

Oracle® Communications

Cloud Native Core, Certificate Management Troubleshooting Guide



Release 24.1.0
F90674-01
April 2024

ORACLE®

Copyright © 2023, 2024, Oracle and/or its affiliates.

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

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

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

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

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

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

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

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

Contents

1	Introduction	
1.1	Overview	1
1.2	Reference	1
2	Logs	
2.1	Log Levels	1
2.2	Collecting Logs	1
2.3	Understanding Logs	2
2.4	Accessing Logs	3
3	Using Debug Tool	
4	Troubleshooting OCCM	
4.1	Configuration Related Issues	1
4.1.1	Issuer and Certificate Related Errors	1
4.1.2	CMP Related Issues	5
4.1.3	TLS Related Issues	6
4.2	Miscellaneous Issues	7
4.3	OCCM Error Codes	8
5	OCCM Alerts	
5.1	OccmCertExpiryWithinMinorThreshold	1
5.2	OccmCertExpiryWithinMajorThreshold	2
5.3	OccmCertExpiryWithinCriticalThreshold	2
5.4	OccmCertExpired	3
5.5	OccmServiceDown	4
5.6	OccmMemoryUsageMinorThreshold	4
5.7	OccmMemoryUsageMajorThreshold	5
5.8	OccmMemoryUsageCriticalThreshold	6
5.9	OccmCPUUsageMinorThreshold	7

5.10	OccmCMPFailureMinor	7
5.11	OccmCMPFailureMajor	8
5.12	OccmCMPFailureCritical	9
5.13	OccmFailureMinor	9
5.14	OccmFailureMajor	10
5.15	OccmFailureCritical	11
5.16	OccmRenewBeforValidityCritical	11
5.17	OCCM Alert and MIB Configuration in Prometheus	12

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.

What's New in This Guide

This section lists the documentation updates for Release 24.1.x in Oracle Communications Cloud Native Core, Certificate Management Troubleshooting Guide.

Release 24.1.0 - F90674-01, April 2024

- Added the [TLS Related Issues](#) section.
- Added the following troubleshooting scenarios to the [Miscellaneous Issues](#) section:
 - Certificate(s) is not created for integrated NF(s)
 - OCCM certificate is expired
- Updated the [OCCM Error Codes](#) section with TLS Error codes.

Acronyms

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

Table Acronyms and Terminologies

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

1

Introduction

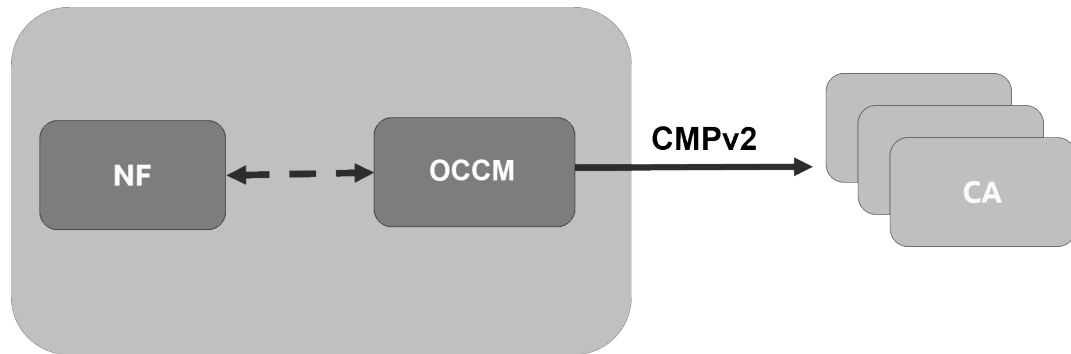
This document provides information about the Oracle Communication Cloud Native Core, Certificate Management troubleshooting scenarios.

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 Configuration Console User Guide*
- *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide*

- *Oracle Communications Cloud Native Core Security Guide*
- *Oracle Communications Cloud Native Core Solution Upgrade 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"

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. Execute 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@master ~]# 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. Execute 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 24.1.0
```

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:

- [Secret Name Format Error](#)
- [Namespace Format Error](#)
- [Secret Key Error](#)
- [Secret Key Error](#)
- [Input String Error](#)
- [Incomplete TrustStore Secret Error](#)
- [Invalid Secret Name](#)
- [Repeated Secret Error](#)
- [Secret Doesn't Exist Error](#)
- [Unique Secret Key Error](#)
- [CA Bundle Secret Error](#)
- [Invalid MAC Secret Error](#)
- [Invalid File Format](#)
- [Delete Issuer Error](#)
- [Issuer ID Error](#)
- [Issuer Already Exists Error](#)

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.

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: Verify the TLS server certificate. One possibility 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 server master: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:

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.

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.

Certificate(s) is not created for integrated NF(s)

Problem: Certificate(s) is not created for integrated NF(s)

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 is expired

Problem: OCCM Certificate gets expired due to any reason.

Solution:

1. Create the OCCM certificate manually using any of the configured issuers in OCCM.
2. Get the Kubernetes secret corresponding to OCCM key/certificate location from the mapped issuer. This information will be present under CMP client authentication options for Other Cert section of the issuer. Update the secret fetched with the manually created certificate from (1).
3. In order to make OCCM aware about the change, delete the existing OCCM cert configuration and create a new one with same configuration and set the CertType as MANUAL.

OCCM will start monitoring the certificate.

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_K8s_SECRET	When the K8s secret is not found for further processing.
ERR_INCORRECT_K8s_SECRET_KEY	When the k8s secret doesn't contain the configured key.
ERR_INVALID_K8S_NAMESPACE_FORMAT	When the format of k8s namespace doesn't comply with the accepted values. Refer installation guide for regex information.
ERR_INVALID_K8S_SECRET_NAME_FORMAT	When the format of k8s 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 cert creation fails with ERR_OCCM_CERT_NOT_READY, if OCCM cert is not yet created.
ERR_MISSING_CERT_TO_BE_RENEWED	When the cert to be renewed doesn't exist in the K8s 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 cert 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.

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 cert 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 installation 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.
ERR_SYNTAX_ERROR_IN_URI	When the URI (configured in server URL and in SAN) has syntax error.

Table 4-7 Miscellaneous Codes

Error Code	Description
ERR_MAX_CERT_LIMIT_REACHED	When the total number of certs created exceeds the configured limit.
ERR_MAX_ISSUER_LIMIT_REACHED	When the total number of issuers created exceeds the configured limit.
ERR_CERT_NOT_FOUND	When the certificate doesn't exist for further processing.
ERR_ISSUER_NOT_FOUND	When the issuer doesn't exist for further processing.
ERR_ISSUER_IN_USE	When the delete or edit is requested for issuer which is referenced by atleast one certificate.

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.

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.

5.1 OccmCertExpiryWithinMinorThreshold

Table 5-1 OccmCertExpiryWithinMinorThreshold

Field	Details
Description	OCCM Certificate Expiry Alert The certificate <code>{{ \$labels.certName }}</code> used by <code>{{ \$labels.nfType }}</code> for <code>{{ \$labels.certPurpose }}</code> will expire soon within 90 days
Summary	namespace: <code>{{ \$labels.namespace }}</code> , podname: <code>{{ \$labels.pod }}</code> , timestamp: <code>{{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}</code> : The certificate <code>{{ \$labels.certName }}</code> used by <code>{{ \$labels.nfType }}</code> for <code>{{ \$labels.certPurpose }}</code> will expire soon within 90 days
Severity	Minor
Condition	Certificate will expire soon within 90 days
OID	1.3.6.1.4.1.323.5.3.54.1.2.7001
Metric Used	<code>occm_cert_expiry</code>
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 days falls below the Minor threshold or when the Certificate expiry days crosses the Major threshold, in which case the <code>OccmCertExpiryWithinMajorThreshold</code> alert is raised. Note: The threshold is configurable in the <code>occm_alertingrules_<version>.yaml</code> file. Steps: 1. Check certificate configuration for renew before days. 2. If this is unexpected, contact My Oracle Support.

5.2 OccmCertExpiryWithinMajorThreshold

Table 5-2 OccmCertExpiryWithinMajorThreshold

Field	Details
Description	OCCM Certificate Expiry Alert The certificate {{labels.certName}} used by {{labels.nfType}} for {{labels.certPurpose}} will expire soon 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	Certificate will expire soon within 30 days
OID	1.3.6.1.4.1.323.5.3.54.1.2.7001
Metric Used	occm_cert_expiry
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 OccmCertExpiryWithinCriticalThreshold alert is raised. Note: The threshold is configurable in the occm_alertingrules_<version>.yaml file. Steps: 1. Check certificate configuration for renew before days. 2. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to thread exceptions. 3. Depending on the failure reason, take the resolution steps. 4. If this is unexpected, contact My Oracle Support.

5.3 OccmCertExpiryWithinCriticalThreshold

Table 5-3 OccmCertExpiryWithinCriticalThreshold

Field	Details
Description	OCCM Certificate Expiry Alert The certificate {{labels.certName}} used by {{labels.nfType}} for {{labels.certPurpose}} will expire soon within 1 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	Certificate will expire soon within 1 week
OID	1.3.6.1.4.1.323.5.3.54.1.2.7001

Table 5-3 (Cont.) OccmCertExpiryWithinCriticalThreshold

Field	Details
Metric Used	occm_cert_expiry
Recommended Actions	Information that Certificate is going to expire within 1 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 certificate configuration for renew before days. 2. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to thread exceptions. 3. Depending on the failure reason, take the resolution steps. 4. If this is unexpected, contact My Oracle Support.

5.4 OccmCertExpired

Table 5-4 OccmCertExpired

Field	Details
Description	OCCM Certificate has Expired Critical Alert 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	Certificate has expired
OID	1.3.6.1.4.1.323.5.3.54.1.2.7002
Metric Used	occm_cert_expiry
Recommended Actions	Information that Certificate has expired. The alert is cleared when the certificate is renewed. 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. Refer user guide for procedure to renew expired certificate. 4. If this is unexpected, contact My Oracle Support.

5.5 OccmServiceDown

Table 5-5 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.

5.6 OccmMemoryUsageMinorThreshold

Table 5-6 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 5-6 (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.

5.7 OccmMemoryUsageMajorThreshold

Table 5-7 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 5-7 (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.

5.8 OccmMemoryUsageCriticalThreshold

Table 5-8 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.

5.9 OccmCPUUsageMinorThreshold

Table 5-9 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.

5.10 OccmCMPFailureMinor

Table 5-10 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	occn_cmp_responses_total

Table 5-10 (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 <code>occm_alertingrules_<version>.yaml</code> file. Steps: 1. Refer to the application logs on Kibana and filter based on <code>occm</code> 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 OccmCMPFailureMajor

Table 5-11 OccmCMPFailureMajor

Field	Details
Description	OCCM CMP Command Execution Failure Alert The certificate <code>{{ \$labels.certName }}</code> used by <code>{{ \$labels.nfType }}</code> has failed while executing CMP cmd with <code>{{ \$labels.statusCode }}</code>.
Summary	namespace: <code>{{ \$labels.namespace }}</code>, podname: <code>{{ \$labels.pod }}</code>, timestamp: <code>{{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}</code>: The certificate <code>{{ \$labels.certName }}</code> used by <code>{{ \$labels.nfType }}</code> has failed while executing CMP cmd with <code>{{ \$labels.statusCode }}</code>.
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	<code>occm_cmp_responses_total</code>
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 <code>occm_alertingrules_<version>.yaml</code> file. Steps: 1. Refer to the application logs on Kibana and filter based on <code>occm</code> 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 OccmCMPFailureCritical

Table 5-12 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.

5.13 OccmFailureMinor

Table 5-13 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_cmp_responses_total

Table 5-13 (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 <code>occm_alertingrules_<version>.yaml</code> 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 OccmFailureMajor

Table 5-14 OccmFailureMajor

Field	Details
Description	OCCM Internal Failure Alert The certificate <code>{{ \$labels.certName }}</code> used by <code>{{ \$labels.nfType }}</code> has failed while creating cert with <code>{{ \$labels.errorReason }}</code> .
Summary	namespace: <code>{{ \$labels.namespace }}</code> , podname: <code>{{ \$labels.pod }}</code> , timestamp: <code>{{ with query "time()" }}{{ . first value humanizeTimestamp }}{{ end }}</code> : The certificate <code>{{ \$labels.certName }}</code> used by <code>{{ \$labels.nfType }}</code> has failed while creating cert with <code>{{ \$labels.errorReason }}</code> .
Severity	Major
Condition	Certificate has failed while creating
OID	1.3.6.1.4.1.323.5.3.54.1.2.7008
Metric Used	<code>occm_cmp_responses_total</code>
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 <code>occm_alertingrules_<version>.yaml</code> 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 OccmFailureCritical

Table 5-15 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_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.

5.16 OccmRenewBeforValidityCritical

Table 5-16 OccmRenewBeforValidityCritical

Field	Details
Description	OCCM Renew Before is greater than Cert Validity Alert The certificate {{labels.certName}} used by {{labels.nfType}} has failed because renew before validity is greater than cert validity {{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 because renew before validity is greater than cert validity {{labels.errorReason}}.
Severity	critical
Condition	Certificate has failed because renew before validity is greater than cert validity
OID	1.3.6.1.4.1.323.5.3.54.1.2.7009

Table 5-16 (Cont.) OccmRenewBeforValidityCritical

Field	Details
Metric Used	occm_cert_request_status_total
Recommended Actions	Information that the Certificate has failed because renew before validity is greater than cert validity. Steps: 1. Check certificate configuration for renew before days. 2. Also Check the validity requested for the Certificate and validity assigned by the CA. 3. Refer to the application logs on Kibana and filter based on occm service name. Check for ERROR WARNING logs related to thread exceptions. 4. Depending on the failure reason, take the resolution steps. 5. If this is unexpected, contact My Oracle Support.

5.17 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 or 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, if 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. OCCMAAlerts 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. Execute 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

here 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*.