Error Codes

Error Code	Description
ERR_SYNTAX_ERROR_IN_URI	When the URI (configured in server URL and in SAN) has synatx error.
ERR_INVALID_NAMESPACE	When the namespace is not included in the accessed namespace list.

Table 4-7 Miscellaneous Codes

Error Code	Description
ERR_MAX_CERT_LIMIT_REACHED	When the total number of certificate created exceeds the configured limit.
ERR_MAX_ISSUER_LIMIT_REACHED	When the total number of issuers created exceeds the configured limit.
ERR_MAX_NS_LIMIT_REACHED	When the total number of namespaces configured exceeds the configured limit.
ERR_CERT_NOT_FOUND	When the certificate does not exist for further processing.
ERR_ISSUER_NOT_FOUND	When the issuer does not exist for further processing.
ERR_ISSUER_IN_USE	When the delete or edit are requested for issuer which is referenced by at least one certificate.
ERR_BULK_CERT_MIGRATION_LIMIT_EXCEEDED	When the total number of bulk certificate migrations exceeds the configured limit.

Table 4-8 TLS Codes

Error Code	Description
ERR_INVALID_SERVER_URL_SCHEME	When TLS is enabled and server URL scheme is other than HTTPS. OR When the TLS is disabled but the server URL scheme is HTTPS.
ERR_MISSING_TLS_TRUST_STORE_DATA	When TLS is enabled but TLS TrustStore is not provided to validate the server certificate.

Table 4-9 Recreate Codes

Error Code	Description
ERR_CERT_ID_MISMATCH	The uuid in the request parameter is either blank or does not match the uuid in the certificate configuration.
ERR_RE_CREATE_REQUEST_EXISTS	A recreate request has already been received for the uuid.
ERR_CERT_NOT_IN_READY_OR_EXPIRED_STATE	The recreate certificate request can't be processed because the certificate is not in either ready or expired status.
ERR_CERT_CONFIG_NOT_EXISTS	The certificate configuration does not exist in the certificate configmap.

Table 4-9 (Cont.) Recreate Codes

Error Code	Description
ERR_CERT_RE_CREATE_REQUEST_IN_PROCESS_RENEW_DELAYED	The renew request is delayed because a certificate recreation request is in progress.
ERR_K8S_WATCHER_ERROR	Certificate secret monitoring fails.

Table 4-10 Bulk Certificate Migrations Codes

Error Code	Description
ERR_BULK_MIGRATE_ALREADY_IN_PROCESS	New migration cannot be triggered or deleted when a bulk certificate migration is already inprocess.
ERR_BULK_CERT_MIGRATION_CONFIG_NOT_FOUND	Bulk certificate migration configuration can not be found. It might have been deleted or not created.
ERR_CERT_NOT_READY_FOR_BULK_MIGRATE	None of the certificates linked to the source issuer are either ready or in the expired state to proceed with bulk certificate migration.

Table 4-11 Edit Certificate Error Codes

Error Code	Description
ERR_CERT_EDIT_ALREADY_IN_PROCESS	When the previously triggered edit is not yet completed, new edit request will not be accepted.
ERR_INVALID_CERT_EDIT_CONFIG	When invalid certificate configuration is sent for edit request.

5
OCCM Alerts

This section describes the alerts available for OCCM.

> **Note**
>
> Alert file is packaged with OCCM CSAR package.
>
> - Review the occm_alerting_rules_promha_<version>.yaml file and edit the value of the parameters in the occm_alerting_rules_promha_<version>.yaml file (if needed to be changed from default values) before configuring the alerts. See above table for details.
> - kubernetes_namespace is configured as kubernetes namespace in which OCCM is deployed. Default value is occm. Please update the occm_alerting_rules_promha_<version>.yaml file to reflect the correct OCCM kubernetes namespace.

Table 5-1 Alerts Levels or Severity Types

Alerts Levels / Severity Types	Definition
Critical	Indicates a severe issue that poses a significant risk to safety, security, or operational integrity. It requires immediate response to address the situation and prevent serious consequences. Raised for conditions may affect the service of OCCM.
Major	Indicates a more significant issue that has an impact on operations or poses a moderate risk. It requires prompt attention and action to mitigate potential escalation. Raised for conditions may affect the service of OCCM.
Minor	Indicates a situation that is low in severity and does not pose an immediate risk to safety, security, or operations. It requires attention but does not demand urgent action. Raised for conditions may affect the service of OCCM.
Info or Warn (Informational)	Provides general information or updates that are not related to immediate risks or actions. These alerts are for awareness and do not typically require any specific response. WARN and INFO alerts may not impact the service of OCCM.

5.1 OccmCmpIdentityCertExpirationMinor

Table 5-2 OccmCmpIdentityCertExpirationMinor

Field	Details
Description	CMP Identity (OCCM) certificate has expired. The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} will expire within 90 days.

Table 5-2 (Cont.) OccmCmpIdentityCertExpirationMinor

Field	Details
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} will expire soon within 90 days'
Severity	Minor
Condition	The CMP Identity (OCCM) certificate will expire within 90 days.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7001
Metric Used	occm_cmp_identity_cert_expiration_seconds
Recommended Actions	Information that certificate is going to expire within 90 days. The alert is cleared when the certificate is renewed so that the certificate expiry day is below the minor threshold or when the certificate expiry day crosses the major threshold, in this case the alert is raised. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Check the certificate configuration to renew before the expiry day. 2. If this is unexpected, contact My Oracle Support.

5.2 OccmCmpIdentityCertExpirationMajor

Table 5-3 OccmCmpIdentityCertExpirationMajor

Field	Details
Description	CMP Identity (OCCM) certificate has expired. The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} will expire within 30 days.
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} will expire soon within 30 days'
Severity	Major
Condition	The CMP Identity (OCCM) certificate will expire within 30 days.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7001
Metric Used	occm_cmp_identity_cert_expiration_seconds

Table 5-3 (Cont.) OccmCmpIdentityCertExpirationMajor

Field	Details
Recommended Actions	Information that certificate is going to expire within 30 days. The alert is cleared when the certificate is renewed or when the certificate expiry days crosses the critical threshold, in which case the alert is raised. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Check the certificate configuration to renew before the expiry day. 2. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to the thread exceptions. 3. Perform the resolution steps depending on the failure reason. 4. If this is unexpected, contact My Oracle Support.

5.3 OccmCmpIdentityCertExpirationCritical

Table 5-4 OccmCmpIdentityCertExpirationCritical

Field	Details
Description	CMP Identity (OCCM) certificate has expired. The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} will expire within one week.
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} will expire soon within 1 week'
Severity	Critical
Condition	The CMP Identity (OCCM) certificate will expire within one week.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7001
Metric Used	occm_cmp_identity_cert_expiration_seconds
Recommended Actions	Information that Certificate is going to expire within one week. The alert is cleared when the certificate is renewed. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Check the certificate configuration to renew before the expiry day. 2. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to the thread exceptions. 3. Perform the resolution steps depending on the failure reason. 4. If this is unexpected, contact My Oracle Support.

5.4 OccmCmpIdentityCertExpired

Table 5-5 OccmCmpIdentityCertExpired

Field	Details
Description	Alert is raised when the certificate expires and then recreation will be triggered. If the certificate recreation is successful then alert will be cleared automatically or the operator has to clear the alert manually. The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} is expired.
Summary	'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} is expired'
Severity	Critical
Condition	The CMP Identity (OCCM) certificate has expired.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7002
Metric Used	occm_cmp_identity_cert_expiration_seconds
Recommended Actions	Information that the certificate has expired. The alert is cleared when the certificate is recreated. Steps: 1. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to the thread exceptions. 2. Perform the following steps if recreate fails and certificates are expired: a. Check logs to identify the root cause. The possible cause may be CA connection failure. In this case operator must manually configure the CMP Identity certificate. b. Get the kubernetes secret name corresponding to OCCM key and certificate location from the mapped issuer. This information is present under CMP client authentication options for Other Cert section of the issuer. c. Manually create CMP Identity (OCCM) certificate and update the secret. d. Manual recreation of certificate can be triggered when CA connection resumes. e. To renew expired certificate, see "Expired Certificate Detection" section in *Oracle Communications Cloud Native Core, Certificate Management Troubleshooting Guide.* 3. If this is unexpected, contact My Oracle Support.

5.5 OccmEndEntityCertExpirationMinor

Table 5-6 OccmEndEntityCertExpirationMinor

Field	Details
Description	End Entity (NF) certificate has expired. The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} will expire within 90 days.
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} will expire soon within 90 days'
Severity	Minor
Condition	The End Entity (NF) certificate will expire within 90 days.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7003
Metric Used	occm_end_entity_cert_expiration_seconds
Recommended Actions	Information that certificate is going to expire within 90 days. The alert is cleared when the certificate is renewed so that the certificate expiry day is below the minor threshold or when the certificate expiry day crosses the major threshold, in this case the alert is raised. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Check the certificate configuration to renew before the expiry day. 2. If this is unexpected, contact My Oracle Support.

5.6 OccmEndEntityCertExpirationMajor

Table 5-7 OccmEndEntityCertExpirationMajor

Field	Details
Description	End Entity (NF) certificate has expired. The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} will expire within 30 days.
Summary	'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} will expire soon within 30 days.
Severity	Major
Condition	End Entity (NF) certificate will expire soon within 30 days.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7003
Metric Used	occm_end_entity_cert_expiration_seconds

Table 5-7 (Cont.) OccmEndEntityCertExpirationMajor

Field	Details
Recommended Actions	Information that Certificate is going to expire within 30 days. The alert is cleared when the certificate is renewed or when the certificate expiry day crosses the critical threshold,, in this case the alert is raised. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Check the certificate configuration to renew before the expiry day. 2. Refer to the application logs on Kibana and filter based on occm service names. Check for ERROR WARNING logs related to thread exceptions. 3. Perform the resolution steps depending on the failure reason. 4. If this is unexpected, contact My Oracle Support.

5.7 OccmEndEntityCertExpirationCritical

Table 5-8 OccmEndEntityCertExpirationCritical

Field	Details
Description	End Entity (NF) certificate has expired. The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} will expire within one week.
Summary	'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} will expire soon within 1 week'
Severity	Critical
Condition	End Entity (NF) certificate will expire soon within one week.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7003
Metric Used	occm_end_entity_cert_expiration_seconds
Recommended Actions	Information that Certificate is going to expire within one week. The alert is cleared when the certificate is renewed. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Check the certificate configuration to renew before the expiry day. 2. Refer to the application logs on Kibana and filter based on occm service names. Check for ERROR WARNING logs related to thread exceptions. 3. Perform the resolution steps depending on the failure reason. 4. If this is unexpected, contact My Oracle Support.

5.8 OccmEndEntityCertExpired

Table 5-9 OccmEndEntityCertExpired

Field	Details
Description	Alert is raised when the certificate expires and then recreation will be triggered. If the certificate recreation is successful then alert will be cleared automatically or the operator has to clear the alert manually. The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} is expired'
Summary	'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} for {{$labels.certPurpose}} is expired'
Severity	Critical
Condition	End Entity (NF) certificate has expired.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7009
Metric Used	occm_end_entity_cert_expiration_seconds
Recommended Actions	Information that certificate has expired. The alert is cleared when the certificate is recreated. Steps: 1. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to the thread exceptions. 2. Perform the following steps if recreate fails and certificates are expired: a. Check logs to identify the root cause. The possible cause may be CA connection failure. b. As a resolution perform the recreate operation when is CA is accessible. Alert will be cleared once recreation is successful. c. If CA is still down then manually create the End-Entity (NF) certificate and update the details in secret, which is automatically monitored by OCCM. d. Manual recreation of certificate can be triggered when CA connection resumes. e. To renew expired certificate, see "Expired Certificate Detection" section in *Oracle Communications Cloud Native Core, Certificate Management Troubleshooting Guide.*. 3. If this is unexpected, contact My Oracle Support.

5.9 OccmServiceDown

Table 5-10 OccmServiceDown

Field	Details
Description	OCCM Service Down Alert New certificates will not be created, and existing ones can not be renewed until OCCM is back
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: OCCM service is down
Severity	Critical
Condition	The pods of the occm service is unavailable.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7004
Metric Used	up **Note:**This is a prometheus metric used for instance availability monitoring. If this metric is not available, use the similar metric as exposed by the monitoring system.
Recommended Actions	The alert is cleared when the occm service is available. Steps: 1. Check the orchestration logs of occm service and check for liveness or readiness probe failures. 2. Refer to the application logs on Kibana and filter based on occm service names. Check for ERROR WARNING logs related to thread exceptions. 3. Depending on the failure reason, take the resolution steps. 4. In case the issue persists, contact My Oracle Support.

5.10 OccmMemoryUsageMinorThreshold

Table 5-11 OccmMemoryUsageMinorThreshold

Field	Details
Description	OCCM Memory Usage Alert OCCM Memory Usage for pod {{ $labels.pod }} has crossed the configured minor threshold (70%) (value={{ $value }}) of its limit.
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Memory Usage of pod exceeded 70% of its limit.
Severity	Minor
Condition	A pod has reached the configured minor threshold(70%) of its memory resource limits.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7005

Table 5-11 (Cont.) OccmMemoryUsageMinorThreshold

Field	Details
Metric Used	container_memory_usage_bytes, **Note** : This is a kubernetes metric used for instance availability monitoring. If the metric is not available, use the similar metric as exposed by the monitoring system.
Recommended Actions	The alert gets cleared when the memory utilization falls below the Minor Threshold or crosses the major threshold, in which case OccmMemoryUsageMajorThreshold alert shall be raised. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to thread exceptions. 2. Depending on the failure reason, take the resolution steps. 3. If this is unexpected, contact My Oracle Support.

5.11 OccmMemoryUsageMajorThreshold

Table 5-12 OccmMemoryUsageMajorThreshold

Field	Details
Description	OCCM Memory Usage Alert OCCM Memory Usage for pod {{ $labels.pod }} has crossed the configured major threshold (80%) (value={{ $value }}) of its limit.
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Memory Usage of pod exceeded 80% of its limit.
Severity	Major
Condition	A pod has reached the configured major threshold(80%) of its memory resource limits.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7005
Metric Used	container_memory_usage_bytes, **Note** : This is a kubernetes metric used for instance availability monitoring.If the metric is not available, use the similar metric as exposed by the monitoring system.

Table 5-12 (Cont.) OccmMemoryUsageMajorThreshold

Field	Details
Recommended Actions	The alert gets cleared when the memory utilization falls below the Major Threshold or crosses the critical threshold, in which case OccmMemoryUsageMajorThreshold alert shall be raised Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to thread exceptions. 2. Depending on the failure reason, take the resolution steps. 3. If this is unexpected, contact My Oracle Support.

5.12 OccmMemoryUsageCriticalThreshold

Table 5-13 OccmMemoryUsageCriticalThreshold

Field	Details
Description	OCCM Memory Usage Alert OCCM Memory Usage for pod {{ $labels.pod }} has crossed the configured critical threshold (90%) (value={{ $value }}) of its limit..
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Memory Usage of pod exceeded 90% of its limit.
Severity	Critical
Condition	A pod has reached the configured critical threshold (90%) of its memory resource limits
OID	1.3.6.1.4.1.323.5.3.54.1.2.7005
Metric Used	container_memory_usage_bytes, Note : This is a kubernetes metric used for instance availability monitoring.If the metric is not available, use the similar metric as exposed by the monitoring system.
Recommended Actions	The alert gets cleared when the memory utilization falls below the Critical Threshold.Note : The threshold is configurable in the alerts.yaml Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to thread exceptions. 2. Depending on the failure reason, take the resolution steps. 3. If this is unexpected, contact My Oracle Support.

5.13 OccmCPUUsageMinorThreshold

Table 5-14 OccmCPUUsageMinorThreshold

Field	Details
Description	OCCM CPU Usage Alert OCCM Pod {{$labels.pod}} has high CPU usage detected.
Summary	namespace: {{ $labels.namespace}}, podname: {{ $labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: CPU usage is {{ $value | printf "%.2f" }} which is usage is above 70% (current value is: {{ $value }})
Severity	Minor
Condition	CPU usage is above 70%
OID	1.3.6.1.4.1.323.5.3.54.1.2.7006
Metric Used	container_cpu_usage_seconds_total
Recommended Actions	Information regarding CPU usage If it is above 70% The alert gets cleared when the CPU usage falls below the Minor Threshold. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to thread exceptions. 2. Depending on the failure reason, take the resolution steps. 3. If this is unexpected, contact My Oracle Support.

5.14 OccmCMPFailureMinor

Table 5-15 OccmCMPFailureMinor

Field	Details
Description	OCCM CMP Command Execution Failure Alert The certificate {{$labels.certName}} used by {{$labels.nfType}} has failed while executing CMP cmd with {{$labels.statusCode}}.
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} has failed while executing CMP cmd with {{$labels.statusCode}}.
Severity	Minor
Condition	Certificate has failed while executing CMP cmds.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7007
Metric Used	occm_cmp_responses_total

Table 5-15 (Cont.) OccmCMPFailureMinor

Field	Details
Recommended Actions	Information that the rate of certificate failure due to CMP command execution error has crossed the threshold. The alert is cleared when the rate of certificate failure due to CMP command execution error falls below the Minor threshold or when the error rate crosses the Major threshold, in which case the OccmCMPFailureMajor alert is raised. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to thread exceptions. 2. Depending on the failure reason, take the resolution steps. 3. If this is unexpected, contact My Oracle Support.

5.15 OccmCMPFailureMajor

Table 5-16 OccmCMPFailureMajor

Field	Details
Description	OCCM CMP Command Execution Failure Alert The certificate {{$labels.certName}} used by {{$labels.nfType}} has failed while executing CMP cmd with {{$labels.statusCode}}.
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} has failed while executing CMP cmd with {{$labels.statusCode}}.
Severity	Major
Condition	Certificate has failed while executing CMP cmds
OID	1.3.6.1.4.1.323.5.3.54.1.2.7007
Metric Used	occm_cmp_responses_total
Recommended Actions	Information that the rate of certificate failure due to CMP command execution error has crossed the threshold. The alert is cleared when the rate of certificate failure due to CMP command execution error falls below the Major threshold or when the error rate crosses the Critical threshold, in which case the OccmCMPFailureCritical alert is raised. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to thread exceptions. 2. Depending on the failure reason, take the resolution steps. 3. If this is unexpected, contact My Oracle Support.

5.16 OccmCMPFailureCritical

Table 5-17 OccmCMPFailureCritical

Field	Details
Description	OCCM CMP Command Execution Failure Alert The certificate {{$labels.certName}} used by {{$labels.nfType}} has failed while executing CMP cmd with {{$labels.statusCode}}.
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} has failed while executing CMP cmd with {{$labels.statusCode}}.
Severity	Critical
Condition	Certificate has failed while executing CMP cmds
OID	1.3.6.1.4.1.323.5.3.54.1.2.7007
Metric Used	occm_cmp_responses_total
Recommended Actions	Information that the rate of certificate failure due to CMP command execution error has crossed the threshold. The alert is cleared when the rate of certificate failure due to CMP command execution error falls below the Critical threshold. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to thread exceptions. 2. Depending on the failure reason, take the resolution steps. 3. If this is unexpected, contact My Oracle Support.

5.17 OccmFailureMinor

Table 5-18 OccmFailureMinor

Field	Details
Description	OCCM Internal Failure Alert The certificate {{$labels.certName}} used by {{$labels.nfType}} has failed while creating cert with {{$labels.errorReason}}.
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} has failed while creating cert with {{$labels.errorReason}}.
Severity	Minor
Condition	Certificate has failed while creating
OID	1.3.6.1.4.1.323.5.3.54.1.2.7008
Metric Used	occm_cert_request_status_total

Table 5-18 (Cont.) OccmFailureMinor

Field	Details
Recommended Actions	Information that the rate of OCCM errors has crossed the threshold. The alert is cleared when the rate OCCM error falls below the Minor threshold or when the error rate crosses the Major threshold, in which case the OccmFailureMajor alert is raised. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to thread exceptions. 2. Depending on the failure reason, take the resolution steps. 3. If this is unexpected, contact My Oracle Support.

5.18 OccmFailureMajor

Table 5-19 OccmFailureMajor

Field	Details
Description	OCCM Internal Failure Alert The certificate {{$labels.certName}} used by {{$labels.nfType}} has failed while creating cert with {{$labels.errorReason}}.
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} has failed while creating cert with {{$labels.errorReason}}.
Severity	Major
Condition	Certificate has failed while creating
OID	1.3.6.1.4.1.323.5.3.54.1.2.7008
Metric Used	occm_cert_request_status_total
Recommended Actions	Information that the rate of OCCM errors has crossed the threshold. The alert is cleared when the rate OCCM error falls below the Major threshold or when the error rate crosses the Critical threshold, in which case the OccmFailureCritical alert is raised. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to thread exceptions. 2. Depending on the failure reason, take the resolution steps. 3. If this is unexpected, contact My Oracle Support.

5.19 OccmFailureCritical

Table 5-20 OccmFailureCritical

Field	Details
Description	OCCM CMP Command Execution Failure Alert The certificate {{$labels.certName}} used by {{$labels.nfType}} has failed while creating cert with {{$labels.errorReason}}.
Summary	namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The certificate {{$labels.certName}} used by {{$labels.nfType}} has failed while creating cert with {{$labels.errorReason}}.
Severity	critical
Condition	Certificate has failed while creating
OID	1.3.6.1.4.1.323.5.3.54.1.2.7008
Metric Used	occm_cert_request_status_total
Recommended Actions	Information that the rate of certificate failure due to CMP command execution error has crossed the threshold. The alert is cleared when the rate of certificate failure due to CMP command execution error falls below the Critical threshold. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to thread exceptions. 2. Depending on the failure reason, take the resolution steps. 3. If this is unexpected, contact My Oracle Support.

5.20 OccmInputSecretModifyMajor

Table 5-21 OccmInputSecretModifyMajor

Field	Details
Description	**Input secret is modified by non-OCCM user** The Secret {{$labels.secret}} in {{$labels.secretNamespace}} is modified by non-occm user, which is used by {{$labels.name}}.'
Summary	'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The Secret {{$labels.secret}} in {{$labels.secretNamespace}} is modified by non-occm user, which is used by {{$labels.name}} and {{$labels.type}}.'
Severity	Major
Condition	Input secrets are modified by non-OCCM users or by the operator manually.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7010
Metric Used	occm_secret_event_total

Table 5-21 (Cont.) OccmInputSecretModifyMajor

Field	Details
Recommended Actions	Information that the input secret is modified by non-OCCM user. Steps: 1. Check input secrets for any modifications. 2. See the alert label for the namespace and to see which secret alert is triggered. 3. Update input secrets with correct data, if require. 4. If this is unexpected, contact My Oracle Support.

5.21 OccmOutputSecretModifyMinor

Table 5-22 OccmOutputSecretModifyMinor

Field	Details
Description	**Output secret is modified by non-OCCM user** The Secret {{$labels.secret}} in {{$labels.secretNamespace}} is modified by non-occm user, which is used by {{$labels.name}}.'
Summary	'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The Secret {{$labels.secret}} in {{$labels.secretNamespace}} is modified by non-occm user, which is used by {{$labels.name}} and {{$labels.type}}.'
Severity	Minor
Condition	Output secrets are modified by non-OCCM user or by operator manually
OID	1.3.6.1.4.1.323.5.3.54.1.2.7011
Metric Used	occm_secret_event_total
Recommended Actions	Information that the output secret is modified by non-OCCM user. Steps: 1. Check output secrets for any modifications. 2. Automatic recreation will be triggered if certificate which is modified does not match with cert config. 3. Updatation of validity will be done, if the modified certificate validation is successful with certification configuration. No recreation will be triggered in this case. 4. If this is unexpected, contact My Oracle Support.

5.22 OccmK8sResourceDeleteMajor

Table 5-23 OccmK8sResourceDeleteMajor

Field	Details
Description	**Kubernetes resource (secret or namespace) is deleted by non-OCCM user** The Kubernetes resource is deleted, which is used in {{$labels.name}} of type {{$labels.type}}. K8s resources, secretNamespace: {{$labels.secretNamespace}} and secret: {{$labels.secret}}'
Summary	{{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: The k8s resource is deleted, which is used in {{$labels.name}} of type {{$labels.type}}. K8s resources, namespace: {{$labels.secretNamespace}} and secret: {{$labels.secret}}.'
Severity	Major
Condition	Kubernetes resources (secret or namespace) are deleted by non-OCCM user or by operator manually.
OID	1.3.6.1.4.1.323.5.3.54.1.2.7012
Metric Used	occm_secret_event_status
Recommended Actions	Information that the Kubernetes resources (secret or namespace) are deleted by non-OCCM user. Steps: 1. Check output secrets for any deletion. 2. Automatic recreation of certificate will be triggered, if secret is deleted. 3. if namespace is deleted, then automatic recreation of certificate does not happen and the operator must delete the certificate configuration from the OCCM which are associated with that namespace. 4. If this is unexpected, contact My Oracle Support.

5.23 OCCM Alert and MIB Configuration in Prometheus

CNE supporting Prometheus HA

This section describes the measurement based Alert rules configuration for OCCM in Prometheus. You must use the updated `occm_alerting_rules_promha_<version>.yaml file`.

Run the following command to create ot update the PrometheusRule resource specified in the alert YAML file:

```
$ kubectl apply -f occm_alerting_rules_promha_<version>.yaml
```

Disabling Alerts

This section describes the procedure to disable the alerts in OCCM. To disable alerts:

1. Edit `occm_alerting_rules_promha_<version>.yaml` file to remove specific alert.

2. Remove complete content of the specific alert from the `occm_alerting_rules_promha_<version>.yaml` file.
   For example, ff you want to remove OccmServiceDown alert, remove the complete content:

```
## ALERT SAMPLE START##
- alert: OccmServiceDown
      annotations:
        description: 'New certificates will not be created, and existing
ones can not be renewed until OCCM is back'
        summary: 'namespace: {{$labels.namespace}}, podname:
{{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value
| humanizeTimestamp }}{{ end }}: OCCM service is down'
      expr: absent(up{pod=~".*occm.*", namespace="occm-ns"}) or
(up{pod=~".*occm.*", namespace="occm-ns"}) == 0
      labels:
        severity: critical
        oid: "1.3.6.1.4.1.323.5.3.54.1.2.7004"
        namespace: ' {{ $labels.namespace }} '
        podname: ' {{$labels.pod}} '
## ALERT SAMPLE END##
```

3. Perform Alert configuration.

Validating Alerts

Configure and Validate Alerts in Prometheus Server. Refer to OCCM Alert Configuration for procedure to configure the alerts.

After configuring the alerts in Prometheus server, a user can verify that by following steps:

1. Open the Prometheus server from your browser using the <IP>:<Port>
2. Navigate to Status and then Rules
3. Search OCCM. OCCMAlerts list is displayed.

> **Note**
>
> If you are unable to see the alerts, it means that the alert file has not loaded in a format which the Prometheus server accepts. Modify the file and try again.

Configuring SNMP-Notifier

Configure the IP and port of the SNMP trap receiver in the SNMP Notifier using following procedure:

1. Run the following command to edit the deployment:

   ```
   kubectl edit deploy <snmp_notifier_deployment_name> -n <namespace>
   ```

   Example:

   ```
   $ kubectl edit deploy occne-snmp-notifier -n occne-infra
   ```

2. Edit the destination as follows:

   ```
   --snmp.destination=<destination_ip>:<destination_port>
   ```

   Example:

   ```
   --snmp.destination=10.75.203.94:162
   ```

MIB Files for OCCM

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.

- `occm_mib_tc_<version>.mib`: This is considered as OCCM top level mib file, where the Objects and their data types are defined
- `occm_mib_<version>.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 along with OCCM CSAR package. Download the file from MOS. For more information, see *Oracle Communications Cloud Native Core, Certificate Management Installation, Upgrade, and Fault Recovery Guide.*