Oracle® Communications

Cloud Native Configuration Console Troubleshooting Guide

Release 25.2.100
G35639-02
November 2025

Oracle Communications Cloud Native Configuration Console Troubleshooting Guide, Release 25.2.100

G35639-02

Contents

5 CNC Console Troubleshooting in OCI Deployment

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:

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

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

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

Acronyms

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

Table Acronyms

Acronym	Description
A-CNCC Core	Agent CNC Console is a CNCC Core instance which manages local NF(s) and local OCCNE common services(s). A-CNCC is managed by M-CNCC. A-CNCC contains A-CNCC Core Ingress Gateway. A-CNCC has no IAM component. A-CNCC is also known as A-CNCC Core or aCncc Core.
AD	Active Directory
ASM	Aspen Service Mesh
BSF	Oracle Communications Cloud Native Core, Binding Support Function
CNC Console	Oracle Communications Cloud Native Configuration Console
cnDBTier	Oracle Communications Cloud Native Core, cnDBTier
CNE	Oracle Communications Cloud Native Core, Cloud Native Environment
CNI	Container Network Interface
CNLB	Cloud Native Load Balancer
CRD	Custom Resource Definitions
CRUD Operations	CREATE, READ, UPDATE, DELETE
CS	Common Service
ECDSA	Elliptic Curve Digital Signature Algorithm
EIR	Equipment Identity Register
GRR	Geo Replication Recovery
HTTPS	Hypertext Transfer Protocol Secure
IAM	Identity Access Management
Instance	NF or CNE common service managed by either M-CNCC Core or A-CNCC Core.
KPI	Key Performance Indicator
LCM	Lifecycle Management
LDAP	Lightweight Directory Access Protocol
LDAPS	Lightweight Directory Access Protocol (Over SSL)
MC	Multi Cluster. In multi cluster, a single CNCC can manage NF instances that accessess different Kubernetes clusters.
MCMI	Multiple Clusters Multiple Instances
M-CNCC	Manager CNC Console or mCncc is a CNC Console instance which manages multiple A-CNCC and local instances. **Non OCI**: M-CNCC has two components M-CNCC IAM and M-CNCC Core **OCI**: M-CNCC has only M-CNCC Core component. M-CNCC IAM is substituted with OCI IAM.

Table (Cont.) Acronyms

Acronym	Description
M-CNCC Core	Manager CNC Console Core or M-CNCC Core (also known as mCncc Core) is a core component of M-CNCC that provides GUI and API access portal for accessing NF and OCCNE common services. M-CNCC Core contains M-CNCC Core Ingress Gateway and M-CNCC Core back-end microservices.
M-CNCC IAM	Manager CNC Console IAM or M-CNCC IAM (also known as mCncc Iam) is an IAM component of M-CNCC. M-CNCC IAM contains M-CNCC IAM Ingress Gateway and M-CNCC IAM back-end microservices.
M-CNCC Kubernetes cluster	Kubernetes cluster hosting M-CNCC
MO	Mananged Objects
MOS	My Oracle Support
mTLS	Mutual Transport Layer Security
NRF	Oracle Communications Cloud Native Core, Network Repository Function
OCI	Oracle Cloud Infrastructure
OCNADD	Oracle Communications Network Analytics Data Director
OCNF	Oracle Communications Network Function
OSDC	Oracle Software Delivery Cloud
OSO	Oracle Communications Operations Services Overlay
PROVGW	Provisioning Gateway
RBAC	Role Based Access Control
Release Stream	A release stream is a sequence of releases for a product available to the customer. Example of release streams are 24.1.x, 24.2.x, 24.3.x, 25.1.1xx, 25.1.2xx, 25.2.1xx, and so on.
REST API	Representational State Transfer Application Programming Interface
SAML	Security Assertion Markup Language
SBA	Service Based Architecture
SBI	Service Based Interface
SCP	Oracle Communications Cloud Native Core, Service Communication Proxy
SCSI	Single Cluster Single Instance
SEPP	Oracle Communications Cloud Native Core, Security Edge Protection Proxy
Site	Kubernetes Cluster
SSO	Single Sign On
TLS	Transport Layer Security
UDR	Oracle Communications Cloud Native Core, Unified Data Repository
UE	User Equipment
URI	Subscriber Location Function

What's New in This Guide

This section lists the documentation updates for Release 25.1.2xx.

Release 25.2.100 - G35639-02, November 2025

No updates have been made in this release.

Release 25.2.100 - G35639-01, November 2025

- The following troubleshooting scenarios have been added in this release:
  - IAM Hook Pod Logs Reports the "Table Count Mismatch for dbName='${dbName}'" Error
  - IAM Hook Pod Logs Report "Unexpected error during SQL script execution: Communications link failure." Error
  - IAM Hook Pod Logs Report "SQLException for query='${SQL_QUERY}'. Error Message: ${SQL_ERROR}. " Error
- Updated the Log Formats section with the following:
  - Updated the Log4j JSON Format.
  - Added the `endOfBatch`, `loggerFqcn`, and `xRequestId` attributes to the Log Event Attributes table.
  - Added the `cnccId` and `instanceId`attributes to the CNC Console message format table.
- Updated the exaples of audit and security logs in the Examples of Logs section.

1 Introduction

This document provides information about troubleshooting Oracle Communications Cloud Native Configuration Console (CNC Console).

1.1 Overview

CNC Console is a single screen solution to configure and manage Network Functions (NFs). The CNC Console has the following two modules:

- CNC Console Core: CNC Console Core acts as GUI or API portal for NFs and CNE common services. CNC Console Core module includes CNC Console and its integration with other Cloud Native Core network functions. The CNC Console provides user interface that can be used to configure parameters for the following CNC network functions:
  - Oracle Communications Cloud Native Core, Binding Support Function (BSF)
  - Oracle Communications Cloud Native Core, Service Communication Proxy (SCP)
  - Oracle Communications Cloud Native Core, Network Repository Function (NRF)
  - Oracle Communications Cloud Native Core, Converged Policy (Policy)
  - Oracle Communications Cloud Native Core, Security Edge Protection Proxy (SEPP)
  - Oracle Communications Cloud Native Core, Unified Data Repository (UDR)
  - Oracle Communications Cloud Native Core Network Slice Selection Function (NSSF)
  - CNE Common Services
  - Data Director (DD)
  - Oracle Communications Network Data Analytics Function (NWDAF)
  - Provisioning Gateway (PROVGW)
  - Oracle Communications Cloud Native Core, Certificate Management (OCCM)
- CNC Console Identity Access Management (CNCC IAM): CNCC IAM acts as local identity provider and broker for external identity provider. CNCC IAM module includes the required authentication and authorization procedures such as creating and assigning roles to users.

1.2 Reference

Following are the reference documents:

- *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core DBTier Installation ans Upgrade Guide*
- *Oracle Communications Cloud Native Configuration Console User Guide*
- *Oracle Communication Cloud Native Core Data Collector Guide*

2
Logs

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

2.1 Collecting Logs

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

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

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

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

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

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

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

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

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

For more information on kubectl commands, see Kubernetes website.

2.2 Log Formats

This section provides information about the log formats.

Log4j JSON Format

CNCC Message Format

Log4j JSON Format

Following is the log format in JSON:

```
{
     "thread": <threadName>,
     "level": <log_level>,
     "loggerName": <name_of_the_logging_class>,
     "message": <message>,
     "instant": <timestamp_in_miliseconds>,
     "messageTimestamp": <timestamp_in_readable_format>,
     "endOfBatch": <default_value_from_log4j>,
```

```
    "loggerFqcn": <qualified_class_name_of_logger module>,
    "threadId": <threadId>,
    "threadPriority": <threadPriority>,
    "pod": <name_of_the_pod>,
    "processId": <processId>,
    "contextMap": <context_map>,
    "ocLogId": <unique_trace_id_for_every_request>,
    "instanceType": <instanceType>,
    "ingressTxId": <IngressTransactionId>,
    "xRequestId": <xRequestId>
}
```

Table 2-1 Log Details

Name	Description	Example
thread	Name of the thread.	"thread": "reactor-http-epoll-1"
level	Level of the log. It can be: Log level (INFO, WARN, DEBUG, TRACE)	"level": "INFO"
loggerName	Name of the class that generated the log.	"loggerName": "ocpm.cne.gateway.cncc.GatewayApplication"
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": 2020-07-04'T'12:00:40.702Z
message	Information about the event.	"message": "Started Application....." By default, all messages are in simple string except ***Audit Log***, ***Security Log*** which are represented in CNC Console Message Format..
instant	The Date and Time the event occurred in epoch second and nano seconds.	"instant": { "epochSecond": 1590045388, "nanoOfSecond": 339789000}
endOfBatch	Default value from log4j	"endOfBatch":false
loggerFqcn	Fully qualified class name of logger module.	"loggerFqcn":"org.apache.logging.log4j.internal.DefaultLogBuilder"
processId	Linux process Identifier (for a multi-process host.	"processId":"1"
threadId	Id of the thread.	"threadId":"43"
threadPriority	Priority assigned to the thread	"threadPriority": 5
pod	Name of the pods where the log is generated	"cncc-mcore-ingress-gateway-77df795fb5-wv2sb"
contextMap	It holds information added to threadContext.	"contextMap": { "hostname": "cncc-mcore-ingress-gateway-77df795fb5-wv2sb", "ingressTxId": "ingress-tx-1460885598"}
ocLogId	It contains the trace id that is uniquely generated for every request of the format "<timestamp(in milliseconds)>_<thread Id>_<POD name>"	It contains the trace id that is uniquely generated for every request of the format "<timestamp (in milliseconds)>_<thread Id>_<POD name>"

Table 2-1 (Cont.) Log Details

Name	Description	Example
instanceType	Static tag which implies that instance type is production	"instanceType": "prod"
ingressTxId	It contains id of the format "ingress-tx-<Random no>" to track every transaction	"ingressTxId": ingress-tx-1904660570
xRequestId	xRequestId is a placeholder for which value will be populated if x-request-id is present in headers	"xRequestId":"e3e59aff-85e6-4b48-92e7-6bf8e45ef06e"

CNC Console Message Format

Table 2-2 CNC Console Message Format

Name	Description	Example	Possible Values
logType	Indicates whether it is ***Security Log*** or ***Audit Log***.	logType=AUDIT	AUDIT SECURITY
type	Indicates nature or action of the log.	type=REQUEST	For Security Log: REQUEST, RESPONSE For Audit Log: LOGIN, ACCESS_RESOURCE, ACCESS_RESOURCE_ERROR, LOGOUT
resourceType	Indicates what is the resource being requested for.	resourceType=SCP	CM_SERVICE (For default route) CNC Console (For User Login Activity) SCP UDR NRF PCF (all CNC Console supported NFs)
cnccId	ID uniquely identifies the deployment and CNE Common Service Instance.	cnccId=Cluster1	NA
instanceId	ID uniquely identified the NF instance or OC-CNE Common Service Instance.	instanceId=Cluster1-scp-instance1	NA
userId	Id of the user who triggered request or action.	userId=3314f54f-08bf-489d-b395-27bf56da1262	NA

Table 2-2 (Cont.) CNC Console Message Format

Name	Description	Example	Possible Values
username	Name of the user	username= "user1"	NA
status	HTTP status of the response.	status=200 OK	NA
operationType	HTTP method of the request.	operationType=GET	NA
scheme	Indicates the scheme of the request.	scheme=http	NA
remoteAddress	The remote address that is associated with the request. It also means the remote address to where this request is connected when available.	remoteAddress=/192.168.219.64:53587	NA
localAddress	The local address that is associated with the request. It also means the local address to where this request is connected when available.	localAddress=cncc-mcore-ingress-gateway.cncc.svc.cluster.local/<unresolved>:30075	NA
resourcePath	Request URI	resourcePath=/soothsayer/v1/canaryrelease/	NA
queryParams	Query parameters associated with request.	queryParams={form_id=9, page=1, view_id=78}	NA
headers	Headers associated with request or response.	headers={Accept=*/*, X-Requested-With=XMLHttpRequest, User-Agent=Mozilla/5.0 (Windows NT 10.0; WOW64; rv:68.0) Gecko/20100101 Firefox/68.0, Connection=keep-alive, Host=cncc-core-ingress-gateway.cncc.svc.cluster.local:30075, Accept-Language=en-US,en;q=0.5, Accept-Encoding=gzip, deflate, DNT=1, Content-Type=application/json; charset=utf-8}	NA
payload	Payload or Data associated with request or response.	payload=[{"serviceName":"n5g-eir-eic","canaryReleaseFlag":true,"apiFullVersion":"2.0.0","canaryTraffic":5}	NA

Table 2-2 (Cont.) CNC Console Message Format

Name	Description	Example	Possible Values
authenticationType	This indicates whether user is requesting resource logged in using CNC Console or directly accessing through postman or curl.	authenticationType=OAUTH	OAUTH -> User is logged in through CNC Console application and accessing resource. JWT -> User is accessing resource directly through postman or curl. UNKNOWN -> Indicates that authenticationType is not applicable for the specific call flow, such as M-CNCC IAM ingress-gateway requests.

2.3 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 CNC Console, the log level of a microservice can be set to any one of the following valid values:

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

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

2.4 Types of Logs

The CNC Console logs can be categorized into following types:

- Audit logs
- Security logs
- Regular logs

Audit Logs

> **Note**
>
> CNC Console IAM Audit logs are not applicable for OCI Deployment.

These logs contain user related information and the activity within the system. Audit logs are supported in M-CNCC Core Ingress Gateway only.

The following events are logged in CNC Console Core:

- Log in: A user has logged in.
- Access Resource: A user is accessing a particular NF resource.
- Access Resource Error: A user is denied from accessing a particular NF resource.
- Logout: A user has logged out.

> **Note**
>
> The user can find the CNC Console Core User Activity logs as part of *cncc-core-ingress-gateway* and are represented in CNC Console message format.

The following events are logged in CNC Console IAM:

- Login events
- Log in: An admin user has logged in.
- Register: An admin user has registered.
- Logout: An admin user has logged out.
- Code to Token: An application or a client has exchanged a code for a token.
- Refresh Token: An application or a client has refreshed a token.

  Account events
- Update Email: The email address for an account has changed.
- Update Profile: The profile for an account has changed.
- Send Password Reset: A password reset email has been sent.
- Update Password: The password for an account has changed.

> **Note**
>
> The user can find the CNC Console IAM User Activity logs as part of *cncc-iam-0*, represented in JSON format. These events are provided by keycloak and documented under Keycloak Auditing End Events.
> Logging Error Logs are recorded by keycloak container as :

```
{"timestamp":"2025-06-05T08:38:57.050760755Z","sequence":9548,"logge
rClassName":"org.jboss.logging.Logger","loggerName":"org.keycloak.ev
ents","level":"WARN","message":"type=\"LOGIN_ERROR\",
realmId=\"master\", realmName=\"master\",
      clientId=\"security-admin-console\", userId=\"8a279153-
ce82-490f-b5d8-884c536f1f9e\",
      ipAddress=\"<IP>\", error=\"invalid_user_credentials\",
      auth_method=\"openid-connect\", redirect_uri=\"https://
<IP>:<Port>/cncc/auth/admin/master/console/\",
      code_id=\"3aaa2451-6e2f-4b6e-8ecc-c527541fb78b\",
username=\"admin\"","threadName":"executor-
thread-2","threadId":44,"mdc":{},"ndc":"","hostName":"cncc-
iam","processName":"/opt/java/jre/bin/java","processId":1}
```

Security Logs

> **Note**
>
> M-CNCC IAM Security logs are not applicable for OCI Deployment.

The security logs contain the header, payload, method, scheme, URI information for all the requests and corresponding responses.

Disabling Security Logs

By default Security Log will be enabled for *M-CCNC IAM*, *M-CNCC Core*, and *A-CNCC Core*. You can disable this by setting *securityLogEnabled* flag to false in *custom values.yaml* file.

```
# CNC Console configuration
cncc:
  # Enable security logs for CNC Console
  securityLogEnabled: false
```

Header Information

At all the log levels, sensitive information like Cookies are masked.

> **Note**
>
> The user can find the Security logs:
>
> - For M-CNCC Core and A-CNCC Core, these are logged as part of *cncc-mcore-ingress-gateway or cncc-acore-ingress-gateway* and are represented in CNC Console message format.
> - For M-CNCC IAM, these are logged as part of *cncc-iam-ingress-gateway* and are represented in CNCC message format.

Regular logs

These logs contain error messages, warnings, or other events written within the application that provide logical, high level information about the application and ongoing events.

Example:

```
M-CNCC IAM Pre Hook logs:
{"instant":
{"epochSecond":1761807160,"nanoOfSecond":450000000},"thread":"main","level":"I
NFO","loggerName":"com.oracle.cgbu.cne.cncc.iam.Application","message":"[APP-
MAIN] Hook Application started at
2025-10-30T06:52:40.449948335Z","contextMap":
{},"endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.AbstractLogge
r","threadId":1,"threadPriority":5,"messageTimestamp":"2025-10-30T06:52:40.450
+0000","application":"cncc","vendor":"oracle","engineering_version":"25.2.100-
rc.50","marketing_version":"25.2.100.0.0","microservice":"cncc-
iampreinstall","cluster":"cncc","namespace":"cncc","node":"master","pod":"cncc
-iam-pre-install-bzpzv"}

M-CNCC Core CMservice logs:
{"instant":
{"epochSecond":1761807196,"nanoOfSecond":73372171},"thread":"main","level":"IN
FO","loggerName":"ocpm.pcf.CmServiceApplication","messa"Starting
CmServiceApplication using Java 17.0.16 with PID 7 (/opt/oracle/app/app.jar
started by cnccuser
in /)","endOfBatch":false,"loggerFqcn"g.apache.commons.logging.LogAdapter$Log4
jLog","threadId":1,"threadPriority":5,"messageTimestamp":"2025-10-30T06:53:16.
073+0000"}

M-CNCC Core Ingress Gateway logs:
{"instant":
{"epochSecond":1761807399,"nanoOfSecond":247660867},"thread":"main","level":"I
NFO","loggerName":"ocpm.cne.gateway.GatewayApplication","message":"Started
GatewayApplication in 56.483 seconds (process running for
62.694)","endOfBatch":false,"loggerFqcn":"org.apache.commons.logging.LogAdapte
r$Log4jLog","threadId":1,"threadPriority":5,"messageTimestamp":"2025-10-30T06:
56:39.247+0000","processId":"1","ingressTxId":"","pod":"","ocLogId":"","instan
ceType":"prod","xRequestId":""}
{"instant":
{"epochSecond":1761807399,"nanoOfSecond":331262912},"thread":"pool-17-
thread-2","level":"INFO","loggerName":"com.oracle.common.metrics.ConfigClientM
etrics","message":"Pegged ConfigClient Request metric with releaseVersion
25.2.104 and configVersion
0","endOfBatch":false,"loggerFqcn":"org.apache.logging.log4j.spi.AbstractLogge
```

r","threadId":75,"threadPriority":5,"messageTimestamp":"2025-10-30T06:56:39.331+0000","processId":"1","ingressTxId":"","pod":"","ocLogId":"","instanceType":"prod","xRequestId":""}

Log Levels

> **Note**
>
> M-CNCC IAM ingress-gateway default log levels are not applicable for OCI deployment.

Default log levels set for M-CNCC Core and A-CNCC Core:

```
ingress-gateway:
  log:
    level:
      cncc:
        root: WARN
        audit: INFO
        security: INFO
```

Default log levels set for M-CNCC IAM:

```
ingress-gateway:
  log:
    level:
      cncc:
        root: WARN
        security: INFO
```

Updating M-CNCC IAM Kubernetes Cluster (KC) log level

> **Note**
>
> M-CNCC IAM KC logs levels are not applicable for OCI Deployment.

- By default the log level of M-CNCC IAM KC is set to **WARN,org.keycloak.events:DEBUG** This means the *root log-level* is set to **WARN** and the *org.kecyalok.events* package is set to **DEBUG**
- In the **level** label, set the log level. Following are the different options available to set the log level:
  - **Log level**
    - TRACE
    - DEBUG
    - INFO
    - WARN

    - ERROR
    - FATAL
  - **level: OFF**
    No logs will appear
- Sample M-CNCC IAM KC log configuration:

```
kc:
 log:
   level: WARN,org.keycloak.events:DEBUG
```

Table 2-3 Supported Headers for Logging

Header	Header values (regex)
Content-Type	^application/x-www-form-urlencoded.*
	^application/json.*
	^application/problem+json.*
Accept	^application/json.*
	^application/ld+json.*
	^application/xml.*
	^multipart/form-data.*

Role of supporting headers in CNC Console Audit and Security logs

- At INFO level, only those request and response that match the supporting headers and values are logged.
- At DEBUG level, no supporting headers used and all request and response are logged.
- At ERROR / WARN, no supporting headers used and only error or warnings are logged.

> **Note**
>
> Any failure in authorizing a request will always be logged irrespective of the supported header configuration.

2.5 Examples of Logs

This section lists the examples of audit and security logs.

Examples of Audit Logs

CNC Console Core

Only message part of the JSON log is shown in the example.

- User successfully logging into CNC Console Core

```
logType=AUDIT, type=LOGIN, resourceType=CNCC,
userId=186f6f2a-ba6a-4812-8a18-b906a5f9e3f6, username=user,
operationType=GET, remoteAddress=/<IP>:<Port>,
localAddress=cncc-core-ingress-gateway.cncc.svc.cluster.local/
```

```
<unresolved>:30075,
resourcePath=/login/oauth2/code/cncc-iam,
authenticationType=OAUTH
```

- User accessing SCP resource having SCP_READ role

```
logType=AUDIT, type=ACCESS_RESOURCE, resourceType=SCP, cnccId=Cluster1,
instanceId=Cluster1-scp-instance1, userId=8238c2be-14bb-420b-
b646-86081a9fd7bc, username=user, operationType=GET, remoteAddress=/
192.168.219.64:53587, localAddress=cncc-mcore-ingress-
gateway.cncc.svc.cluster.local/<unresolved>:30075, resourcePath=/ocscp/
scpc-configuration/v1/cncc/datamodel/canaryRelease,
authenticationType=OAUTH
```

- User updating(PUT) SCP resource having SCP_WRITE role

```
logType=AUDIT, type=ACCESS_RESOURCE, resourceType=SCP, cnccId=Cluster1,
instanceId=Cluster1-SCP-Instance1, userId=8238c2be-14bb-420b-
b646-86081a9fd7bc, username=user, operationType=PUT, remoteAddress=/
192.168.219.64:53587, localAddress=cncc-mcore-ingress-
gateway.cncc.svc.cluster.local/<unresolved>:30075, resourcePath=/ocscp/
scpc-configuration/v1/canary-release/n5g-eir-eic, authenticationType=OAUTH
```

- User accessing NRF resource without having NRF_READ role

```
logType=AUDIT, type=ACCESS_RESOURCE_ERROR, status=403 FORBIDDEN,
resourceType=NRF, cnccId=Cluster1, instanceId=Cluster1-nrf-instance1,
userId=81fd0f84-5632-4a63-b066-f8a7e87d88c2, username=user,
operationType=PUT, remoteAddress=/192.168.219.64:53587, localAddress=cncc-
mcore-ingress-gateway.cncc.svc.cluster.local/<unresolved>:30075,
resourcePath=/nrf-configuration/v1/system-options, message=User do not
have required NF Level permission!!, authenticationType=OAUTH
```

- User successful logout

```
logType=AUDIT, type=LOGOUT, resourceType=CNCC, userId=81fd0f84-5632-4a63-
b066-f8a7e87d88c2, username=user, operationType=POST, remoteAddress=/
192.168.219.64:53587, localAddress=cncc-mcore-ingress-
gateway.cncc.svc.cluster.local/<unresolved>:30075, resourcePath=/logout,
authenticationType=OAUTH
```

CNC Console IAM:

> **Note**
>
> - The CNC Console IAM Audit log examples defined below are not applicable for OCI Deployment.
> - M CNCC-IAM KC Audit event logs are visible only in DEBUG level.
> - To see M CNCC-IAM KC Audit event logs, set the *org.keycloak.events* package log to DEBUG in the custom_values.yaml. Refer Updating M-CNCC IAM KC loglevel for log level configuration.

- Login Error when password entered was wrong


```
{"timestamp":"2025-06-05T08:38:57.050760755Z","sequence":9548,"loggerClassN
ame":"org.jboss.logging.Logger","loggerName":"org.keycloak.events","level":
"WARN","message":"type=\"LOGIN_ERROR\",
      realmId=\"master\", realmName=\"master\", clientId=\"security-admin-
console\",
      userId=\"8a279153-ce82-490f-b5d8-884c536f1f9e\",
ipAddress=\"10.75.212.182\",
      error=\"invalid_user_credentials\", auth_method=\"openid-connect\",
      redirect_uri=\"https://<IP>:<Port>/cncc/auth/admin/master/
console/\", code_id=\"3aaa2451-6e2f-4b6e-8ecc-c527541fb78b\",
      username=\"admin\"","threadName":"executor-
thread-2","threadId":44,"mdc":{},"ndc":"","hostName":"cncc-
iam","processName":"/opt/java/jre/bin/java","processId":1}
```


- Login with correct credential


```
{"timestamp":"2025-06-05T08:41:29.529227833Z","sequence":9550,"loggerClassN
ame":"org.jboss.logging.Logger","loggerName":"org.keycloak.events","level":
"DEBUG","message":"type=\"LOGIN\", realmId=\"master\",
realmName=\"master\", clientId=\"security-admin-console\",
userId=\"8a279153-ce82-490f-b5d8-884c536f1f9e\",
sessionId=\"3aaa2451-6e2f-4b6e-8ecc-c527541fb78b\",
ipAddress=\"10.75.212.182\", auth_method=\"openid-connect\",
response_type=\"code\", redirect_uri=\"https://10.75.212.182:30085/cncc/
auth/admin/master/console/\", consent=\"no_consent_required\",
code_id=\"3aaa2451-6e2f-4b6e-8ecc-c527541fb78b\", username=\"admin\",
response_mode=\"query\", authSessionParentId=\"3aaa2451-6e2f-4b6e-8ecc-
c527541fb78b\", authSessionTabId=\"4R9Z3s2rcLw\"","threadName":"executor-
thread-7","threadId":51,"mdc":{},"ndc":"","hostName":"cncc-
iam","processName":"/opt/java/jre/bin/java","processId":1}
{"timestamp":"2025-06-05T08:41:38.918832364Z","sequence":9551,"loggerClassN
ame":"org.jboss.logging.Logger","loggerName":"org.keycloak.events","level":
"DEBUG","message":"type=\"CODE_TO_TOKEN\", realmId=\"master\",
realmName=\"master\", clientId=\"security-admin-console\",
userId=\"8a279153-ce82-490f-b5d8-884c536f1f9e\",
sessionId=\"3aaa2451-6e2f-4b6e-8ecc-c527541fb78b\",
ipAddress=\"10.75.212.182\", token_id=\"24ed07bc-e362-4f32-
b3dc-7815d1e0aa72\", grant_type=\"authorization_code\",
refresh_token_type=\"Refresh\", scope=\"openid email profile\",
refresh_token_id=\"f6739f5b-5505-4a8b-8e1b-f64f92e139d3\",
code_id=\"3aaa2451-6e2f-4b6e-8ecc-c527541fb78b\",
client_auth_method=\"client-secret\"","threadName":"executor-
thread-6","threadId":50,"mdc":{},"ndc":"","hostName":"cncc-
iam","processName":"/opt/java/jre/bin/java","processId":1}
```


- User created


```
{"timestamp":"2025-06-05T08:44:45.741470322Z","sequence":9555,"loggerClassN
ame":"org.jboss.logging.Logger","loggerName":"org.keycloak.events","level":
"DEBUG","message":"operationType=\"CREATE\",
      realmId=\"master\", realmName=\"master\",
      clientId=\"6d475927-19a8-4082-a9f4-8d6b68afbb1a\",
      userId=\"8a279153-ce82-490f-b5d8-884c536f1f9e\",
ipAddress=\"10.75.212.182\",
```

```
      resourceType=\"USER\",
      resourcePath=\"users/
44ed4289-0b2a-4c9f-9e39-16ca70f7e80f\"","threadName":"executor-
thread-11","threadId":56,"mdc":{},"ndc":"","hostName":"cncc-
iam","processName":"/opt/java/jre/bin/java","processId":1}
```


- Deleted user


```
{"timestamp":"2025-06-05T08:46:32.099434501Z","sequence":9560,"loggerClassN
ame":"org.jboss.logging.Logger","loggerName":"org.keycloak.events","level":
"DEBUG","message":"operationType=\"DELETE\",
      realmId=\"master\", realmName=\"master\",
      clientId=\"6d475927-19a8-4082-a9f4-8d6b68afbb1a\",
      userId=\"8a279153-ce82-490f-b5d8-884c536f1f9e\",
ipAddress=\"10.75.212.182\",
      resourceType=\"USER\",
      resourcePath=\"users/
44ed4289-0b2a-4c9f-9e39-16ca70f7e80f\"","threadName":"executor-
thread-14","threadId":59,"mdc":{},"ndc":"","hostName":"cncc-
iam","processName":"/opt/java/jre/bin/java","processId":1}
```


- Admin Role removed for a user


```
{"timestamp":"2025-06-05T08:48:59.017544916Z","sequence":9562,"loggerClassN
ame":"org.jboss.logging.Logger","loggerName":"org.keycloak.events","level":
"DEBUG","message":"operationType=\"DELETE\",
      realmId=\"master\", realmName=\"master\",
      clientId=\"6d475927-19a8-4082-a9f4-8d6b68afbb1a\",
      userId=\"8a279153-ce82-490f-b5d8-884c536f1f9e\",
ipAddress=\"10.75.212.182\",
      resourceType=\"REALM_ROLE_MAPPING\",
      resourcePath=\"users/4772a1e5-9f7d-416b-9f43-8e735bba0589/role-
mappings/realm\"","threadName":"executor-thread-18","threadId":63,"mdc":
{},"ndc":"","hostName":"cncc-iam","processName":"/opt/java/jre/bin/
java","processId":1}
```


- Admin Role added for a user


```
{"timestamp":"2025-06-05T08:49:07.940955161Z","sequence":9563,"loggerClassN
ame":"org.jboss.logging.Logger","loggerName":"org.keycloak.events","level":
"DEBUG","message":"operationType=\"CREATE\",
      realmId=\"master\", realmName=\"master\",
      clientId=\"6d475927-19a8-4082-a9f4-8d6b68afbb1a\",
      userId=\"8a279153-ce82-490f-b5d8-884c536f1f9e\",
ipAddress=\"10.75.212.182\",
      resourceType=\"REALM_ROLE_MAPPING\",
      resourcePath=\"users/4772a1e5-9f7d-416b-9f43-8e735bba0589/role-
mappings/realm\"","threadName":"executor-thread-18","threadId":63,"mdc":
{},"ndc":"","hostName":"cncc-iam","processName":"/opt/java/jre/bin/
java","processId":1}
```


- Realm setting update


```
{"timestamp":"2025-06-05T08:50:03.516786633Z","sequence":9565,"loggerClassN
ame":"org.jboss.logging.Logger","loggerName":"org.keycloak.events","level":
```

```
"DEBUG","message":"operationType=\"UPDATE\",
      realmId=\"master\", realmName=\"master\",
      clientId=\"6d475927-19a8-4082-a9f4-8d6b68afbb1a\",
      userId=\"8a279153-ce82-490f-b5d8-884c536f1f9e\",
ipAddress=\"10.75.212.182\",
      resourceType=\"REALM\",
      resourcePath=\"null\"","threadName":"executor-
thread-23","threadId":68,"mdc":{},"ndc":"","hostName":"cncc-
iam","processName":"/opt/java/jre/bin/java","processId":1}
```

Examples of Security Logs

Representation for IAM and Core are same as these logs are part of ingress-gateway. Only message part of the JSON log is shown in the example.

CNC Console Core

- SCP request

  ```
  logType=SECURITY, type=REQUEST, resourceType=SCP, cnccId=Cluster1,
  instanceId=Cluster1-scp-instance1, userId=8238c2be-14bb-420b-
  b646-86081a9fd7bc, username=user, operationType=GET, scheme=http,
  remoteAddress=/192.168.219.64:53587, localAddress=cncc-mcore-ingress-
  gateway.cncc.svc.cluster.local/<unresolved>:30075, resourcePath=/ocscp/
  scpc-configuration/v1/canary-release, queryParams={},
  headers={Cookie={masked}, X-Requested-With=XMLHttpRequest, Accept=*/*, oc-
  cncc-id=Cluster1, Connection=keep-alive, User-Agent=Mozilla/5.0 (Windows
  NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/
  141.0.0.0 Safari/537.36, Host=10.75.212.255:30030, DNT=1, Accept-
  Encoding=gzip, deflate, ocLogId=1761714786603_70_cncc-mcore-ingress-
  gateway-767c7f57f5-2f5xq, NettyLatency=1761714786603, Accept-Language=en-
  GB,en;q=0.9, oc-cncc-instance-id=Cluster1-scp-instance1, Content-
  Type=application/json; charset=utf-8}, payload=Either the payload is empty
  or too large, authenticationType=OAUTH
  ```

- SCP response

  ```
  logType=SECURITY, type=RESPONSE, status=200 OK, resourceType=SCP,
  cnccId=Cluster1, instanceId=Cluster1-scp-instance1,
  userId=8238c2be-14bb-420b-b646-86081a9fd7bc, username=user,
  operationType=GET, scheme=http, resourcePath=/ocscp/scpc-configuration/v1/
  canary-release, headers={date=Wed, 29 Oct 2025 08:12:18 GMT, X-Frame-
  Options=SAMEORIGIN, Referrer-Policy=no-referrer, content-length=351, Cache-
  Control=no-cache, no-store, max-age=0, must-revalidate, X-Content-Type-
  Options=nosniff, content-type=application/json, Pragma=no-cache,
  Expires=0, X-XSS-Protection=0},
  payload=[{\"configName\":\"defaultCanaryConfigName\",\"canaryData\":
  {\"apiFullVersion\":\"2.0.0\",\"canaryTraffic\":5},\"createdTimestamp\":\"2
  025-10-27 05:50:06\",\"updatedTimestamp\":\"2025-10-27 05:50:06\"},
  {\"configName\":\"n5g-eir-eic\",\"canaryData\":
  {\"apiFullVersion\":\"2.0.0\",\"canaryTraffic\":5},\"createdTimestamp\":\"2
  025-10-29 05:12:30\",\"updatedTimestamp\":\"2025-10-29 05:12:30\"}],
  authenticationType=OAUTH
  ```

CNC Console IAM

The logs for IAM Ingress Gateway contain the username and user ID for all actions performed by CNCC IAM admin users post login, alongside other relevant information.

> **Note**
>
> - There may be a few logs with **userId=UNKNOWN, username=UNKNOWN** as user information is not available at that time. However, for request having user information, Username and UserId will be populated.
> - – The `authenticationType` header is not applicable for the M-CNCC IAM call flow, so it is set to **UNKNOWN** in the M-CNCC IAM ingress-gateway security logs.
> - Below defined CNCC IAM Security log examples are not applicable for OCI Deployment.

- Request

```
logType=SECURITY, type=REQUEST, userId=2811c05e-
bc84-4597-8014-3a88ef7535c2, username=admin, operationType=GET,
scheme=http, remoteAddress=/10.233.118.0:52597, localAddress=/
10.233.80.179:8081, resourcePath=/cncc/auth/admin/realms/master/users/
0931c7a1-34a6-45f1-9db5-5cd8a043fefb,
queryParams={userProfileMetadata=true}, headers={Connection=keep-alive,
User-Agent=Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/
537.36 (KHTML, like Gecko) Chrome/131.0.0.0 Safari/537.36,
uidToken=07338817-dOac-4b18-948d-e3d5c0c91879, Host=10.121.42.53, Accept-
Encoding=gzip, deflate, svcName=cncc-iam-kc-http.amanp-ns.svc.thrust2a,
ocLogId=1733063953903_110_cncc-iam-ingress-gateway-5f7d9b9d87-dt6vj,
accept=application/json, text/plain, */*, sbi-timer-publish-headers=false,
content-type-application/json, Accept-Language=en-GB,
en; q=0.9,
sbi-timer-feature=false}, payload={}, authenticationType=UNKNOWN",
"endOfBatch": false, "loggerFqcn": "org-apache. logging.
10g4j.internal.DefaultLogBuilder", "threadId":112, "threadPriority":5,
"messageTimestamp":"2024-12-01T14:39:13.910+0000"' ,
"processId": "1", "ingressTxId":"ingress-tx-1093411995", "pod": "cncc-iam-
ingress-gateway-5f7d9b9d87-dt6vj" ,
"ocLogId":"1733063953903 110 cncc-iam-ingress-qateway-5f7d9b9d87-dt6vi"
"instanceType":"prod" "RequestId": ''}


{"instant":
{"epochSecond":1734550917,"nanoOfSecond":36483628},"thread":"igw-app-
thread14","level":"INFO","loggerName":"ocpm.cne.gateway.cncc.filters.CnccLo
ggingFilter","message":"logType=SECURITY, type=REQUEST, userId=UNKNOWN,
username=UNKNOWN, operationType=GET, scheme=http, remoteAddress=/
10.75.225.53:35507, localAddress=/192.168.219.120:8081, resourcePath=/cncc/
auth/admin/serverinfo, queryParams={}, headers={Connection=keep-alive,
User-Agent=Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/
537.36 (KHTML, like Gecko) Chrome/131.0.0.0 Safari/537.36,
NettyLatency=1734550917034, Host=10.75.225.53:30086, content-
type=application/json, Accept-Encoding=gzip, deflate, Accept-Language=en-
IN,en-GB;q=0.9,en-US;q=0.8,en;q=0.7, ocLogId=1734550917034_64_cncc-iam-
ingress-gateway-76dcb96ffb-sf46x, accept=application/json, text/plain,
```

/}, payload={},
authenticationType=UNKNOWN","endOfBatch":false,"loggerFqcn":"org.apache.log
ging.log4j.internal.DefaultLogBuilder","threadId":116,"threadPriority":5,"m
essageTimestamp":"2024-12-18T19:41:57.036+0000","processId":"1","ingressTxI
d":"ingress-tx-1584635114","pod":"cncc-iam-ingress-gateway-76dcb96ffb-
sf46x","ocLogId":"1734550908490_97_cncc-iam-ingress-gateway-76dcb96ffb-
sf46x","instanceType":"prod","xRequestId":""}

- Response

"message":" logType=SECURITY, type=RESPONSE, status=200 0K,
userId=2811c05e-bc84-4597-8014-3a88ef7535c2, username=admin,
operationType=GET, schemeshttp, resourcePath=/cncc/auth/admin/ realms/
master/clients/a6f486d4-0d0f-468d-a609-f655e9f1c492, headers={transfer-
encoding=chunked, content-length=772, Cache-Control=no-cache, Content-
Type=application/json;charset=UTF-8, Referrer-Policy=no-referrer, Strict-
Transport-Security=max-age=31536000; includeSubDomains, X-Content-Type-
Options=nosniff, X-Frame-Options=SAMEORIGIN, X-XSS-Protection=1;
mode=block, NettyLatency=1733064487125, RequestMethod=GET}, payload=
{\"id\":\"a6f486d4-0d0f-468d-a609-f655e9f1c492\", \"clientId\" :\"cncc-
realm\", \"name\": \"cncc Realm\" ,
\"surrogateAuthRequired)": false, \"enabled)": true,
\"alwaysDisplayInConsole\": false, \"clientAuthenticatorType\" :\"client-
secret\", \"redirectUris\": [1, \"webOrigins\": [1, \"notBefore\":0,
\"beareronly)": true, \"consentRequired\": false, \"standardF
lowEnabled\": true, \"implicitFlowEnabled\": false,
\"directAccessGrantsEnabled\": false, \"serviceAccountsEnabled\": false,
\"publicClient\": false, \"frontchannelLogout)": false, \"attributes)":
{}, \"authenticationFlowBindingOverrides\": {},\"fullScopeAllowed\": true,
\"nodeReRegistrationTimeout\":0, \"defaultClientScopes\": [\"web-
origins\", \"roles\", \"profile\", \"email\"], \"optionalClientScopes\":
[\"address\", \"phone\", \"offline_access\", \"microprofile-jwt\"],
\"access\":{\"view|": true, \"configure)": true, \"manage)": true}},
authenticationType=UNKNOWN", "endofBatch": false, "loggerFqcn": "org-
apache. logging. log4j. internal.DefaultLogBuilder", "threadId":122,
"threadPriority":5, "messageTimestamp": "2024-12-01T14:48:07.152+0000",
"processId": "1" ,
"ingressTxId" :"ingress-tx-577080018", "pod":"cncc-iam-ingress-
gateway-5f7d9b9d87-dt6vj", "ocLogId" :"'","instanceType": "prod" ,
"RequestId" : ""}

{"instant":
{"epochSecond":1734550917,"nanoOfSecond":49219335},"thread":"igw-app-
thread4","level":"INFO","loggerName":"ocpm.cne.gateway.cncc.filters.CnccLog
gingFilter","message":"logType=SECURITY, type=RESPONSE, status=401
UNAUTHORIZED, userId=UNKNOWN, username=UNKNOWN, operationType=GET,
scheme=http, resourcePath=/cncc/auth/admin/serverinfo, headers={transfer-
encoding=chunked, content-length=33, Content-Type=application/json,
Referrer-Policy=no-referrer, Strict-Transport-Security=max-age=31536000;
includeSubDomains, X-Content-Type-Options=nosniff, X-Frame-
Options=SAMEORIGIN, X-XSS-Protection=1; mode=block,
NettyLatency=1734550917034, RequestMethod=GET}, payload={\"error\":\"HTTP
401 Unauthorized\"},
authenticationType=UNKNOWN","endOfBatch":false,"loggerFqcn":"org.apache.log
ging.log4j.internal.DefaultLogBuilder","threadId":105,"threadPriority":5,"m

essageTimestamp":"2024-12-18T19:41:57.049+0000","processId":"1","ingressTxId":"ingress-tx-1153883762","pod":"cncc-iam-ingress-gateway-76dcb96ffb-sf46x","ocLogId":"","instanceType":"prod","xRequestId":""}

2.6 Configuring Security Logs

This section provides the details about configuring security logs.

Setting at Log Level
By default, Security Log is set to the "INFO" level for both CNC Console Core and CNC Console IAM. You can change the log level by setting log.level.cncc.security to the required level in core and iam values.yaml file.

values.yaml

```
#Set the root log level
log:
  level:
    root: WARN
    ingress: INFO
    oauth: INFO
    cncc:
      security: INFO
```

Disabling Security Log
By default, the Security Log is enabled for both CNCC Core and CNCC IAM. You can disable this by setting `securityLogEnabled` flag to false in core and iam values.yaml file.

values.yaml

```
# CNCC configuration
cncc:
  enabled: false
  enablehttp1: false
  securityLogEnabled: false
```

2.6.1 Accessing logs

This section gives information about how to access the logs.

2.6.1.1 Accessing Logs in Non OCI Deployment

The CNC Console application logs can be accessed in following ways:

1. Run the following command to view logs of a CNC Console application pod:

   ```
   kubectl logs -f -n <cncc_namespace> <pod_name> -c <container_name>
   ```

   Example:

CNC Console Core:

```
$ kubectl logs -f -n cncc cncc-mcore-ingress-gateway-77df795fb5-wv2sb -c
mcore-ingress-gateway (Security & Audit Log)
```

CNC Console IAM:

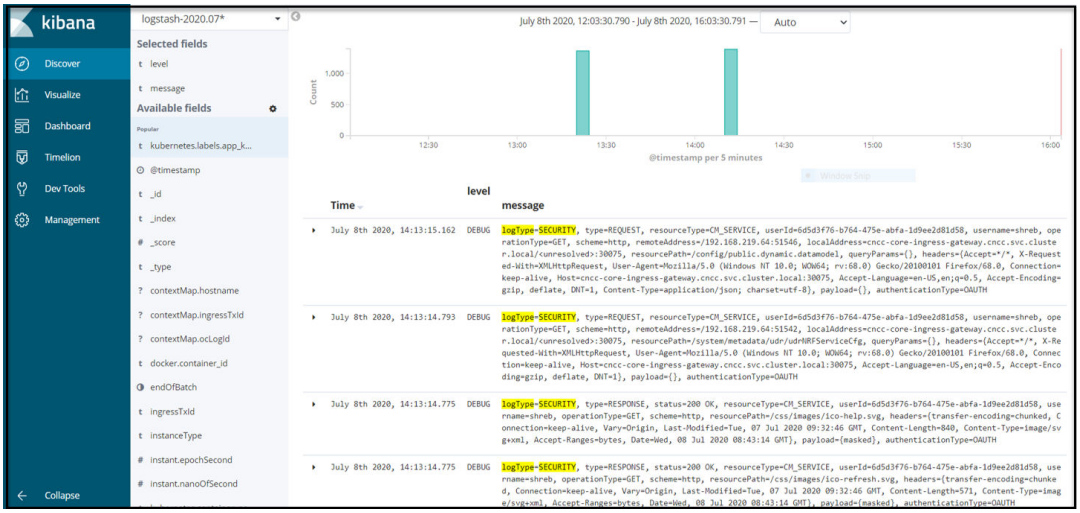
```
$ kubectl logs -f -n cncc cncc-iam-ingress-gateway-77df795fb5-wv2sb -c iam-
ingress-gateway (Security Log)
```

```
$ kubectl logs -f -n cncc cncc-iam-0 -c iam-kc(Audit Log)
```

2. CNC Console uses cloud native supported logging framework to view the logs.

   Example : Elasticsearch, Fluentd, and Kibana (EFK) can be used here with CNC Console to view the logs as follows:

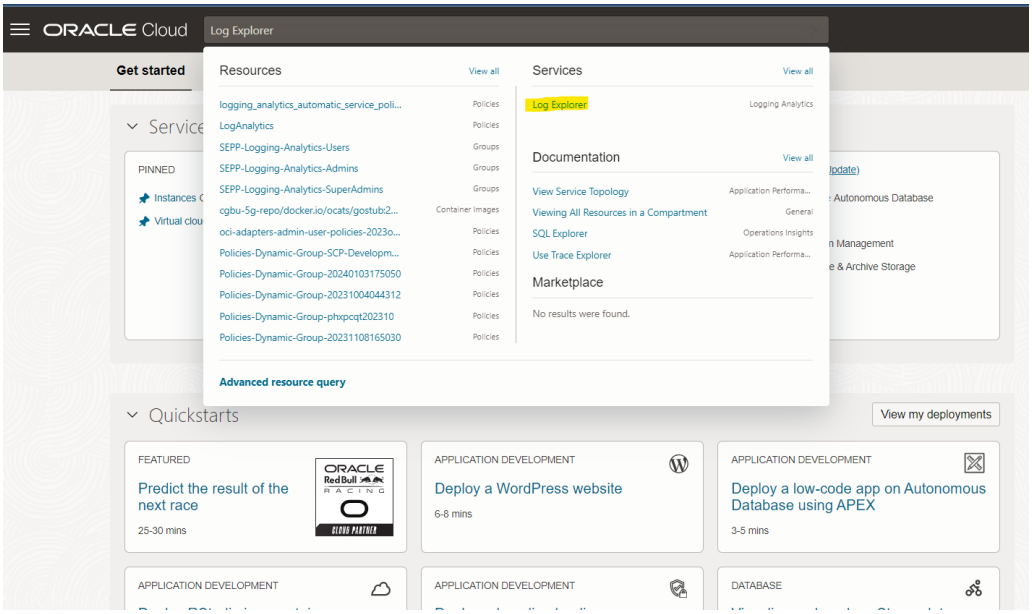
Figure 2-1 Log View



2.6.1.2 Accessing logs in OCI Deployment

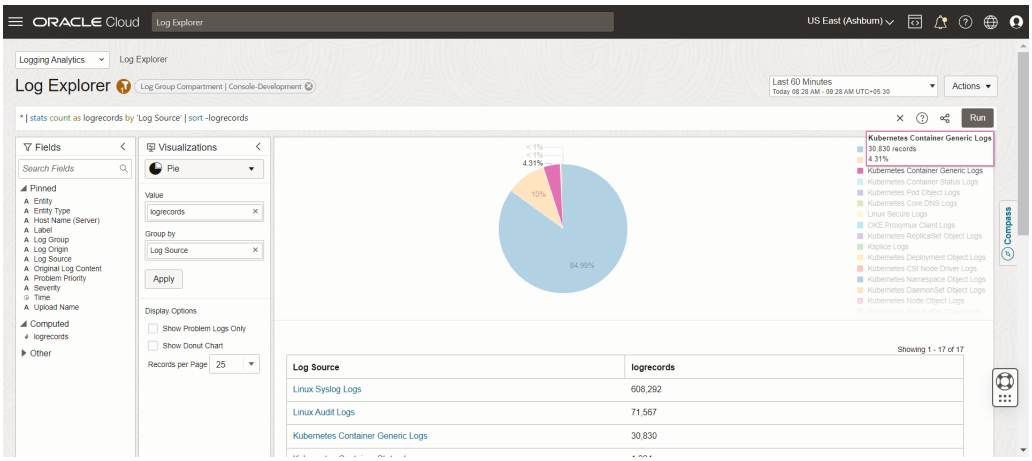
1. Viewing logs for OCI IAM in OCI Console GUI:
   See the Generate Identity and Access Management Reports from Oracle Cloud Infrastructure Audit document to view OCI IAM logs.
2. Viewing logs for M-CNCC Core and A-CNCC Core in OCI Console GUI:
   a. Log in to OCI Console GUI and search for **Log Explorer** or Click the Hamburger menu, click **Observability and Management**, then click **Log Explorer** in **Logging Analytics** section. This will open the Log Explorer page.

Figure 2-2 Log Explorer



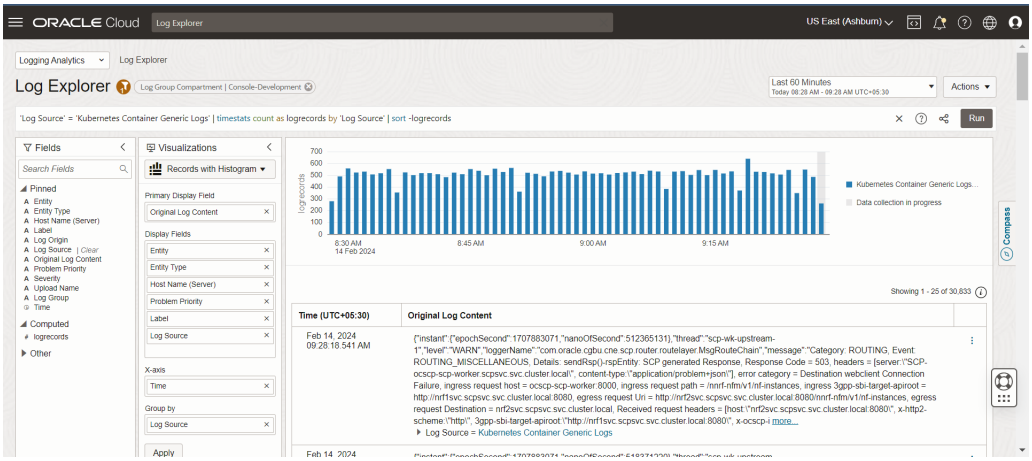
b. On the Log Explorer page, click **Kubernetes Container Generic logs.**

Figure 2-3 Kubernetes Container Generic logs



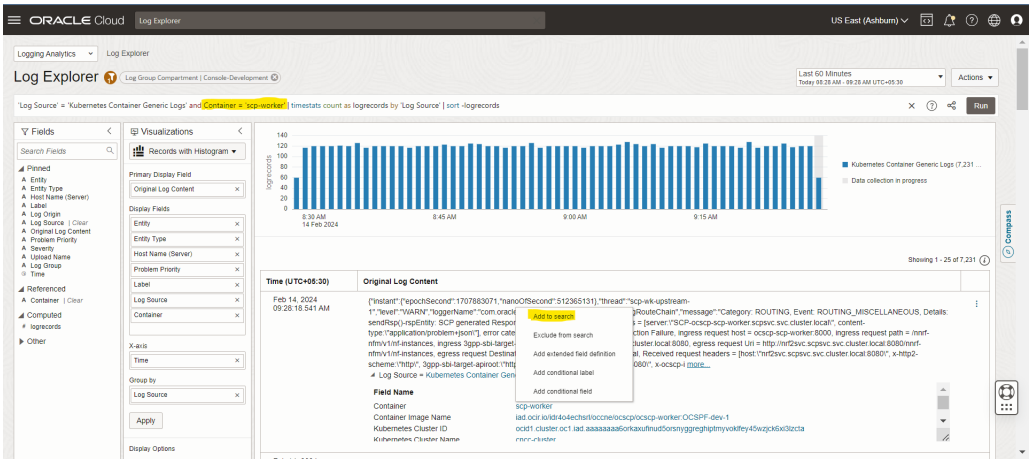
c. After clicking the **Kubernetes Container Generic Logs** link, logs will be visible.

Figure 2-4 View Logs



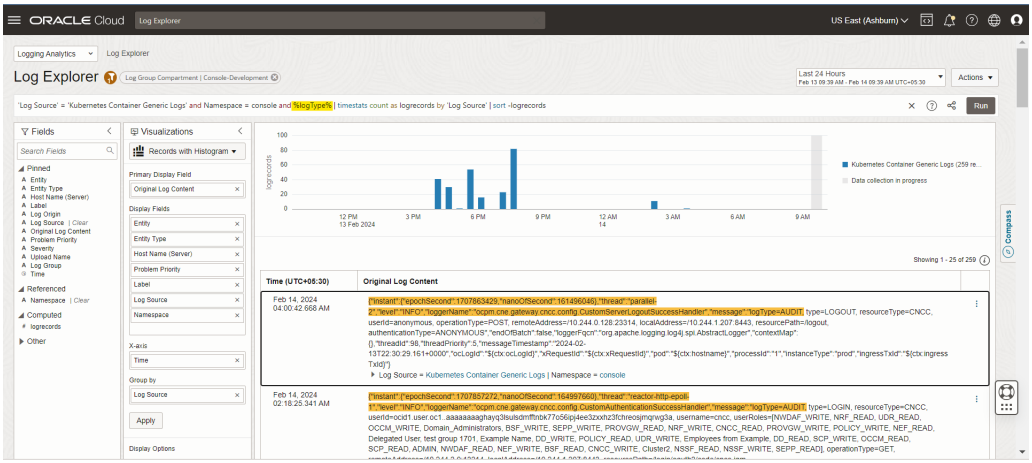
d. You can filter the logs by the container by clicking **Add to search**. Similarly other log source fields can also be added to filter logs.

Figure 2-5 Filter the logs by the container



e. Filter logs based on the logType regex pattern for security and audit logs. Similarly required regex patterns can be used to filter the logs.

Figure 2-6 Regex Patterns for Filtering Logs



> **Note**
>
> For more details, see the Explore Logs section in the Oracle Cloud Infrastructure Logging Analytics Quick Start Guide.

2.6.2 Debugging using Logs

This section provides information to debug CNC Console using Logs.

Table 2-4 CNCC Core Debugging through Logs

Scenario	Level	Logs to be searched
CNC Core Login	INFO	Login successful
Session Timeout Value	INFO	Session timeout
Validating user authorization	INFO	User Authorization Details
Accessing a resource	DEBUG	Mapping [Exchange: GET
Updating a resource	DEBUG	Mapping [Exchange: PATCH 'or' Mapping [Exchange: PUT
Creating a new resource	DEBUG	Mapping [Exchange: POST 'or' Mapping [Exchange: PUT
CNCC Core Logout	INFO	Logout successful

3 Debug Tool

Overview

> **Note**
>
> CNC Console IAM is not applicable for OCI.

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

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

> **Note**
>
> Debug Tool is only applicable for lab setup, and not recommended for production use.

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

Prerequisites

This section explains the prerequisites for using debug tool.

- For CNE 23.2.0 and later versions, follow step a of Configuration in CNE.
- For CNE versions prior to 23.2.0, follow step b of Configuration in CNE.

1. **Configuration in CNE**
   The following configurations must be performed in the Bastion Host.
   a. When CNC Console is installed on CNE version 23.2.0 or above

> **Note**
>
> - You need admin privileges to edit or patch the clusterpolicies that are mentioned in following steps.
> - In CNE v23.2.0 onwards, kyverno policy, disallow-capabilities does not allow NET_ADMIN and NET_RAW capabilities required for debug tool.

Adding Namespace to the Empty Resource

If the current disallow-capabilities cluster policy has no namespace in it, for example:

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

Sample output

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

Then, run the following command to add the namespace:

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

Example:

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

Sample output

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

To remove the namespace, run the following:

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

Sample output

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

Adding a Namespace to the Existing Namespace List

If the current disallow-capabilities cluster policy has already namespaces added in it, for example:

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

Output:

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

Then, run the following command to add your namespace:

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

Example:

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

Sample Output:

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

To remove the namespace, run the following command:

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

Example:

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

Output:

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

b. **Configurations in CNE Versions Prior to 23.2.0**
The following configurations must be performed in the Bastion Host.

> **Note**
>
> These steps are needed only when you have PSP admission controller enabled in your kubernetes environment.

PodSecurityPolicy (PSP) Creation

i. Log in to the Bastion Host.

ii. Create a new PSP by running the following command. The parameters *readOnlyRootFileSystem*, *allowPrivilegeEscalation*, *allowedCapabilities* are needed by debug container.

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

PodSecurityPolicy

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

Table 3-1 PodSecurityPolicy

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

Table 3-1 (Cont.) PodSecurityPolicy

Parameter	Description
metadata.name	Name must be unique within a namespace.
spec	spec defines the policy enforced.
spec.readOnlyRootFilesystem	Controls whether the containers run with a read-only root filesystem (that is no writable layer).
spec.allowPrivilegeEscalation	Gates whether or not a user is allowed to set the security context of a container to allowPrivilegeEscalation=true.
spec.allowedCapabilities	Provides a list of capabilities that are allowed to be added to a container.
spec.fsGroup	Controls the supplemental group applied to some volumes. RunAsAny allows any fsGroup ID to be specified.
spec.runAsUser	Controls which user ID the containers are run with. RunAsAny allows any runAsUser to be specified.
spec.seLinux	RunAsAny allows any seLinuxOptions to be specified.
spec.supplementalGroups	Controls which group IDs containers add. RunAsAny allows any supplementalGroups to be specified.
spec.volumes	Provides a list of allowed volume types. The allowable values correspond to the volume sources that are defined when creating a volume.

Role Creation

Create a role for the PSP by running the following commands:

Role

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

Table 3-2 Role

Parameter	Description
apiGroups	APIGroups is the name of the APIGroup that contains the resources.
apiVersion	APIVersion defines the versioned schema of this representation of an object.
kind	Kind is a string value representing the REST resource this object represents.

Table 3-2 (Cont.) Role

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

RoleBinding Creation

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

RoleBinding

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

Table 3-3 RoleBinding

Parameter	Description
apiVersion	APIVersion defines the versioned schema of this representation of an object.
kind	Kind is a string value representing the REST resource this object represents.
metadata	Standard object's metadata.
metadata.name	Name must be unique within a namespace.
metadata.namespace	Namespace defines the space within which each name must be unique.
roleRef	RoleRef can reference a Role in the current namespace or a ClusterRole in the global namespace.
roleRef.apiGroup	APIGroup is the group for the resource being referenced
roleRef.kind	Kind is the type of resource being referenced
roleRef.name	Name is the name of resource being referenced

Table 3-3 (Cont.) RoleBinding

Parameter	Description
subjects	Subjects holds references to the objects the role applies to.
subjects.kind	Kind of object being referenced. Values defined by this API group are "User", "Group", and "ServiceAccount".
subjects.apiGroup	APIGroup holds the API group of the referenced subject.
subjects.name	Name of the object being referenced.

Configuration Changes in CNCC Helm Charts

To enable debug tools container, make changes to *occncc_custom_values_<version>.yaml* file at Global level by setting *extraContainers: ENABLED.*

```
global:
  # Allowed Values: DISABLED, ENABLED
  # Preference is to set  "resources" request and limit to same values to
avoid HPA issues.
  extraContainers: ENABLED
  debugToolContainerMemoryLimit: 4Gi
  extraContainersVolumesTpl: |
    - name: debug-tools-dir
      emptyDir:
        medium: Memory
        sizeLimit: {{ .Values.global.debugToolContainerMemoryLimit | quote }}
  extraContainersTpl: |
    - command:
        - /bin/sleep
        - infinity
      image: {{ .Values.global.dockerRegistry }}/occncc/ocdebug-
tools:<debugtool_version
      imagePullPolicy: IfNotPresent
      name: {{ printf "%s-tools-%s" (include "getprefix" .) (include
"getsuffix" .) | trunc 63 | trimPrefix "-" | trimSuffix "-"  }}
      resources:
        limits:
          ephemeral-storage: "512Mi"
          cpu: "0.5"
          memory: {{ .Values.global.debugToolContainerMemoryLimit | quote }}
        requests:
          ephemeral-storage: "512Mi"
          cpu: "0.5"
          memory: {{ .Values.global.debugToolContainerMemoryLimit | quote }}
      securityContext:
        allowPrivilegeEscalation: true
        capabilities:
          drop:
          - ALL
          add:
          - NET_RAW
          - NET_ADMIN
        #runAsUser: <user-id>
      volumeMounts:
```

```
      - mountPath: /tmp/tools
        name: debug-tools-dir
```

To enable debug tools at service level, make changes to *occncc_custom_values_<version>.yaml* file at service level by setting *extraContainers: USE_GLOBAL_VALUE*

occncc_custom_values_<version>.yaml

```
cncc-iam:
  kc:
    # Allowed Values: DISABLED, ENABLED, USE_GLOBAL_VALUE
    extraContainers: USE_GLOBAL_VALUE

  ingress-gateway:
    # Allowed Values: DISABLED, ENABLED, USE_GLOBAL_VALUE
    extraContainers: USE_GLOBAL_VALUE




mcncc-core:
  cmservice:
    # Allowed Values: DISABLED, ENABLED, USE_GLOBAL_VALUE
    extraContainers: USE_GLOBAL_VALUE

  ingress-gateway:
    # Allowed Values: DISABLED, ENABLED, USE_GLOBAL_VALUE
    extraContainers: USE_GLOBAL_VALUE




acncc-core:
  ingress-gateway:
    # Allowed Values: DISABLED, ENABLED, USE_GLOBAL_VALUE
    extraContainers: USE_GLOBAL_VALUE
```

Note

Debug Tool Container comes up with the default user ID - 7000. If the operator wants to override this default value, it can be done using the **runAsUser** field, otherwise the field can be skipped.

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

To override runAsUser add this line under securityContext in extraContainersTpl

```
runAsUser: <user-id>
```

Configuration Options

Table 3-4 Configuration Options

Parameter	Description
global.extraContainersTpl	Describes the kubernetes container template for the debug-tool.
global.debugToolContainerMemoryLimit	This field describes the memory size (request and limit) and emptyDir volume size for the Debug-tool container.
global.extraContainersTpl.command	String array used for container command.
global.extraContainersTpl.image	Docker image name
global.extraContainersTpl.imagePullPolicy	Image Pull Policy
global.extraContainersTpl.name	Name of the container
global.extraContainersTpl.resources	Compute Resources required by this container
global.extraContainersTpl.resources.limits	Limits describes the maximum amount of compute resources allowed
global.extraContainersTpl.resources.requests	Requests describes the minimum amount of compute resources required
global.extraContainersTpl.resources.limits.cpu	CPU limits
global.extraContainersTpl.resources.limits.memory	Memory limits
global.extraContainersTpl.resources.limits.ephemeral-storage	Ephemeral Storage limits
global.extraContainersTpl.resources.requests.cpu	CPU requests
global.extraContainersTpl.resources.requests.memory	Memory requests
global.extraContainersTpl.resources.requests.ephemeral-storage	Ephemeral Storage requests
global.extraContainersTpl.securityContext	Security options the container should run with.
global.extraContainersTpl.securityContext.allowPrivilegeEscalation	AllowPrivilegeEscalation controls whether a process can gain more privileges than its parent process. This boolen directly controls if the no_new_privs flag will be set on the container process
global.extraContainersTpl.securityContext.capabilities	The capabilities to add or drop when running containers. Defaults to the default set of capabilities granted by the container runtime.
global.extraContainersTpl.securityContext.capabilities.drop	Removed capabilities
global.extraContainersTpl.secuirtyContext.capabilities.add	Added capabilities
global.extraContainersTpl.securityContext.runAsUser	The UID to run the entrypoint of the container process.
global.extraContainersTpl.volumeMounts.mountPath	The path where the emptyDir volume has to be mounted inside the container.
global.extraContainersTpl.volumeMounts.name	Name of the emptyDir volume.

Debug Tool Volume Parameters

Table 3-5 Debug Tool Volume Parameters

Parameter	Description
global.extraContainersVolumesTpl	Describes the kubernetes volume template for the debug-tool container
global.extraContainersVolumesTpl.name	Name of the emptyDir volume.
global.extraContainersVolumesTpl.emptyDir.sizeLimit	The size of the emptyDir volume.
global.extraContainersVolumesTpl.emptyDir.medium	Describes where emptyDir volumes are stored.

Debug Tool Usage

Following is the procedure to run Debug Tool:

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

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

   Example:

   ```
   # kubectl get pod -n cncc
   ```

   Sample Output:

   ```
    NAME                                          READY   STATUS    RESTARTS
   AGE
   cncc-acore-ingress-gateway-764f7f5f77-qnr5p   2/2     Running   0
   19m
   cncc-iam-ingress-gateway-55987f7dc9-x5nt2     2/2     Running   0
   147m
   cncc-iam-kc-0                                 2/2     Running   0
   147m
   cncc-mcore-cmservice-947cf4c89-76vq6          2/2     Running   0
   19m
   cncc-mcore-ingress-gateway-764f7f5f77-qnr5p   2/2     Running   0
   19m
   ```

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

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

   Example:

   ```
   kubectl exec -it cncc-mcore-ingress-gateway-599d858867-x9pvz -c tools -n
   cncc bash
   ```

3. Run the debug tools:

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

   Example:

   ```
   bash -4.2$ tcpdump
   ```

4. Run the following command to copy output files from container to host:

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

   Example:

   ```
   $ kubectl cp -c tools -n cncc cncc-mcore-ingress-gateway-764f7f5f77-
   qnr5p:/tmp/capture.pcap /tmp/
   ```

Steps to Enable Debug Tools Container

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

CNC Console

Run the following command to enable or disable CNC Console IAM after updating *occncc_custom_values_<version>.yaml* file on a installed setup:

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

Example:

```
$ helm upgrade cncc -f occncc_custom_values_25.1.200.yaml ocspf_helm-repo/
cncc --version 25.2.100
```

4

CNC Console Troubleshooting in Non OCI Deployment

This section provides information to troubleshoot the common errors which can be encountered during the installation and upgrade of CNC Console.

4.1 Unable to display the release version of the NF at CNC Console banner

Unable to display the release version of the NF at CNC Console banner

Problem: CNC Console banner displays the release version of CNC Console, but not displaying the release version of the NF.

Solution:

- The “About” section and Application name displayed next to Oracle logo use the envSystemName and envNFVersion helm fields.
- The value set of *envSystemName* and *envNFVersion* combines to display the Application name (Application name = envSystemName + envNFVersion).
- CNC Console Core Custom values have *envSystemName* and *envNFVersion* mentioned in it, but these values can be overridden.

4.2 Unable to reach CNC Console Core IP or port directly

Unable to reach CNC Console Core IP or port directly

Problem: Unable to reach CNC Console Core IP or port directly. *redirect_uri* is inserted instead of directly accessing the CNC Console Core.

Solution: As per the design, CNC Console redirects requests to CNC Console IAM for authentication. On successful authentication, CNC Console IAM redirects the user back to CNC Console GUI.

4.3 Admin user created under CNC Console realm is unable to access CNC Console IAM

Admin user created under CNC Console realm is unable to access CNC Console IAM

Problem: The user with ‘Admin’ privileges is unable to access CNC Console IAM.

Solution: Users created under the *Cncc* realm have access only to CNC Console Core and not to CNC Console IAM. To access CNC Console IAM, create the admin user under the *Master* realm.

4.4 CNC Console returns 403 error during NF Configuration

CNC Console returns 403 error during NF Configuration

Problem: CNCConsole returns a 403 Error Code and error *"Forbidden. Data could not be saved".*

Error Code/Error Message:
403/Forbidden

Solution: Log into CNC Console IAM to check the roles of the user. The user must have <NF>_READ and <NF>_WRITE roles assigned to perform the write operation on any NF through the CNC Console.

4.5 CNC Console returns 500 - Internal Server Error

CNC Console returns 500 - Internal Server Error

Problem: CNC Console returns a 500 Error Code while accessing NF Resource.

Error Code/Error Message:
500/Internal Server Error

Solution: The internal server error occurs when the NF routes are not configured correctly. To resolve this error, ensure that correct routes for each NF are configured during deployment. You can provide routes in either of the IP/FQDN in the Instances section:

```
id: <Instance ID>
type: <NF type>
owner: <ID of cluster owning the Instance>
ip: <IP of NF deployment>
port: <Port of NF deployment
```

4.6 CNC Console IAM is accessible, but CNC Console Core is not accessible

CNC Console IAM is accessible, but CNC Console Core is not accessible

Problem: CNC Console IAM is accessible, but CNC Console Core is not accessible.

Error Message:
The ID Token contains invalid claims, which is a JWT validation error, indicating that the system clock on your server is off.

Observation: This issue occurs when Ingress Gateway is behind in time and when CNC Console IAM is ahead of time. For example, If IAM (node1) is ahead of time and Ingress Gateway (node2) is 5 minutes behind, the Ingress Gateway invalidates the received token and throws *"The ID Token contains invalid claims: {iat=2020-05-26T08:32:12Z}"* error.

Solution: To resolve the error, you must ensure that the same time is maintained in CNC Console IAM and Ingress Gateway when they run in the same instance or different NTP server instances.

4.7 CNC Console IAM admin password configured through Kubectl secret is not reflected

CNC Console IAM admin password configured through Kubectl secret is not reflected

Problem:
CNC Console IAM admin password change through cncc-iam-secret is not working (Example: if configured cncc-iam-secret).

Solution: During the first installation, CNC Console IAM reads the password from the cncc-iam-secret and stores it in the database. So any further changes to the admin password must be done through the CNC Console IAM GUI.

4.8 Access Error in CNC Console Core GUI

Access Error in CNC Console Core GUI

Problem:

Unable to access CNC Console Core GUI and an “Invalid redirect URI” error occurs.

Observation:

This error occurs when there is a mismatch between the Root URL provided in CNC Console IAM Admin Console and the URI through which you access the CNC Console Core GUI.

For example, In CNC Console IAM, the Root URL is mentioned as *http://cncc-core-ingress-gateway.cncc.svc.cluster.local:30075/* and if you are accessing the CNC Console Core GUI with IP and NodePort, that is, *http://10.75.xx.xx:30075/** or vice-versa, you get “invalid redirect_uri” error on CNC Console Core GUI.

Solution: To resolve this error, ensure that the Root URL provided in CNC Console IAM and the URI through which you access the CNC Console Core GUI are the same.

4.9 Changing the CNC Console IAM admin password

Changing the CNC Console IAM admin password

Problem:
How to change the CNC Console IAM admin password using the REST API call.

Solution: Refer the following sections in CNC Console User Guide:

- Accessing NF Resources through Curl or Postman
- CNC Console IAM REST APIs

4.10 Unable to access Kibana

Unable to access Kibana

Problem:
Kibana Common Service is not accessible

Solution: To resolve this issue, ensure that you are accessing Kibana through the correct path. The default access path to Kibana is through "/kibana". You can also access Kibana through the URL *<node-ip>:<node-port>/mycne-cluster/kibana.*

4.11 CNC Console installation failure while installing using cnDBTier

CNC Console installation failure while installing using cnDBTier

Problem:
While installing CNC Console using cnDBTier, the cncc-iam-kc pod does not come up and goes into a crash state.

Solution: cnDBTier needs additional grants such as "REFERENCES, INDEX" due to the addition of db hook job.

4.12 CNC Console IAM kc pod fails while ASM is enabled

CNC Console IAM kc pod fails while ASM is enabled

Problem:
While ASM is enabled, CNC Console IAM kc pod fails due to Readiness probe failure.

Solution: Check whether annotation *"sidecar.istio.io/rewriteAppHTTPProbers"* is enabled and set to true under *'nonlbStatefulSets'* in *custom_cncc-iam_values.yaml* during CNC Console IAM deployment.

4.13 Unable to Access CNC Console GUI when ASM is Enabled

Unable to Access CNC Console GUI when ASM is Enabled

Problem:
Unable to access CNC Console GUI after installation as cncc-iam-ingress-gateway is listening on port 8080 instead of port 8081(ASM enabled).

Solution: After installing CNC Console, the cncc-iam-ingress-gateway is listening on port 8080 instead of port 8081 when ASM is enabled. To resolve this issue, configure the parameters in the *custom_cncc-iam_values.yaml* file as follows:

- Annotation: sidecar.istio.io/rewriteAppHTTPProbers: "\"true\""
- serviceMeshCheck: true
- Annotation: sidecar.istio.io/inject: "true"

4.14 CNC Console Core GUI does not get loaded after logging in

CNC Console Core GUI does not get loaded after logging in

Problem
CNC Console Core microservices are up and running but CNC Console Core GUI does not get loaded after logging in.

Solution

CNC Console supports only single pod deployment, check the following configurations (must be set to 1).

```
ingress-gateway:
  # Number of Pods must always be available, even during a disruption.
  minAvailable: 1
  # Min replicas to scale to maintain an average CPU utilization
  minReplicas: 1
  # Max replicas to scale to maintain an average CPU utilization
  maxReplicas: 1
```

> **Note**
>
> These are preset to 1 and these parameters are not exposed in custom values.

4.15 CNC Console is not supporting ASM with mTLS disabled configuration

CNC Console is not supporting ASM with mTLS disabled configuration

Problem
When service mesh is enabled and mTLS is set either to disabled or to permissive with insecure HTTP connections, CNC Console Core microservice doesn't come up or CNC Console Core GUI takes time to load because some internal CSS ir JS calls are still triggered with HTTPs.

Solution

1. Update *serviceMeshHttpsEnabled* to false in *custom-cncc-core_values.yaml* file to allow insecure HTTP connections.

   ```
   #Mandatory: This parameter must be set to "true" when CNC Console is
   deployed with the Service Mesh
   serviceMeshCheck: true
   # If Service Mesh is deployed with TLS/MTLS disabled then set this flag to
   false
   serviceMeshHttpsEnabled: false
   ```

2. Add a PeerAuthentication rule stating mTLS is disabled for CNC Console namespace. For example,

   ```
   kubectl apply -n <namespace> -f - <<EOF
   apiVersion: security.istio.io/v1beta1
   kind: PeerAuthentication
   metadata:
     name: cnccpeer
   spec:
     mtls:
       mode: DISABLE
   EOF

   ###################################
   ```

```
# Example with cncc as namespace #
###################################
kubectl apply -n cncc -f - <<EOF
apiVersion: security.istio.io/v1beta1
kind: PeerAuthentication
metadata:
  name: cnccpeer
spec:
  mtls:
    mode: DISABLE
EOF
```

4.16 Pods not coming up in an ASM enabled CNC Console deployment in istio injected namespace

Pods not coming up in an ASM enabled CNC Console deployment in istio injected namespace

Problem

CNC Console pods don't come up when sidecar injection is enabled and CNC Console is deployed in CNE, and the following error is seen:

```
job/cncc-iam-pre-install          Error creating: pods "cncc-iam-pre-
install-" is forbidden: PodSecurityPolicy: unable to admit pod:
 [spec.initContainers[0].securityContext.runAsNonRoot: Invalid value: false:
must be true spec.initContainers[0].securityContext: Invalid value:
[]int64{0}: group 0 must be in the ranges: [{1 65535}]
spec.initContainers[0].securityContext.capabilities.add: Invalid value:
"NET_ADMIN": capability may not be added
spec.initContainers[0].securityContext.capabilities.add: Invalid value:
"NET_RAW": capability may not be added
spec.initContainers[0].securityContext.runAsNonRoot: Invalid value: false:
must be true]
```

Solution

You must bind the ClusterRole "psp:privileged" to the current namespace:

```
kubectl apply -f - <<EOF
apiVersion: rbac.authorization.k8s.io/v1
kind: RoleBinding
metadata:
  name: "psp:<namespace>:cs-restricted"
  namespace: "<namespace>"
roleRef:
  kind: ClusterRole
  apiGroup: rbac.authorization.k8s.io
  name: "psp:privileged"
subjects:
- kind: Group
  apiGroup: rbac.authorization.k8s.io
```

```
  name: "system:serviceaccounts"
EOF
```

4.17 Failed to allocate IP for CNC Console IAM Ingress Gateway

Failed to allocate IP for CNC Console IAM Ingress Gateway

Problem
Installation of CNC Console IAM is successful but while checking CNC Console IAM service status, unable to assign the external IP for svc cncc-iam-ingress-gateway and received the following error: *Warning Allocation Failed 61s (x3 over 8m48s) metallb-controller Failed to allocate IP for "cncc/cncc-iam-ingress-gateway": no available IPs.*

Solution
Check if the annotations are missing from the cncc-iam-ingress-gateway service. Add the missing annotations, due to which the dynamic metalLbIpAllocation will work properly.

4.18 Unable to Create required tables in CNC Console IAM DB

Unable to Create required tables in CNC Console IAM DB

Problem
Deployment needs two instances of CNC Console where only the first instance is deployed correctly. After installing the second instance of CNC Console in a different namespace, the pod "cncc-voice-iam-kc-0" repeatedly crashes

Observation

After analyzing the logs, it was found that during the preinstall checks, the hook pods did not create all the required tables in the DB. For example, in the first instance DB, all tables created, while in the second instance DB, there are only 43 tables created. The cbDBTier has a maximum table limit of 512. So, during the deployment of the second instance of CNC Console, the maximum table limit threshold has exceeded, and hence 43 tables were created.

Deployment needs two instances of CNC Console, first instance is deployed correctly.

After the installation of the second cncc-iam in a different namespace, the *pod cncc-voice-iam-kc-0* is crashed repeatedly. By analyzing the logs it seems that not all the tables has been created by the hook pods during the pre-install checks. In the first instance db we can see that there are all tables created while in the second instance we can see only 43 tables.

Solution
To resolve this issue, you must either increase the maximum table limit or clean up unwanted databases to bring table count within the threshold limits. For more information about configuring the table limits, see *Oracle Communications Cloud Native Core DBTier Installation and Upgrade Guide*.

Default limits in ndb:

MaxNoOfOrderedIndexes: 512

MaxNoOfTables: 512

NoOfFragmentLogFiles: 256

4.19 Resolve CNC Console Validation hook error

Resolve CNC Console Validation hook error

Problem

Validation hook error occurs during CNC Console Core Deployment.

Solution

To resolve this issue, enable Helm Configuration Validation for CNC Console Deployment, applicable for M-CNCC Core and A-CNCC deployment.

Check the *cncc-acore-validation-hook* or *cncc-mcore-validation-hook* pod logs for the error codes. Make the required corrections in the *custom-cncc-core_values.yaml* file and reinstall M-CNCC Core or A-CNCC. For more information about validation hook and error details, see "CNC Console Multi Cluster Deployment Helm Configuration Validation" section in *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide.*

Table 4-1 Validation Hook Error Codes

Error Code	Error Message Format	Error Scenarios	Sample Error Messages
1001	Invalid value. Resource: <Configuration Name>, ID: <ID>, Attribute: <Attribute>. <More Info>	• Port should be Numeric • Scheme should be either HTTP/HTTPS • IDs should follow the alphanumeric pattern • Max Limit should be satisfied for M-CNCC IAM, A-CNCC and Instance • Max Length for Instance Id • Max Length for Self Cncc Id • CS instance must have one of these CS subtypes <grafana, kibana, jaeger, prometheus, alertmanager> • Both ip and fqdn cannot be provided. • Unsupported type • InvalidConfig • multicluster flag should be false in case of single-cluster deployment • multicluster flag should be true in case of multi-cluster deployment	Invalid value. Resource: mCnccIam, ID: Cluster1, Attribute: Port. It should be numeric value. Invalid value. Resource: instance, ID: Cluster3 Cluster3-instance1, Attribute: Scheme. Allowed values are: [http, https]. Invalid value. Resource: instance, ID: Cluster1-grafana##$$%, Attribute: id. Ids should be alphanumeric with hyphen allowed as special character. The count of mCnccIam exceeded max limit. Allowed Value:x. Actual Value: y Max limit exceeded. Allowed Value:x. Actual Value: y Invalid value. Resource: aCncc, ID: Cluster3, Attribute: N/A. Both ip and fqdn cannot be provided. Invalid value. Resource: isMultiClusterEnabled, ID:,Attribute: False. isMultiClusterEnabled is set as false, only single cluster configuration is allowed. Invalid value. Resource: isMultiClusterEnabled, ID:,Attribute: True. isMultiClusterEnabled is set as true, only multi cluster configuration is allowed.
1002	Duplicate value. Resource: <Configuration Name>, ID: <ID>, Attribute: <Attribute>. <More Info>	• All A-CNCC IDs must be unique • API prefix must be unique for all instances • Owner(Cluster) must have unique CS subtype	Duplicate value(s). Resource: aCncc, ID: [Cluster3], Attribute: id.
1003	Invalid Reference. Resource: <Configuration Name>, ID: <ID>, Attribute: <Attribute>. <More Info>	• All the Instance owners must be referenced in M-CNCC IAM IDs or A-CNCC IDs • M-CNCC IAM IDs and M-CNCC Core IDs must be same	Invalid Reference. Resource: instance, ID: Cluster5, Attribute: Owner. Not present in mCncc ids or aCncc ids. Invalid Reference. Resource: instance, ID: N/A, Attribute: N/A. M-Cncc Iam ids and M-Cncc Core ids do not match.

Table 4-1 (Cont.) Validation Hook Error Codes

Error Code	Error Message Format	Error Scenarios	Sample Error Messages
1004	Missing value. Resource: <Configuration Name>, ID: <ID>, Attribute: <Attribute>. <More Info>	• Missing apiPrefix parameter for type CS • Either of IP/FQDN should be present	Missing value. Resource: instance, ID: Cluster4-grafana, Attribute: apiPrefix. Missing value. Resource: instance, ID: Cluster3-PolicyInstance, Attribute: N/A. Either ip or fqdn is required.

4.20 Does CNC Console support Command Line Interface (CLI)

Does CNC Console support Command Line Interface (CLI)

Problem: Can NF APIs integrated with CNC Console be accessed through curl or postman.

Solution The NF configuration APIs can be accessed through CNC Console GUI or directly using postman or curl. CNC Console providess authentication and authorization in both ways. For more information, see "Generating Access Tokens and Accessing NF Resources" section in *Oracle Communications Cloud Native Configuration Console User Guide.*

4.21 Upgrade or Rollback Failure

Upgrade or Rollback Failure

Problem: Upgrade or Rollback Failure

Solution

When CNC Console upgrade or rollback fails, perform the following procedure:

1. Check the pre or post upgrade or rollback hook logs as applicable.
2. If the failure occurs, then check the cause of the failure from the logs by running the following command:

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

3. After detecting the cause of failure, do the following:
   - For upgrade failure:
     - If the cause of upgrade failure is database or network connectivity issue, then resolve the issue and rerun the upgrade command.
     - If the upgrade failure occurs during the postupgrade phase, for example, post upgrade hook failure due to target release pod not moving to ready state, then perform a rollback.
   - For rollback failure: If the cause of rollback failure is database or network connectivity issue, then resolve the issue and rerun the rollback command.

4. If the issue persists, contact My Oracle Support.

4.22 CNC Console Upgrade Results IP in Pending state

CNC Console Upgrade Results IP in Pending state

Problem: CNC Console deployment using static IP is not allocated to the new mcore service during upgrade.

Solution

CNC Console supports the single helm chart deployment for deploying all three components M-CNCC IAM, M-CNCC Core and A-CNCC Core.

Earlier CNC Console IAM and CNC Console Core were deployed independently, now with single helm chart all 3 components can be deployed using single helm install command.

Upgrade from two helm deployments to one helm deployment is supported but one of the helm deployment must be manually deleted.

CNC Console IAM deployment can be upgraded which upgrades M-CNCC IAM and freshly install M-CNCC Core and A-CNCC Core services. User can manuallydelete CNC Console Core deployment. For more information, see Upgrade and Rollback sections of *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide*.

In case, if static LoadBalancer IP is used in existing deployment, after the upgrade, new mcore service IP will be shown as pending. IP will be allocated once the existing M-CNCC Core service is uninstalled.

4.23 CNC Console Upgrade Displays Port Already in Use Error

CNC Console Upgrade Displays Port Already in Use Error

Problem: CNC Console deployment using static node port throws P*ort already in use error* during upgrade.

Solution

If static port is used in existing deployment, before upgrade, in custom values file port needs to be updated to use another port to avoid port conflict error.

4.24 CNC Console Helm Test Fails

CNC Console Helm Test Fails

Problem: CNC Console helm test fails when there are stale jobs or pods.

Solution

In some cases, Helm RC builds have intermittent issues which blocks auto deletion of jobs.

Ensure stable helm version is installed in your environment.

4.25 CNC Console Helm Test Fails with Service Account Error

CNC Console Helm Test Fails with Service Account Error

Problem: CNC Console helm test fails when there are stale jobs or pods.

CNCC helm test fails with error message "Unauthorized! Configured service account doesn't have access. Service account may have been revoked.".
Solution

The time sync between worker nodes is must for helm test to work. Ensure CNE worker nodes time is in sync.

4.26 Unable to integrate LDAP with Console deployed on ASM setup

Problem: While integrating LDAP/LDAPs with CNC console, **Test Connection** is successful but **Test Authentication** fails. Generic error is displayed on screen "Error when trying to connect to LDAP. See server.log for details. LDAP test error "

Solution:

Apply the following yaml. This YAML sets up Kubernetes resources and Istio configurations to facilitate communication with an external LDAP connectivity service from within the cncc namespace. It ensures that internal services can access the external LDAP service using DNS resolution and defines routing and traffic policies specific to Istio.

```
kubectl apply -n cncc -f - <<EOF
apiVersion: v1
kind: Endpoints
metadata:
  name: <Unique ServiceEntry Name for Service>
  namespace: <CNCC-NAMESPACE>
subsets:
- addresses:
  - ip: <Service-public-IP>
  ports:
  - port: <Service-public-PORT>
    protocol: <Service-PROTOCOL>
---
apiVersion: v1
kind: Service
metadata:
  name: <Unique Endpoint Name for Service>-headless
  namespace: <CNCC-NAMESPACE>
spec:
  clusterIP: None
  ports:
  - port: <Service-public-PORT>
    protocol: <Service-PROTOCOL>
    targetPort: <Service-public-PORT>
  sessionAffinity: None
  type: ClusterIP
---
```

```
apiVersion: v1
kind: Service
metadata:
  name: <Unique ServiceEntry Name for Service>
  namespace: <CNCC-NAMESPACE>
spec:
  externalName: <Unique ServiceEntry Name for Service>-headless.<CNCC-
NAMESPACE>.svc.cluster.local
  sessionAffinity: None
  type: ExternalName
---
apiVersion: networking.istio.io/v1alpha3
kind: ServiceEntry
metadata:
  name: <Unique ServiceEntry Name for Service>
  namespace: <CNCC-NAMESPACE>
spec:
  hosts:
  - <Service-public-FQDN>
  ports:
  - number: <Service-public-PORT>
    name: <Service-PORTNAME>
  location: MESH_EXTERNAL
---
apiVersion: networking.istio.io/v1alpha3
kind: DestinationRule
metadata:
  name: <Unique DestinationRule Name for Service>
  namespace: cncc
spec:
  host: <Service-public-FQDN>
  trafficPolicy:
    tls:
      mode: DISABLE
---
EOF
```

Example:

```
kubectl apply -n cncc -f - <<EOF
apiVersion: v1
kind: Endpoints
metadata:
  name: ldap-connectivity-service-headless
  namespace: cncc
subsets:
- addresses:
  - ip: 10.75.212.154
  ports:
  - port: 30763
    protocol: TCP
---
apiVersion: v1
kind: Service
metadata:
```

```
  name: ldap-connectivity-service-headless
  namespace: cncc
spec:
  clusterIP: None
  ports:
  - port: 30763
    protocol: TCP
    targetPort: 30763
  sessionAffinity: None
  type: ClusterIP
---
apiVersion: v1
kind: Service
metadata:
  name: ldap-connectivity-service
  namespace: cncc
spec:
  externalName: ldap-connectivity-service-headless.cncc.svc.cluster.local
  sessionAffinity: None
  type: ExternalName
---
apiVersion: networking.istio.io/v1alpha3
kind: ServiceEntry
metadata:
  name: ldap-external-se
  namespace: cncc
spec:
  hosts:
  - ldap-connectivity-service-headless.cncc.svc.cluster.local
  ports:
  - number: 30763
    name: ldap
  location: MESH_EXTERNAL
---
apiVersion: networking.istio.io/v1alpha3
kind: DestinationRule
metadata:
  name: ldap-external-dr
  namespace: cncc
spec:
  host: ldap-connectivity-service-headless.cncc.svc.cluster.local
  trafficPolicy:
    tls:
      mode: DISABLE
---
EOF
```

Once the yaml file is applied, LDAP Server can be integrated.

While setting up User Federation with CNCC IAM, when providing your company LDAP server details, connection URL needs to be configured using ldap-connectivity-service.

Example: ldap://ldap-connectivity-service:30362

4.27 CNC Console IAM GUI gives an error while accessing some resources under master realm

CNC Console IAM GUI gives an error while accessing some resources under master realm.

Problem:

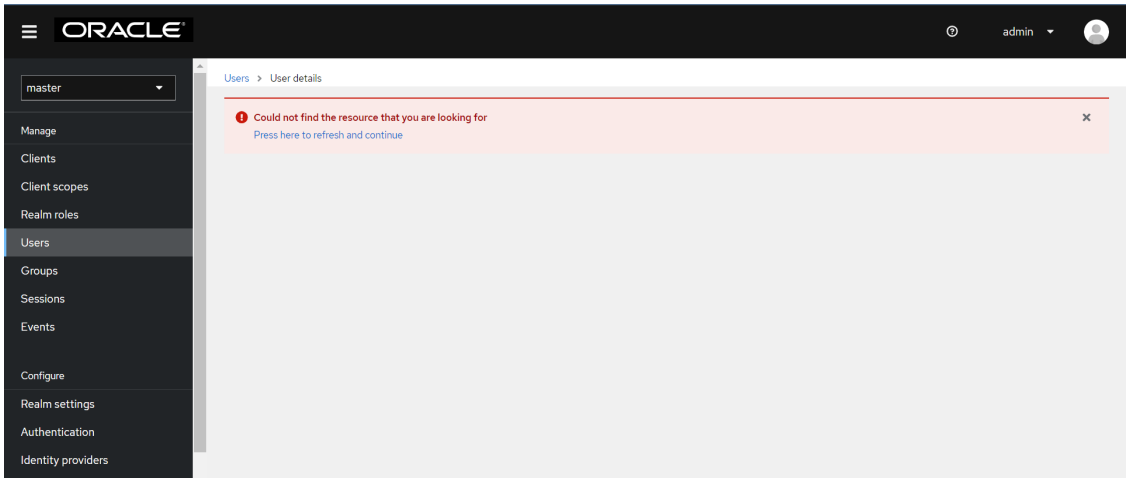
CNC Console IAM GUI throws the following error when some resources are accessed under the master realm:

"Error: Could not find the resource that you are looking for. Press here to refresh and continue."

Solution:

Some resources (APIs) are blocked under the "master realm" or "cncc realm" because of security reasons. This restricts the user from making any configuration changes. These resources are not applicable for Console. So, whenever the user tries to access blocked resources(APIs), the admin user always gets the error message asking the user to refresh and continue. It is expected behavior. Also note that CNC Console IAM is enabled with a single admin user. All admin operations must be carried out by the same admin user.

Figure 4-1 Unable to Find Resource Error



4.28 Unable to login to CNC Console Core when TLS version is v1.2 and Ciphers Other Than That of TLSv1.2 are Provided

Problem: CNC Console user is unable to log into CNC Console GUI.

After the user enters the username and password, the 503-service unavailable error is displayed on GUI. In Pod logs “The specified CipherSuites array contains invalid null or empty string elements” exception is seen.

The following conditions should occur for the issue to be observed:

1. IAM service is HTTPS.
2. M-CNCC Core or A-CNCC service has the TLS version set to TLSv1.2.
3. M-CNCC Core or A-CNCC service has ciphers other than that of TLSv1.2.

Solution: The user should specify the cipher suites applicable to the corresponding TLS version. Cipher Suites can be specified in custom values using mcncc-core.ingress-gateway.cipherSuites or acncc-core.ingress-gateway.cipherSuites.

For more information, see *Oracle communications, Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide.*

4.29 Database Connectivity Failure When Instance Level Access Control is Enabled

Problem: The operator faces database connectivity failure with the CNC Console IAM hook when the instance level access control feature is enabled.

Solution: When the connectivity is established again, operator must delete the INSTANCE_ALL role manually from the UI and run the installation for the CNC Console IAM hooks again to assign the INSTANCE_ALL role to the existing users.

4.30 403 Forbidden Error when Accessing NF or CS Resources With Instance Level Access Control Enabled

Problem: A user with some required permissions encounters the 403 forbidden error when they try to acess any NF or CS resources.

Solution: Check the acore-ingress-gateway pod logs to identify the missing permissions for the user.

4.31 Recovery of Admin Account in CNCC IAM

Problem: In CNCC IAM, if only one admin user is created and it gets locked out, and there is no alternative account available to regain access or manage the system.

Solution: There are two ways to unlock the current admin user.

Solution 1: Create a new admin user

1. In the `occncc_custom_values_<version>.yaml` file used for the current deployment, update the `global.iamUserNamefield` parameter to create a new admin user. For example, change it from `&iamUserName admin` (existing) to `&iamUserName admin2`.
   If password policies are enabled in CNCC IAM, ensure that new password adheres to the guidelines configured in CNCC IAM. For more details about the password policies, refer 'Password Policies for CNC Console Users' section in *Oracle Communications Cloud Native Configuration Console User Guide.*
   a. Run the following command to delete the existing `cncc-iam-secret`:

      ```
      $ kubectl delete secret cncc-iam-secret -n <namespace>
      ```

b. Run the following command to recreate the secret with a password that complies with the guidelines:

```
$ kubectl create secret generic <secret-name> --from-
literal=iamAdminPasswordKey='<password>' --namespace <namespace>
```

Example:

```
$ kubectl create secret generic cncc-iam-secret --from-
literal=iamAdminPasswordKey='xxxxxxxxx' --namespace cncc
```

2. Run the following Helm upgrade command to upgrade the deployment by using the updated yaml files:

```
helm upgrade <cncc_iam_release_name> <helm_chart> -f
<occncc_custom_values_<version>.yaml> -n <namespace>
```

For more details about the Helm upgrade, refer 'Upgrading CNC Console' section in the *Oracle ® Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide.*

3. After completing the upgrade, log in to CNCC IAM using the updated admin credentials. Once logged in, navigate to the **Users** section under the default realm to update the password for the old admin account if necessary.

   For more details about update the admin user password, refer 'Creating or Updating Admin Password in CNC Console IAM' section in the *Oracle ® Communications Cloud Native Configuration Console User Guide.*

   The above steps ensure you regain access to CNCC IAM by creating a new user and then resetting an existing admin user.

Solution 2: Reset the Password of the Existing Admin User

To reset the password of the existing user, do the following:

1. Run the following command to get into the mysql pod:

```
kubectl exec -it <mysql pod name> -n <namespace> bash
```

2. Run the following command to login into the mysql pod:

```
mysql -h 127.0.0.1 -u <username> -p<password>
```

3. Run the following command to use the database which you used to deploy cncc application:

```
use <database name>;
```

4. Run the following command to find the admin user id from the USER_ENTITY table:

```
select * from USER_ENTITY;
```

5. Run the following command to delete the entry from CREDENTIAL TABLE:

   ```
   delete from CREDENTIAL where user_id = '<user-id>'
   ```

6. Run the following command to delete the row from USER_ROLE_MAPPING table with user id as admin user id:

   ```
   delete from USER_ROLE_MAPPING where user_id = '<user-id>';
   ```

7. Run the following command to delete the row from USER_ENTITY table with user id as admin user id:

   ```
   delete from USER_ENTITY where ID = '<user-id>';
   ```

8. If password policies are enabled in CNCC IAM, ensure that new password adheres to the guidelines configured in CNCC IAM. For more details about the password policies, refer 'Password Policies for CNCC Users' section in the *Oracle Communications Cloud Native Configuration Console User Guide.*

   a. Run the following command to delete the existing `cncc-iam-secret`.

      ```
      $ kubectl delete secret cncc-iam-secret -n <namespace>
      ```

   b. Run the following command to recreate it with a password that complies with the guidelines:

      ```
      $ kubectl create secret generic <secret-name> --from-
      literal=iamAdminPasswordKey='<password>' --namespace <namespace>
      ```

      Example:

      ```
      $ kubectl create secret generic cncc-iam-secret --from-
      literal=iamAdminPasswordKey='xxxxxxxxx' --namespace cncc
      ```

   c. Run the following command to restart the CNCC-IAM-KC-0 pod and login to iam using the password given in cncc-iam secret:

      ```
      Kubectl delete po <IAM-KC Pod> -n <namespace>
      ```

> **Note**
>
> If the password is not compliant with configured password policies, user will get an error while login to CNCC IAM.

4.32 Pods Enter the CreateContainerConfigError State

Problem: The pods enter the `CreateContainerConfigError` state.
Solution: Do the following to debug the issue. Check the secrets (cncc-iam-secret and cncc-db-secret), and recreate them if required:

- The pods may go into `CreateContainerConfigError` state if the secrets are not configured correctly or if they are not present. The following error will be displayed:

  ```
  cncc-iam-kc-0          1/2     CreateContainerConfigError
  ```

- Run the following command to check the pod description:

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

  Sample Error Output:

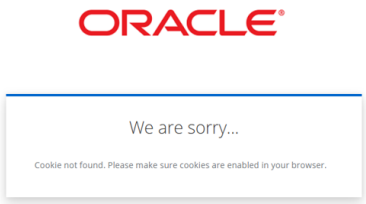
  ```
  Warning  Failed     3s (x10 over 96s)  kubelet       Error: secret "cncc-
  iam-secret" not found
  ```

- Create the secret if its missing or recreate the secret if its not configured correctly, based on the displayed error.
- Install the CNC Console again, the pods must be up and running now.

4.33 Login Error When Accessing CNC Console IAM or CNC Console Core

Problem: CNC Console IAM or CNC Console Core is inaccessible, and the browser displays the "Cookie not found. Please make sure cookies are enabled in your browser." error. This issue typically occurs when the CNC Console is rolled back to a previous version.

Figure 4-2 Login Error



Solution: Close the current browser window and open a new browser window to access CNC Console IAM or CNC Console Core GUI.

4.34 Unable to Access CNC Console Through FQDN

Problem: When setting up CNC Console with FQDN, we usually map load balancer IPs to FQDN. However, sometimes the deployment is accessible through IP but not with FQDN.

Solution: The issue may be related to how the FQDN is being resolved in the user's environment. It could eitherbe because of DNS or Network settings, or how the browser handles the URL.

Perform the following steps as a potential workaround:

1. Update the `/etc/hosts` file by adding an entry mapping the hostname K8s_SVC_NAME to the appropriate IP address instead of using the full FQDN.
2. Modify the CNC Console `custom values.yaml` file by updating the hostname reference from full FQDN to K8s_SVC_NAME.
3. Redeploy or upgrade the CNC Console deployment.

4.35 Ingress Gateway Pods go Into CrashLoopBackOff

Problem: During the CNC Console upgrade process, the upgrade completes without errors, but the pods encounter an issue and enter a CrashLoopBackOff state, preventing them from functioning properly.

Solution: Check if you are using an older Kubernetes/Go version. When deploying the CNC Console on an older Kubernetes/Go version, such as v1.23.10/go1.17.13, you must modify the CNC Console custom values file by removing `ec_point_formats` from both `clientDisabledExtensions` and `serverDisabledExtensions` to prevent deployment failures related to version compatibility.

4.36 IAM Hook Pod Logs Reports the "Table Count Mismatch for dbName='${dbName}'" Error

Problem: When you run the CNC Console installation or upgrade, you see error messages in the pod logs. IAM Hook Pod logs are displaying the following custom error messages that indicate a table count mismatch in one or more CNC Console databases:

```
[VERIFY-TABLE-COUNT] Database '${dbName}' contains '${ActualTableCount}'
tables (required='${RequiredTableCount}').
[VERIFY-TABLE-COUNT] Table count mismatch for dbName='${dbName}': present='$
{ActualTableCount}', required='${RequiredTableCount}'
```

Where,

- `${dbName}` is the name of your CNC Console Database
- `${version}` is the CNC Console version
- `${ActualTableCount}` is the number of tables currently in the database
- `${RequiredTableCount}` is the expected number of tables that should be present in the database

For example,

```
[VERIFY-TABLE-COUNT] Database 'cnccdb' contains 88 tables (required=87).
[VERIFY-TABLE-COUNT] Table count mismatch for dbName='cnccdb': present=88,
required=87
```

Solution: This error may have been caused by

- SQL script failures like driver errors, syntax errors, or timeouts.
- Insufficient user privileges to create schemas or tables.
- Interrupted installation or upgrade preventing the full schema from being applied.
- Outdated cnDBTier settings that do not match the current CNC Console MaxLimits recommendations described in the "cnDBTier Profiles" section in *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide*.

To resolve this,

- If this error is caused by outdated cnDBTier max limits, perform an in-solution upgrade of cnDBTier using the updated values.
- If this is caused by any other reason, do not restart the installation or upgrade process, or perform a rollback. Contact My Oracle Support for further analysis. When reaching out, be sure to collect and submit:
  - Complete hook job logs
  - The CNC Console Custom Values file

4.37 IAM Hook Pod Logs Report "Unexpected error during SQL script execution: Communications link failure." Error

Problem

When you run the CNC Console installation or upgrade, you see error messages in the pod logs. The messages look like this:

```
[CREATE-SCHEMAS] Schema creation failed for dbName:version ='${dbName}:$
{version}'.
Reason: [RUN-SQL-SCRIPT] Unexpected error during SQL script execution:
Communications link failure.
The driver has not received any packets from the server.
```

Where,

- `${dbName}` is the name of your CNC Console database.
- `${version}` is the CNC Console version.

For example,

```
[PRE-UPGRADE] Error during pre-upgarde process: [CREATE-SCHEMAS] Schema
creation failed for dbName:version ='cnccdb_tmp:25.1.200'. Reason: [RUN-SQL-
SCRIPT] Unexpected error during SQL script execution: Communications link
failure\n\nThe last packet sent successfully to the server was 0 milliseconds
ago. The driver has not received any packets from the server..
```

Solution

You may have encoutered this issue because

- the connection to the MySQL database is not working. This may be due to the hostname being wrong, the port being closed, the host being unreachable, firewall settings, or problems with the DNS lookup.
- there may have been a network interruption during the schema creation.

Perform the following procedure to resolve this issue:

1. Check that you can connect to the database from where you are running the CNC Console process.
2. Make sure the MySQL server hostname and port are correct and the server is accessible.
3. Restart the CNC Console installation or upgrade process.

If the problem continues, contact My Oracle Support for help. When you contact them, collect and share:

- The full hook job logs
- The CNC Console Custom Values file

4.38 IAM Hook Pod Logs Report "SQLException for query='${SQL_QUERY}'. Error Message: ${SQL_ERROR}. " Error

When you run the CNC Console installation or upgrade, you see an error in the pod logs. The logs show a message similar to:

```
[PRE-UPGRADE] Error during pre-upgrade process: [CREATE-SCHEMAS] Schema
creation failed for
        dbName:version ='${dbName}:${version}'. Reason: [RUN-SQL-SCRIPT]
SQLException for
        query='${SQL_QUERY}'. Error Message: ${SQL_ERROR}.
```

Where,

- `${dbName}` is the name of your CNC Console database.
- `${version}` is the CNC Console version.

For example:

```
[PRE-UPGRADE] Error during pre-upgarde process: [CREATE-SCHEMAS] Schema
creation failed for
        dbName:version ='cnccdb_tmp'. Reason: [RUN-SQL-SCRIPT] SQLException
for
        query='CREATE TABLE IF NOT - EXISTS KEYCLOAK_DML_TABLE (\n ID
varchar(36) NOT NULL,\n NAME
        varchar(36) NOT NULL,\n PRIMARY KEY ID)\n) ENGINE=ndbcluster DEFAULT
CHARSET=latin1'. Error
        Message: You have an error in your SQL syntax
```

Solution

You may encouter this error if

- The SQL script could not run because of driver errors, issues with the SQL syntax, or timeouts.
- The user running the script may not have enough privileges to create schemas or tables.
- The upgrade or installation may have been interrupted, which could leave the schema only partly applied.

To resolve this issue,

- Do not restart the installation, upgrade process, or perform a rollback.
- Contact My Oracle Support for help.

When you contact My Oracle Support, be sure to collect and provide the following:

- The complete hook job logs.
- The CNC Console Custom Values file used.

5 CNC Console Troubleshooting in OCI Deployment

This section provides information to troubleshoot the common errors which can be encountered when CNC Console is deployed in OCI.

5.1 CNC Console Deployment pods are not coming up

CNC Console Deployment pods are not coming up

This error can arise due to the following problems -

Problem 1: *CreateContainerConfigError* is observed in acore-ingress-gateway and mcore-ingress-gateway pods.

Solution: Cross check if **oci-iam-secret** is created or not. If not, create the secret properly as mentioned in *Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide.*

Problem 2: mcore-ingress-gateway pods is going to *CrashLoopBackOff* state.

Solution: Please check the occncc_custom_values.yaml file for the following -

1. Incorrect domain URL is provided in **global.mCncclams.fqdn** section of Instance configuration. This field should contain the URL of OCI Domain used to deploy CNC Console.

   ```
   mCnccIams:
       - id: Cluster1
         fqdn: idcs-37e739602b574bb6848c8cd96640f11d.identity.oraclecloud.com
         scheme: https
   ```

2. Scheme is missing in Instance Configuration of **global.mCncclams.scheme** User should provide **HTTPS** scheme.

Problem 3: *ImagePullBackOff* Error coming up for CNC Console Deployment pods.

Solution: All the image repositories must be public. Run the following steps to make all image repositories public:

1. Go to **OCI Console → Developer Services → Containers & Artifacts → Container Registry.**
2. Select the compartment.
3. In the **Repositories** and **Images Search** option, the images will be listed. Select each image and change to **Public**. This step must be preformed for all the images sequentially.

Problem 4: A *CreateContainerConfigError* is seen in cncc-mcore-validationhook.

Solution: Verify that the oci-iam-admin-secret has been created. If not, create the secret by following the procedure described in the *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide.*

5.2 CNC Console Core GUI is not loading

CNC Console Core GUI is not loading

Problem: Unable to access CNC Console Core GUI, getting "Invalid credentials" Error

This error can arise due to the following incorrect configurations -

- Incorrect Configuration 1: When **oci-iam-secret** is incorrect.
  Solution: Please make sure to create the secret properly as described in the *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide.*
- Incorrect Configuration 2: Incorrect **Issuer URL** is provided in OAuth settings of OCI IAM domain
  Solution: Follow the below steps to correct the Issuer URL in OAuth settings:
  1. Open the navigation menu and click **Identity & Security**. Under **Identity**, click **Domains**. Select the identity domain where CNC Console is deployed.
  2. Navigate to Security → OAuth
  3. Update the issuer URL with correct value (If the OCI IAM domain URL is with default port 80/443, make sure to remove the port from the domain URL while updating the **Issuer** field and it should match with what is provided in the **global.mCncclams.fqdn** section of occncc_custom_values.yaml).

5.3 Getting invalid redirect uri error while accessing CNC Console Core

Getting 'invalid redirect uri error' while accessing CNC Console Core

Problem: Getting the below error while accessing to CNC Console Core

```
{"error":"invalid_redirect_uri","error_description":"Client
c2151db267684a8fa15bcd19d56cd2a3 requested an invalid redirect URL: https://
100.xx.xx.xx/login/oauth2/code/cncc-iam. ECID:
      OZjBv13L900000000"}
```

Solution:

This problem is observed when incorrect redirect-uri is configured in the OAuth configuration of Integrated Application . Please follow the below steps -

1. Open the navigation menu and click **Identity & Security**. Under **Identity**, click **Domains**. Select the identity domain where CNC Console is deployed.
2. Click on **Integrated Applications**
3. Click on the Integrated Application that you want to modify. (Ex: **cncc-iam**)
4. Click **Edit OAuth Configuration**
5. Scroll down to find Redirect URL
6. Make sure to update the correct Redirect URL as per *Oracle Communications Cloud Native Configuration Console Installation, Upgrade, and Fault Recovery Guide.*

5.4 LDAP Integration Issues

LDAP Integration Issues

Problem 1: Not Able to find Delegate Authentication option.

Solution: Check for Type of Domain, it should be Premium for Delegate Authentication.

Problem 2: Not Able to find The Active Directory's (AD) Organization Unit (OU) for user and group in Oracle Cloud Infrastructure IAM (OCI IAM) Console.

Solution:

1. Check for the Organizational Unit in Active Directory.
2. If Organizational Unit is present in Active Directory.
3. Refresh the web Browser.

Problem 3: Not Able to Authenticate User using Active Directory user's password

Solution:

1. Make sure **Enable local authentication** is Enabled in **Authentication Settings** of OCI IAM.
2. Provide the required permission to Active Directory administrator. For more information see, the *Oracle Cloud Infrastructure Documentation.*

5.5 Issues in SAML Authentication

Issues in SAML Authentication

Problem: Logging in using SAML SSO throws following error: *No user was returned during the SAML assertion to user mapping for partner SSO*

Solution: In this case, recheck the SAML JIT configuration done in the **IDP** section of OCI IAM.

1. In the **Map User Attributes** section, check if the proper mapping is done between the IDP and Identity domain attributes.
2. If any of the IDP attribute mapped under this section is missing in SAML Assertion coming from IDP, then the authentication will fail.

5.6 CNC Console Access when private LoadBalancer IP is assigned

CNC Console Access when private LoadBalancer IP is assigned

Problem: CNC Console GUIs cannot be accessed when private LoadBalancer IP is assigned using the annotation oci-network-load-balancer.oraclecloud.com/internal: "true"

Solution: Access the CNC Console GUI via tunneling.

You can use any online emulator available for tunneling or you can follow the below steps mentioned

1. Run the following command for tunnelling using ssh command

```
ssh -v -f -N -i <id_rsa_private_key> -o StrictHostKeyChecking=no -o
ProxyCommand="ssh -i <id_rsa_private_key> -o StrictHostKeyChecking=no -W
%h:%p <user>@<bastion_ip>" <user>@<operator_instance_ip> -L
<local_port>:<worker_node_ip>:<loadbalancer_service_port> -o
ServerAliveInterval=60 -o ServerAliveCountMax=300
```

Example:

```
ssh -v -f -N -i id_rsa -o StrictHostKeyChecking=no -o ProxyCommand="ssh -i
id_rsa -o StrictHostKeyChecking=no -W %h:%p opc@10.xx.xx.xx"
opc@10.xx.xx.xx -L 443:10.xx.xx.xx:31152 -o ServerAliveInterval=60 -o
ServerAliveCountMax=300
```

where each attributes are

Table 5-1 Tunnelling Attributes

Attribute	Description
<id_rsa_private_key>	Specifies the private key to connect to the Bastion host.
<user>	username to connect to Bastion and Operator instances
<bastion_ip>	IP of the bastion host
<operator_instance_ip>	IP of the Operator instance
<worker_node_ip>	IP of the Worker node where CNC Console Application is installed
<local_port>	The port on your local machine where the browser is installed that requires forwarding from a remote host.
<loadbalancer_service_port>	Load Balancer service port of CNC Console, that is, M-CNCC Core ingress-gateway service

2. Open the /etc/hosts file.
For Mac
   a. Open the Terminal application on your Mac computer.
   b. Access the /etc/ hosts folder using a text editor (sudo user).

   For Windows
   a. Press the Windows key.
   b. Type Notepad in the search field.
   c. In the search results, right-click Notepad and select Run as administrator.
   d. From Notepad, open the following file: c:\Windows\System32\Drivers\etc\hosts
3. Add following entries under /etc/hosts file

   For M-CNCC Core

```
127.0.0.1 <cncc_release_name>-mcore-ingress-
```

```
gateway.<cncc_namespace>.svc.<cluster_domain>
```

Example:

```
127.0.0.1 cncc-mcore-ingress-gateway.cncc.svc.cluster.local
```

4. Access the CNC Console GUI using the above configured FQDN.

5.7 OCI IAM Connectivity Failure when Instance Level Access Control is Enabled

Problem: The operator faces database connectivity failure with the validation hook when the instance level access control feature is enabled.

Solution: When the connectivity is established again, operator must delete the INSTANCE_ALL role manually from the UI and run the installation for the validation hooks again to assign the INSTANCE_ALL role to the existing users.

5.8 403 Forbidden Error when Accessing NF Resources With Instance Level Access Control Enabled

Problem: A user with some required permissions encounters the 403 forbidden error when they try to acess any NF resources.

Solution: Check the acore-ingress-gateway pod logs to identify the missing permissions for the user.

6
CNC Console Alerts

This section provides information about CNC Console Alerts.

> **Note**
>
> **For OCI:**
> The only section applicable for OCI is CNC Console Alerts on OCI.

> **Note**
>
> Alert file is present in the Scripts directory. `occncc_csar_<version>.zip` can be downloaded from MOS. Unzip the file to get occncc_alertrules_<version>.yaml file.
>
> - Review the *occncc_alerting_rules_promha_<version>.yaml* file and edit the value of the parameters in the *occncc_alerting_rules_promha_<version>.yaml* file (if needed to be changed from default values) before configuring the alerts.
> - *kubernetes_namespace* is configured as kubernetes namespace in which CNCC is deployed. Default value is cncc. Update the *occncc_alertrules_<version>.yaml* file to reflect the correct CNCC kubernetes namespace.
>
> Two sample Alert files are provided, one for supporting CNE 1.8 or lower and second one supporting CNE Prometheus HA.
>
> - CNC Console Alert Rules file: *occncc_alertrules_<version>.yaml* file.
> - CNC Console Alert Rules file supporting CNE Prometheus HA: *occncc_alerting_rules_promha_<version>.yaml* file.

6.1 CNC Console IAM Alerts

This section provides information about CNC Console IAM Alerts.

6.1.1 CncclamTotalIngressTrafficRateAboveMinorThreshold

Table 6-1 CncclamTotalIngressTrafficRateAboveMinorThreshold

Field	Details
Trigger Condition	The total CNCC IAM Ingress Message rate has crossed the configured minor threshold of 700 TPS. Default value of this alert trigger point in *occncc_alertrules_<version>.yaml*is when CNCC IAM Ingress Rate crosses 70 % of 1000 (Maximum ingress request rate)
Severity	Minor

Table 6-1 (Cont.) CncclamTotalIngressTrafficRateAboveMinorThreshold

Field	Details
Alert details provided	**Description**: CNCC IAM Ingress traffic Rate is above the configured minor threshold i.e. 700 requests per second (current value is: {{ $value }}) **For CNE with Prometheus HA Operator:** **summary**: 'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . \| first \| value \| humanizeTimestamp }}{{ end }}: Traffic Rate is above 70 Percent of Max requests per second(1000)' **For CNE without Prometheus Operator :** **summary**: 'namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Traffic Rate is above 70 Percent of Max requests per second(1000)'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7001
Metric Used	oc_ingressgateway_http_requests_total
Resolution	The alert is cleared either when the total Ingress Traffic rate falls below the Minor threshold or when the total traffic rate crosses the Major threshold, in which case the *CncclamTotalIngressTrafficRateAboveMajorThreshold* alert is raised. **Note:** The threshold is configurable in the *occncc_alertrules_<version>.yaml* file. Steps: 1. Reassess why the CNCC IAM is receiving additional traffic. 2. If this is unexpected, contact My Oracle Support.

6.1.2 CncclamTotalIngressTrafficRateAboveMajorThreshold

Table 6-2 CncclamTotalIngressTrafficRateAboveMajorThreshold

Field	Details
Trigger Condition	The total CNCC IAM Ingress Message rate has crossed the configured major threshold of 800 TPS. Default value of this alert trigger point ino *ccncc_alertrules_<version>.yaml* is when CNCC IAM Ingress Rate crosses 80 % of 1000 (Maximum ingress request rate)
Severity	Major

Table 6-2 (Cont.) CncclamTotalIngressTrafficRateAboveMajorThreshold

Field	Details
Alert details provided	**Description**: 'CNCC IAM Ingress traffic Rate is above the configured major threshold i.e. 800 requests per second (current value is: {{ $value }})' **For CNE with Prometheus HA Operator:** **summary**: 'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . \| first \| value \| humanizeTimestamp }}{{ end }}: Traffic Rate is above 80 Percent of Max requests per second(1000)' **For CNE without Prometheus Operator :** **summary**: 'namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Traffic Rate is above 80 Percent of Max requests per second(1000)'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7001
Metric Used	oc_ingressgateway_http_requests_total
Resolution	The alert is cleared when the total Ingress Traffic rate falls below the Major threshold or when the total traffic rate crosses the Critical threshold, in which case the *CncclamTotalIngressTrafficRateAboveCriticalThreshold* alert is raised. **Note:** The threshold is configurable in the *occncc_alertrules_<version>.yaml* file. Steps: 1. Reassess why the CNCC IAM is receiving additional traffic. 2. If this is unexpected, contact My Oracle Support.

6.1.3 CncclamTotalIngressTrafficRateAboveCriticalThreshold

Table 6-3 CncclamTotalIngressTrafficRateAboveCriticalThreshold

Field	Details
Trigger Condition	The total CNCC IAM Ingress Message rate has crossed the configured critical threshold of 900TPS. Default value of this alert trigger point in *occncc_alertrules_<version>.yaml* is when CNCC IAM Ingress Rate crosses 90 % of 1000 (Maximum ingress request rate)
Severity	Critical

Table 6-3 (Cont.) CncclamTotalIngressTrafficRateAboveCriticalThreshold

Field	Details
Alert details provided	**Description**:CNCC IAM Ingress traffic Rate is above the configured critical threshold, that is, 900 requests per second (current value is: {{ $value }}) **For CNE with Prometheus HA Operator:** **summary**: 'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . \| first \| value \| humanizeTimestamp }}{{ end }}: Traffic Rate is above 90 Percent of Max requests per second(1000)' **For CNE without Prometheus Operator:** **summary**: 'namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Traffic Rate is above 90 Percent of Max requests per second(1000)'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7001
Metric Used	oc_ingressgateway_http_requests_total
Resolution	The alert is cleared when the Ingress Traffic rate falls below the Critical threshold. **Note:** The threshold is configurable in the *occncc_alertrules_<version>.yaml* file. Steps: 1. Reassess why the CNCC IAM is receiving additional traffic. 2. If this is unexpected, contact My Oracle Support.

6.1.4 CncclamMemoryUsageCrossedMinorThreshold

Table 6-4 CncclamMemoryUsageCrossedMinorThreshold

Field	Details
Trigger Condition	A pod has reached the configured minor threshold(70%) of its memory resource limits.
Severity	Minor
Alert details provided	**Description**: 'CNCC IAM 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.'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7002
Metric Used	container_memory_usage_bytes, kube_pod_container_resource_limits **Note:**This is a Kubernetes metric used for instance availability monitoring. If the metric is not available, use a similar metric as exposed by the monitoring system.

Table 6-4 (Cont.) CncclamMemoryUsageCrossedMinorThreshold

Field	Details
Resolution	The alert gets cleared when the memory utilization falls below the Minor Threshold or crosses the major threshold, in which case *CncclamMemoryUsageCrossedMajorThreshold* alert is raised. **Note:** The threshold is configurable in the *occncc_alertrules_<version>.yaml* file. If guidance is required, contact My Oracle Support.

6.1.5 CncclamMemoryUsageCrossedMajorThreshold

Table 6-5 CncclamMemoryUsageCrossedMajorThreshold

Field	Details
Trigger Condition	A pod has reached the configured major threshold(80%) of its memory resource limits.
Severity	Major
Alert details provided	**Description**: 'CNCC IAM 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.'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7002
Metric Used	container_memory_usage_bytes, kube_pod_container_resource_limits **Note:** This is a Kubernetes metric used for instance availability monitoring.If the metric is not available, use a similar metric as exposed by the monitoring system.
Resolution	The alert gets cleared when the memory utilization falls below the Major Threshold or crosses the critical threshold, in which case *CncclamMemoryUsageCrossedCriticalThreshold* alert shall be raised. **Note:** The threshold is configurable in the *occncc_alertrules_<version>.yaml* file. If guidance is required, contact My Oracle Support.

6.1.6 CncclamMemoryUsageCrossedCriticalThreshold

Table 6-6 CncclamMemoryUsageCrossedCriticalThreshold

Field	Details
Trigger Condition	A pod has reached the configured critical threshold (90%) of its memory resource limits
Severity	Critical

Table 6-6 (Cont.) CncclamMemoryUsageCrossedCriticalThreshold

Field	Details
Alert details provided	**Description**: 'CNCC IAM 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.'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7002
Metric Used	container_memory_usage_bytes, kube_pod_container_resource_limits **Note:** This is a Kubernetes metric used for instance availability monitoring. If the metric is not available, use a similar metric as exposed by the monitoring system.
Resolution	The alert gets cleared when the memory utilization falls below the Critical Threshold. Note : The threshold is configurable in the *occncc_alertrules_<version>.yaml* file. If guidance is required, contact My Oracle Support.

6.1.7 CncclamTransactionErrorRateAbove0.1Percent

Table 6-7 CncclamTransactionErrorRateAbove0.1Percent

Field	Details
Trigger Condition	The number of failed transactions is above 0.1 percent of the total transactions.
Severity	Warning
Alert details provided	**Description**: 'CNCC IAM transaction Error rate is above 0.1 Percent of Total Transactions (current value is {{ $value }})' **Summary**: 'CNCC IAM transaction Error Rate detected above 0.1 Percent of Total Transactions'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7003
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert is cleared when the number of failed transactions is below 0.1% of the total transactions or when the number of failed transactions crosses the 1% threshold in which case the *CncclamTransactionErrorRateAbove1Percent* is raised. Steps: 1. Check the Service specific metrics to understand the specific service request errors. 2. If guidance is required, contact My Oracle Support.

6.1.8 CncclamTransactionErrorRateAbove1Percent

Table 6-8 CncclamTransactionErrorRateAbove1Percent

Field	Details
Trigger Condition	The number of failed transactions is above 1 percent of the total transactions.
Severity	Warning
Alert details provided	**Description**: 'CNCC IAM transaction Error rate is above 1 Percent of Total Transactions (current value is {{ $value }})' **Summary**: 'CNCC IAM transaction Error Rate detected above 1 Percent of Total Transactions'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7003
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert is cleared when the number of failed transactions is below 1% of the total transactions or when the number of failed transactions crosses the 10% threshold in which case the CncclamTransactionErrorRateAbove10Percent is raised. Steps: 1. Check the Service specific metrics to understand the specific service request errors. 2. If guidance is required, contact My Oracle Support.

6.1.9 CncclamTransactionErrorRateAbove10Percent

Table 6-9 CncclamTransactionErrorRateAbove10Percent

Field	Details
Trigger Condition	The number of failed transactions is above 10 percent of the total transactions.
Severity	Minor
Alert details provided	**Description**: CNCC IAM transaction Error rate is above 10 Percent of Total Transactions (current value is {{ $value }})' **Summary**: 'CNCC IAM transaction Error rate is above 10 Percent of Total Transactions (current value is {{ $value }})'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7003
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert is cleared when the number of failed transactions is below 10% of the total transactions or when the number of failed transactions crosses the 25% threshold in which case the *CncclamTransactionErrorRateAbove25Percent* is raised. Steps: 1. Check the Service specific metrics to understand the specific service request errors. 2. If guidance is required, contact My Oracle Support.

6.1.10 CncclamTransactionErrorRateAbove25Percent

Table 6-10 CncclamTransactionErrorRateAbove25Percent

Field	Details
Trigger Condition	The number of failed transactions is above 25 percent of the total transactions.
Severity	Major
Alert details provided	**Description**: 'CNCC IAM transaction Error Rate detected above 25 Percent of Total Transactions (current value is {{ $value }})' **Summary**: 'CNCC IAM transaction Error Rate detected above 25 Percent of Total Transactions'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7003
Metric Used	oc_ingressgateway_http_responses_total
Resolution	TThe alert is cleared when the number of failed transactions are below 25% of the total transactions or when the number of failed transactions cross the 50% threshold in which case the *CncclamTransactionErrorRateAbove50Percent* is raised. Steps: 1. Check the Service specific metrics to understand the specific service request errors. 2. If guidance is required, contact My Oracle Support.

6.1.11 CncclamTransactionErrorRateAbove50Percent

Table 6-11 CncclamTransactionErrorRateAbove50Percent

Field	Details
Trigger Condition	The number of failed transactions is above 50 percent of the total transactions.
Severity	Critical
Alert details provided	**Description**: The number of failed transactions is above 50 percent of the total transactions. **Summary**: 'CNCC IAM transaction Error Rate detected above 50 Percent of Total Transactions'.
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7003
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert is cleared when the number of failed transactions is below 50 percent of the total transactions. The threshold is configurable in the *occncc_alertrules_<version>.yaml* file. Steps: 1. Check the Service specific metrics to understand the specific service request errors. 2. If guidance is required, contact My Oracle Support.

6.1.12 CncclamIngressGatewayServiceDown

Table 6-12 CncclamIngressGatewayServiceDown

Field	Details
Trigger Condition	The pods of the CNCC IAM Ingress Gateway microservice is available.
Severity	Critical
Alert details provided	**Description**:'CNCC IAM Ingress-Gateway service InstanceIdentifier=~".*cncc-iam_ingressgateway" is down' **For CNE with Prometheus HA Operator:** **summary**: 'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Ingress-gateway service down' **For CNE without Prometheus Operator:** **summary**: 'namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Ingress-gateway service down'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7004
Metric Used	'up' **Note:** This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.
Resolution	The alert is cleared when the cncc-iam_ingressgateway service is available. Steps: 1. Check the orchestration logs of cncc-iam_ingressgateway service and check for liveness or readiness probe failures. 2. Refer to the application logs on Kibana and filter based on cncc-iam_ingressgateway 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.

6.1.13 CncclamFailedLogin

Table 6-13 CncclamFailedLogin

Field	Details
Trigger Condition	The count of failed login attempts in CNCC-IAM by a user goes above '3'
Severity	Warning

Table 6-13 (Cont.) CncclamFailedLogin

Field	Details
Alert details provided	**Description**:'{{ $value }} failed Login attempts have been detected in CNCC IAM for user {{$labels.UserName}}, the configured threshold value is 3 failed login attempts for every 5 min' **For CNE with Prometheus HA Operator:** **summary**: 'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: failed login attempts are more than the configured threshold value' **For CNE without Prometheus Operator:** **summary**: 'namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: failed login attempts are more than the configured threshold value'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7005
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert gets cleared when the total failed login attempts for a particular user goes below the threshold value (default value is '3') in the last 5 min (default value is 5 m). **Note:** The threshold and time is configurable in the *alerts.yaml* file. If guidance is required, contact My Oracle Support.

6.1.14 AdminUserCreation

Table 6-14 AdminUserCreation

Field	Details
Trigger Condition	If a new admin account is created in the last 5 min
Severity	Warning
Alert details provided	**For CNE with Prometheus HA Operator:** **Description**: '{{ $value }} admin users have been created by {{$labels.UserName}} ' summary: 'namespace: {{$labels.namespace}} **summary**: {{$labels.pod}}, user: {{$labels.UserName}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Admin users have been created ' **For CNE without Prometheus Operator:** **Description**: '{{ $value }} admin users have been created by {{$labels.UserName}} ' summary: 'namespace: {{$labels.kubernetes_namespace}} **summary**: {{$labels.kubernetes_pod_name}}, user: {{$labels.UserName}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Admin users have been created '
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7006
Metric Used	oc_ingressgateway_http_requests_total

Table 6-14 (Cont.) AdminUserCreation

Field	Details
Resolution	The alert gets cleared when the total failed login attempts for a particular user go below the threshold value (default value is '3') in the last 5 min (default value is 5 m) **Note:** The threshold and time is configurable in the *occncc_alertrules_<version>.yaml* file. Login to admin GUI and review the user created. If guidance is required, contact My Oracle Support.

6.1.15 CncclamAccessTokenFailure

Table 6-15 CncclamAccessTokenFailure

Field	Details
Trigger Condition	If the count of failed token for CNCC-IAM goes above configured value of '3'
Severity	Warning
Alert details provided	**Description**: 'CNCC Iam Access Token Failure count is above the configured value i.e. 3 for every 5 min. Failed access token request count per second is (current value is: {{ $value }})' **For CNE with Prometheus HA Operator:** **summary**: 'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . \| first \| value \| humanizeTimestamp }}{{ end }}: Access Token Failure count is above the configured threshold value' **For CNE without Prometheus Operator:** **summary**: 'namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Access Token Failure count is above the configured threshold value'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.7007
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert gets cleared when the total failed tokens for a particular user go below the threshold value (default value is '3') in the last 5 min (default value is 5 m) **Note:** The threshold and time is configurable in the *occncc_alertrules_<version>.yaml* file. If guidance is required, contact My Oracle Support.

6.2 CNC Console Core Alerts

This section provides the information about CNC Console Core Alerts.

6.2.1 CnccCoreTotalIngressTrafficRateAboveMinorThreshold

Table 6-16 CnccCoreTotalIngressTrafficRateAboveMinorThreshold

Field	Details
Trigger Condition	The total CNCC Core Ingress Message rate has crossed the configured minor threshold of 700 TPS. Default value of this alert trigger point in cncc_alert_rules.yaml is when CNCC Core Ingress Rate crosses 70 % of 1000 (Maximum ingress request rate)
Severity	Minor
Alert details provided	**Description**: 'CNCC Core Ingress traffic Rate is above the configured minor threshold i.e. 700 requests per second (current value is: {{ $value }})' **For CNE with Prometheus HA Operator:** **summary**: 'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . \| first \| value \| humanizeTimestamp }}{{ end }}: Traffic Rate is above 70 Percent of Max requests per second(1000)' **For CNE without Prometheus Operator:** **summary**: 'namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Traffic Rate is above 70 Percent of Max requests per second(1000)'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8001
Metric Used	oc_ingressgateway_http_requests_total
Resolution	The alert is cleared either when the total Ingress traffic rate falls below the minor threshold or when the total traffic rate crosses the major threshold, in which case the *CnccCoreTotalIngressTrafficRateAboveMajorThreshold* alert is raised. **Note:** The threshold is configurable in the *occncc_alertrules_<version>.yaml* file. Steps: 1. Reassess why the CNCC Core is receiving additional traffic. 2. If this is unexpected, contact My Oracle Support.

6.2.2 CnccCoreTotalIngressTrafficRateAboveMajorThreshold

Table 6-17 CnccCoreTotalIngressTrafficRateAboveMajorThreshold

Field	Details
Trigger Condition	The total CNCC Core Ingress Message rate has crossed the configured major threshold of 800 TPS. Default value of this alert trigger point in cncc_alert_rules.yaml is when CNCC Core Ingress Rate crosses 80 % of 1000 (Maximum ingress request rate)
Severity	Major

Table 6-17 (Cont.) CnccCoreTotalIngressTrafficRateAboveMajorThreshold

Field	Details
Alert details provided	**Description**: 'CNCC Core Ingress traffic Rate is above the configured major threshold i.e. 800 requests per second (current value is: {{ $value }})' **For CNE with Prometheus HA Operator:** **summary**: 'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . \| first \| value \| humanizeTimestamp }}{{ end }}: Traffic Rate is above 80 Percent of Max requests per second(1000)' **For CNE without Prometheus Operator :** **summary**: 'namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Traffic Rate is above 80 Percent of Max requests per second(1000)'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8001
Metric Used	oc_ingressgateway_http_requests_total
Resolution	The alert is cleared when the total Ingress Traffic ratefalls below the Major threshold or when the total traffic rate cross the Critical threshold, in which case the CnccCoreTotalIngressTrafficRate Above CriticalThreshold. **Note:** The threshold is configurable in the *alerts.yaml* file. Steps: 1. Reassess why the CNCC Core is receiving additional traffic. 2. If this is unexpected, contact My Oracle Support.

6.2.3 CnccCoreTotalIngressTrafficRateAboveCriticalThreshold

Table 6-18 CnccCoreTotalIngressTrafficRateAboveCriticalThreshold

Field	Details
Trigger Condition	The total CNCC Core Ingress Message rate has crossed the configured critical threshold of 900TPS. Default value of this alert trigger point in cncc_alert_rules.yaml is when CNCC Core Ingress Rate crosses 90 % of 1000 (Maximum ingress request rate)
Severity	Critical

Table 6-18 (Cont.) CnccCoreTotalIngressTrafficRateAboveCriticalThreshold

Field	Details
Alert details provided	**Description**: 'CNCC Core Ingress traffic Rate is above the configured critical threshold i.e. 900 requests per second (current value is: {{ $value }})' **For CNE with Prometheus HA Operator:** **summary**: 'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . \| first \| value \| humanizeTimestamp }}{{ end }}: Traffic Rate is above 90 Percent of Max requests per second(1000)' **For CNE without Prometheus Operator :** **summary**: 'namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Traffic Rate is above 90 Percent of Max requests per second(1000)'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8001
Metric Used	oc_ingressgateway_http_requests_total
Resolution	The alert is cleared when the Ingress Traffic rate falls below the Critical threshold. **Note:** The threshold is configurable in the *occncc_alertrules_<version>.yaml* file. Steps: 1. Reassess why the CNCC IAM is receiving additional traffic. 2. If this is unexpected, contact My Oracle Support.

6.2.4 CnccCoreMemoryUsageCrossedMinorThreshold

Table 6-19 CnccCoreMemoryUsageCrossedMinorThreshold

Field	Details
Trigger Condition	A pod has reached the configured minor threshold(70%) of its memory resource limits.
Severity	Minor
Alert details provided	**Description**: 'CNCC Core 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.'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8002
Metric Used	container_memory_usage_bytes kube_pod_container_resource_limits 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 6-19 (Cont.) CnccCoreMemoryUsageCrossedMinorThreshold

Field	Details
Resolution	The alert gets cleared when the memory utilization falls below the Minor Threshold or crosses the major threshold, in which case CnccCoreMemoryUsageCrossedMajorThreshold alert is raised. **Note:** The threshold is configurable in the *occncc_alertrules_<version>.yaml* file. If guidance is required, contact My Oracle Support.

6.2.5 CnccCoreMemoryUsageCrossedMajorThreshold

Table 6-20 CnccCoreMemoryUsageCrossedMajorThreshold

Field	Details
Trigger Condition	A pod has reached the configured major threshold(80%) of its memory resource limits.
Severity	Major
Alert details provided	**Description**: 'CNCC Core 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.'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8002
Metric Used	container_memory_usage_bytes kube_pod_container_resource_limits 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.
Resolution	The alert gets cleared when the memory utilization falls below the Major Threshold or crosses the critical threshold, in which case CnccCoreMemoryUsageCrossedCriticalThreshold alert is raised **Note:** The threshold is configurable in the *occncc_alertrules_<version>.yaml* file. If guidance is required, contact My Oracle Support.

6.2.6 CnccCoreMemoryUsageCrossedCriticalThreshold

Table 6-21 CnccCoreMemoryUsageCrossedCriticalThreshold

Field	Details
Trigger Condition	A pod has reached the configured critical threshold (90%) of its memory resource limits
Severity	Critical

Table 6-21 (Cont.) CnccCoreMemoryUsageCrossedCriticalThreshold

Field	Details
Alert details provided	**Description**: 'CNCC Core 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.'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8002
Metric Used	container_memory_usage_bytes kube_pod_container_resource_limits 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.
Resolution	The alert gets cleared when the memory utilization falls below the Critical Threshold. **Note:** The threshold is configurable in the *occncc_alertrules_<version>.yaml* file. If guidance is required, contact My Oracle Support.

6.2.7 CnccCoreTransactionErrorRateAbove0.1Percent

Table 6-22 CnccCoreTransactionErrorRateAbove0.1Percent

Field	Details
Trigger Condition	The number of failed transactions is above 0.1 percent of the total transactions..
Severity	Warning
Alert details provided	**Description**:'CNCC Core transaction Error rate is above 0.1 Percent of Total Transactions (current value is {{ $value }})' **Summary**:'CNCC Core transaction Error rate is above 0.1 Percent of Total Transactions (current value is {{ $value }})'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8003
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert is cleared when the number of failed transactions are below 0.1% of the total transactions or when the number of failed transactions cross the 1% threshold in which case the *CnccCoreTransactionErrorRateAbove1Percent* is raised. The threshold is configurable in the *alerts.yaml* file. Steps: 1. Check the Service specific metrics to understand the specific service request errors. 2. If guidance is required, contact My Oracle Support.

6.2.8 CnccCoreTransactionErrorRateAbove1Percent

Table 6-23 CnccCoreTransactionErrorRateAbove1Percent

Field	Details
Trigger Condition	The number of failed transactions is above 1 percent of the total transactions.
Severity	Warning
Alert details provided	**Description**: 'CNCC Core transaction Error rate is above 1 Percent of Total Transactions (current value is {{ $value }})' **Summary**:'CNCC Core transaction Error Rate detected above 1 Percent of Total Transactions'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8003
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert is cleared when the number of failed transactions are below 1% of the total transactions or when the number of failed transactions crosses the 10% threshold in which case the *CnccCoreTransactionErrorRateAbove10Percent* is raised. Steps: 1. Check the Service specific metrics to understand the specific service request errors. 2. If guidance is required,contact My Oracle Support.

6.2.9 CnccCoreTransactionErrorRateAbove10Percent

Table 6-24 CnccCoreTransactionErrorRateAbove10Percent

Field	Details
Trigger Condition	The number of failed transactions is above 10 percent of the total transactions.
Severity	Minor
Alert details provided	**Description**: 'CNCC Core transaction Error rate is above 10 Percent of Total Transactions (current value is {{ $value }})' **summary**: 'CNCC Core ransaction Error Rate detected above 10 Percent of Total Transactions'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8003
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert is cleared when the number of failed transactions are below 10% of the total transactions or when the number of failed transactions crosses the 25% threshold in which case the *CnccCoreTransactionErrorRateAbove25Percent* is raised. Steps: 1. Check the Service specific metrics to understand the specific service request errors. 2. If guidance is required, contact My Oracle Support.

6.2.10 CnccCoreTransactionErrorRateAbove25Percent

Table 6-25 CnccCoreTransactionErrorRateAbove25Percent

Field	Details
Trigger Condition	The number of failed transactions is above 25 percent of the total transactions.
Severity	Major
Alert details provided	**Description**: 'CNCC Core transaction Error Rate detected above 25 Percent of Total Transactions (current value is {{ $value }})' **Summary**: 'CNCC Core transaction Error Rate detected above 25 Percent of Total Transactions'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8003
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert is cleared when the number of failed transactions are below 25% of the total transactions or when the number of failed transactions crosses the 50% threshold in which case the *CnccCoreTransactionErrorRateAbove50Percent* is raised. Steps: 1. Check the Service specific metrics to understand the specific service request errors. 2. If guidance is required, contact My Oracle Support.

6.2.11 CnccCoreTransactionErrorRateAbove50Percent

Table 6-26 CnccCoreTransactionErrorRateAbove50Percent

Field	Details
Trigger Condition	The number of failed transactions is above 50 percent of the total transactions.
Severity	Critical
Alert details provided	**Description**: 'CNCC Core transaction Error Rate detected above 50 Percent of Total Transactions (current value is {{ $value }})' **Summary**: 'CNCC Core transaction Error Rate detected above 50 Percent of Total Transactions'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8003
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert is cleared when the number of failed transactions are below 50 percent of the total transactions Steps: 1. Check the Service specific metrics to understand the specific service request errors. 2. If guidance is required, contact My Oracle Support.

6.2.12 CnccCoreIngressGatewayServiceDown

Table 6-27 CnccCoreIngressGatewayServiceDown

Field	Details
Trigger Condition	Cncc Core Ingress Gateway service is down
Severity	Critical
Alert details provided	**Description**: 'CNCC Core Ingress-Gateway service InstanceIdentifier=~".*core_ingressgateway" is down' **For CNE with Prometheus HA Operator:** **summary**: 'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Ingress-gateway service down' **For CNE without Prometheus Operator:** **summary**: 'namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }} : Ingress-gateway service down'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8004
Metric Used	'up' Note: This is a Prometheus metric used for instance availability monitoring. If this metric is not available, use a similar metric as exposed by the monitoring system.
Resolution	The alert is cleared when the cncc-core_ingressgateway service is available. Steps: 1. Check the orchestration logs of cncc-core_ingressgateway service and check for liveness or readiness probe failures. 2. Refer the application logs on Kibana and filter based on cncc-core_ingressgateway 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.

6.2.13 CnccCoreFailedLogin

Table 6-28 CnccCoreFailedLogin

Field	Details
Trigger Condition	The count of failed login attempts in CNCC-Core by a user goes above '3'
Severity	Warning

Table 6-28 (Cont.) CnccCoreFailedLogin

Field	Details
Alert details provided	**Description**:'{{ $value }} failed Login attempts have been detected in CNCC Core for user {{$labels.UserName}}, the configured threshold value is 3 failed login attempts for every 5 min' **For CNE with Prometheus HA Operator:** **summary**: 'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: failed login attempts are more than the configured threshold value' **For CNE without Prometheus Operator :** **summary**: 'namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: failed login attempts are more than the configured threshold value'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8005
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert gets cleared when the total failed login attempts for a particular user go below the threshold value (default value is '3') in the last 5 min (default value is 5 m) **Note:** The threshold and time is configurable in the *alerts.yaml* file. If guidance is required, contact My Oracle Support.

6.2.14 CnccCoreUnauthorizedAccess

Table 6-29 CnccCoreUnauthorizedAccess

Field	Details
Trigger Condition	The count of failed login attempts in CNCC-Core by a user goes above '3'
Severity	Warning
Alert details provided	**Description**:'{{ $value }} Unauthorized Accesses have been detected in CNCC-Core for {{$labels.ResourceType}} for {{$labels.Method}} request. The configured threshold value is 3 for every 5 min' **For CNE with Prometheus HA Operator:** **summary**: 'namespace: {{$labels.namespace}}, podname: {{$labels.pod}}, timestamp: {{ with query "time()" }}{{ . \| first \| value \| humanizeTimestamp }}{{ end }}: Unauthorized Access for CNCC-Core are more than threshold value' **For CNE without Prometheus Operator :** **summary**: 'namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Unauthorized Access for CNCC-Core are more than threshold value'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8006
Metric Used	oc_ingressgateway_http_responses_total

Table 6-29 (Cont.) CnccCoreUnauthorizedAccess

Field	Details
Resolution	The alert gets cleared when the total failed login attempts for a particular user go below the threshold value (default value is '3') in the last 5 min (default value is 5 m) **Note:** The threshold and time is configurable in the alerts.yaml If guidance is required, contact My Oracle Support.

6.2.15 CnccCoreAccessTokenFailure

Table 6-30 CnccCoreAccessTokenFailure

Field	Details
Trigger Condition	If the count of failed token for CNCC-Core goes above configured value of '3'
Severity	Warning
Alert details provided	**Description**: 'CNCC Core Access Token Failure count is above the configured value i.e. 3 for every 5 min. Failed access token request count per second is (current value is: {{ $value }})' **For CNE with Prometheus HA Operator:** **summary:** 'namespace: {{$labels.namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . \| first \| value \| humanizeTimestamp }}{{ end }}: Access Token Failure count is above the configured threshold value' **For CNE without Prometheus Operator :** **summary**: 'namespace: {{$labels.kubernetes_namespace}}, podname: {{$labels.kubernetes_pod_name}}, timestamp: {{ with query "time()" }}{{ . | first | value | humanizeTimestamp }}{{ end }}: Access Token Failure count is above the configured threshold value'
OID used for SNMP Traps	1.3.6.1.4.1.323.5.3.51.1.2.8007
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert gets cleared when the total failed tokens for a particular user go below the threshold value (default value is '3') in the last 5 min (default value is 5 m) **Note:** The threshold and time is configurable in the *alerts.yaml* file. If guidance is required, Contact My Oracle Support.

6.3 CNC Console Alerts on OCI

This section provides information about CNC Console Alerts on OCI:

6.3.1 CnccCoreTotalIngressTrafficRateAboveMinorThreshold

Table 6-31 CnccCoreTotalIngressTrafficRateAboveMinorThreshold

Field	Details
Trigger Condition	The total CNCC Core Ingress Message rate has crossed the configured minor threshold of 700 TPS. Default value of this alert trigger point in cncc_alert_rules.yaml is when CNCC Core Ingress Rate crosses 70 % of 1000 (Maximum ingress request rate)
Severity	minor
Alert details provided	CNCC Core Ingress traffic Rate is above the configured minor threshold i.e. 700 requests per second
Metric Used	oc_ingressgateway_http_requests_total
Resolution	The alert is cleared either when the total Ingress Traffic rate falls below the Minor threshold or when the total traffic rate cross the Major threshold, inwhich case the CnccCoreTotalIngressTrafficRateAboveMajorThresholdalert shall be raised.Note: The threshold is configurable in the alerts.yamlSteps:Reassess why the CNCC Core is receiving additionaltraffic.If this is unexpected, contact My Oracle Support.

6.3.2 CnccCoreTotalIngressTrafficRateAboveMajorThreshold

Table 6-32 CnccCoreTotalIngressTrafficRateAboveMajorThreshold

Field	Details
Trigger Condition	The total CNCC Core Ingress Message rate has crossed the configured major threshold of 800 TPS. Default value of this alert trigger point in cncc_alert_rules.yaml is when CNCC Core Ingress Rate crosses 80 % of 1000 (Maximum ingress request rate)
Severity	major
Alert details provided	CNCC Core Ingress traffic Rate is above the configured major threshold i.e. 800 requests per second
Metric Used	oc_ingressgateway_http_requests_total
Resolution	The alert is cleared when the total Ingress Traffic ratefalls below the Major threshold or when the total traffic rate cross the Critical threshold, in which casethe CnccCoreTotalIngressTrafficRateAboveCriticalThresholdNote: The threshold is configurable in the alerts.yaml alert shall be raised.Steps:Reassess why the CNCC Core is receiving additionaltraffic.If this is unexpected, contact My Oracle Support.

6.3.3 CnccCoreTotalIngressTrafficRateAboveCriticalThreshold

Table 6-33 CnccCoreTotalIngressTrafficRateAboveCriticalThreshold

Field	Details
Trigger Condition	The total CNCC Core Ingress Message rate has crossed the configured critical threshold of 900TPS. Default value of this alert trigger point in cncc_alert_rules.yaml is when CNCC Core Ingress Rate crosses 90 % of 1000 (Maximum ingress request rate)
Severity	critical
Alert details provided	CNCC Core Ingress traffic Rate is above the configured critical threshold i.e. 900 requests per second
Metric Used	oc_ingressgateway_http_requests_total
Resolution	The alert is cleared when the Ingress Traffic rate falls below the Critical threshold.Note: The threshold is configurable in the alerts.yamlSteps:Reassess why the CNCC Core is receiving additional traffic.If this is unexpected, contact My Oracle Support.

6.3.4 CnccCoreMemoryUsageCrossedMinorThreshold

Table 6-34 CnccCoreMemoryUsageCrossedMinorThreshold

Field	Details
Trigger Condition	A pod has reached the configured minor threshold(70%) of its memory resource limits.
Severity	minor
Alert details provided	CNCC Core Memory Usage for pod has crossed the configured minor threshold (70%) of its limit.
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.
Resolution	The alert gets cleared when the memory utilization falls below the Minor Threshold or crosses the major threshold, in which case CnccCoreMemoryUsageCrossedMajorThreshold alert shall be raised Note : The threshold is configurable in the alerts.yaml If guidance required, Contact My Oracle Support.

6.3.5 CnccCoreMemoryUsageCrossedMajorThreshold

Table 6-35 CnccCoreMemoryUsageCrossedMajorThreshold

Field	Details
Trigger Condition	A pod has reached the configured major threshold(80%) of its memory resource limits.
Severity	major

Table 6-35 (Cont.) CnccCoreMemoryUsageCrossedMajorThreshold

Field	Details
Alert details provided	CNCC Core Memory Usage for pod has crossed the configured major threshold (80%) of its limit.
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.
Resolution	The alert gets cleared when the memory utilization falls below the Major Threshold or crosses the critical threshold, in which case CnccCoreMemoryUsageCrossedCriticalThreshold alert shall be raised Note : The threshold is configurable in the alerts.yaml If guidance required, Contact My Oracle Support.

6.3.6 CnccCoreMemoryUsageCrossedCriticalThreshold

Table 6-36 CnccCoreMemoryUsageCrossedCriticalThreshold

Field	Details
Trigger Condition	A pod has reached the configured critical threshold (90%) of its memory resource limits
Severity	critical
Alert details provided	CNCC Core Memory Usage for pod has crossed the configured critical threshold (90%) of its limit.
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.
Resolution	The alert gets cleared when the memory utilization falls below the Critical Threshold. Note : The threshold is configurable in the alerts.yaml If guidance required, Contact My Oracle Support.

6.3.7 CnccCoreTransactionErrorRateAbovePointOnePercent

Table 6-37 CnccCoreTransactionErrorRateAbovePointOnePercent

Field	Details
Trigger Condition	The number of failed transactions is above 0.1 percent of the total transactions.
Severity	warning
Alert details provided	CNCC Core transaction Error rate is above 0.1 Percent of Total Transactions
Metric Used	oc_ingressgateway_http_responses_total

Table 6-37 (Cont.) CnccCoreTransactionErrorRateAbovePointOnePercent

Field	Details
Resolution	The alert is cleared when the number of failure transactions are below 0.1% of the total transactions or when the number of failure transactions cross the 1% threshold in which case the CnccCoreTransactionErrorRateAbove1Percent shall beraised.Steps:1. Check the Service specific metrics to understand the specific service request errors.2. If guidance required, contact My Oracle Support.

6.3.8 CnccCoreTransactionErrorRateAboveOnePercent

Table 6-38 CnccCoreTransactionErrorRateAboveOnePercent

Field	Details
Trigger Condition	The number of failed transactions is above 1 percent of the total transactions.
Severity	warning
Alert details provided	CNCC Core transaction Error rate is above 1 Percent of Total Transactions
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert is cleared when the number of failure transactions are below 1% of the total transactions or when the number of failure transactions cross the 10% threshold in which case the CnccCoreTransactionErrorRateAbove10Percent shall beraised.Steps:1. Check the Service specific metrics to understand the specific service request errors.2. If guidance required, contact My Oracle Support.

6.3.9 CnccCoreTransactionErrorRateAboveTenPercent

Table 6-39 CnccCoreTransactionErrorRateAboveTenPercent

Field	Details
Trigger Condition	The number of failed transactions is above 10 percent of the total transactions.
Severity	minor
Alert details provided	CNCC Core transaction Error rate is above 10 Percent of Total Transactions
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert is cleared when the number of failure transactions are below 10% of the total transactions or when the number of failure transactions cross the 25% threshold in which case the CnccCoreTransactionErrorRateAbove25Percent shall beraised.Steps:1. Check the Service specific metrics to understand the specific service request errors.2. If guidance required, contact My Oracle Support.

6.3.10 CnccCoreTransactionErrorRateAboveTwentyFivePercent

Table 6-40 CnccCoreTransactionErrorRateAboveTwentyFivePercent

Field	Details
Trigger Condition	The number of failed transactions is above 25 percent of the total transactions.
Severity	major
Alert details provided	CNCC Core transaction Error Rate detected above 25 Percent of Total Transactions
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert is cleared when the number of failure transactions are below 25% of the total transactions or when the number of failure transactions cross the 50% threshold in which case the CnccCoreTransactionErrorRateAbove50Percent shall beraised.Steps:1. Check the Service specific metrics to understand the specific service request errors.2. If guidance required, contact My Oracle Support.

6.3.11 CnccCoreTransactionErrorRateAboveFiftyPercent

Table 6-41 CnccCoreTransactionErrorRateAboveFiftyPercent

Field	Details
Trigger Condition	The number of failed transactions is above 50 percent of the total transactions.
Severity	critical
Alert details provided	CNCC Core transaction Error Rate detected above 50 Percent of Total Transactions
Metric Used	oc_ingressgateway_http_responses_total
Resolution	The alert is cleared when the number of failure transactions are below 50 percent of the total transactions.Steps:1. Check the Service specific metrics to understand the specific service request errors.2. If guidance required, contact My Oracle Support.

6.3.12 CnccCoreUnauthorizedAccess

Table 6-42 CnccCoreUnauthorizedAccess

Field	Details
Trigger Condition	If the count of unauthorized access goes above the configured value of '3'
Severity	warning
Alert details provided	Unauthorized Accesses have been detected in CNCC-Core for request. The configured threshold value is 3 for every 5 min
Metric Used	oc_ingressgateway_http_responses_total

Table 6-42 (Cont.) CnccCoreUnauthorizedAccess

Field	Details
Resolution	The alert gets cleared when the total unauthorized accesses for a particular user go below the threshold value (default value is '3') in the last 5 min (default value is 5 m) Note : The threshold and time is configurable in the alerts.yaml If guidance required, Contact My Oracle Support.