Oracle® Communications
Cloud Native Core, OCI Adaptor User Guide

Release 24.3.0

ORACLE®

Oracle Communications Cloud Native Core, OCI Adaptor User Guide, Release 24.3.0

G14586-01

Contents

Acronyms

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

Table   Acronyms and Terminologies

Acronyms	Description
APM	Application Performance Monitoring
CIDR	Classless Inter-Domain Routing
DRG	Dynamic Routing Gateway
EGW	Egress Gateway
IGW	Ingress Gateway
LA	Logging Analytics
NF	A 3GPP-defined processing function in a network, which has defined functional behaviour and 3GPP-defined interfaces.
NF service	It is a functionality exposed by an NF through a service-based interface and consumed by other authorized NFs.
NF service instance	An identifiable instance of the NF service
NLB	Network Load Balancer
OCI	Oracle Cloud Infrastructure
OCI Adaptor	OCI Adaptor components conduit between CNC NFs and OCI's observability services.
OKE	Oracle Engine for Kubernetes
OTEL	OpenTelemetry
Private VM	A Private Virtual Machine is a virtual machine hosted on-premise within a private network or infrastructure, usually owned and managed by an organization.
Public VM	A Public Virtual Machine is a virtual machine hosted on a public cloud infrastructure, such as Amazon Web Services (AWS), Microsoft Azure, and so on.
RDP	Remote Desktop Protocol
RPC	Remote Peering Connection
SSH	Secure Shell
Terraform	Infrastructure as Code (IaC) software tool to provision and manage infrastructure in any cloud.
VCN	Virtual Cloud Network

What's New in This Guide

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

Release 24.3.0- G14586-01, October 2024

There are no updates made to this document.

Release 24.2.0- G12401-02, August 2024

The following change is made in this release:

- Moved the [Enable NF Observability](#) section from *Oracle Communications Cloud Native Core, OCI Deployment Guide*.

Release 24.2.0- G12401-01, July 2024

The following changes are made in this release:

- Added **Compartment Admin** in the [User or Group Management](#) section for creating OCI infrastructure using terraform scripts.
- Replaced the "Bastion host VM" with "Bastion service" in the entire document for session based changes.

Release 24.1.0- F92376-01, April 2024

This is the initial release of the document.

1
Introduction

This document contains information about the functionalities of Cloud Native Core Oracle Cloud Infrastructure (OCI) Adaptor and how to use it for integrating the OCI Observability stack with CNC NFs on OCI.

The information in this document provides the users to:

- Deploy the OCI Adaptor on the public cloud OCI platform.
- Deploy the NFs in OCI.
- Collect logs, metrics, and traces of CNC NFs and share them with OCI PaaS (Platform as a Service) using OCI Adaptor.

2
Enable NF Observability

The following are the five essential components necessary to enable NF on OCI:

- OCI Adaptor
- OCI Infrastructure Layer
- User Management
- OCI Observability and Management
- CNC Applications (Cloud Native Core Network Functions, cnDBTier, CNC Console)

The following diagram represents the OCI Adaptor on OCI:

Figure 2-1   OCI Adaptor on OCI
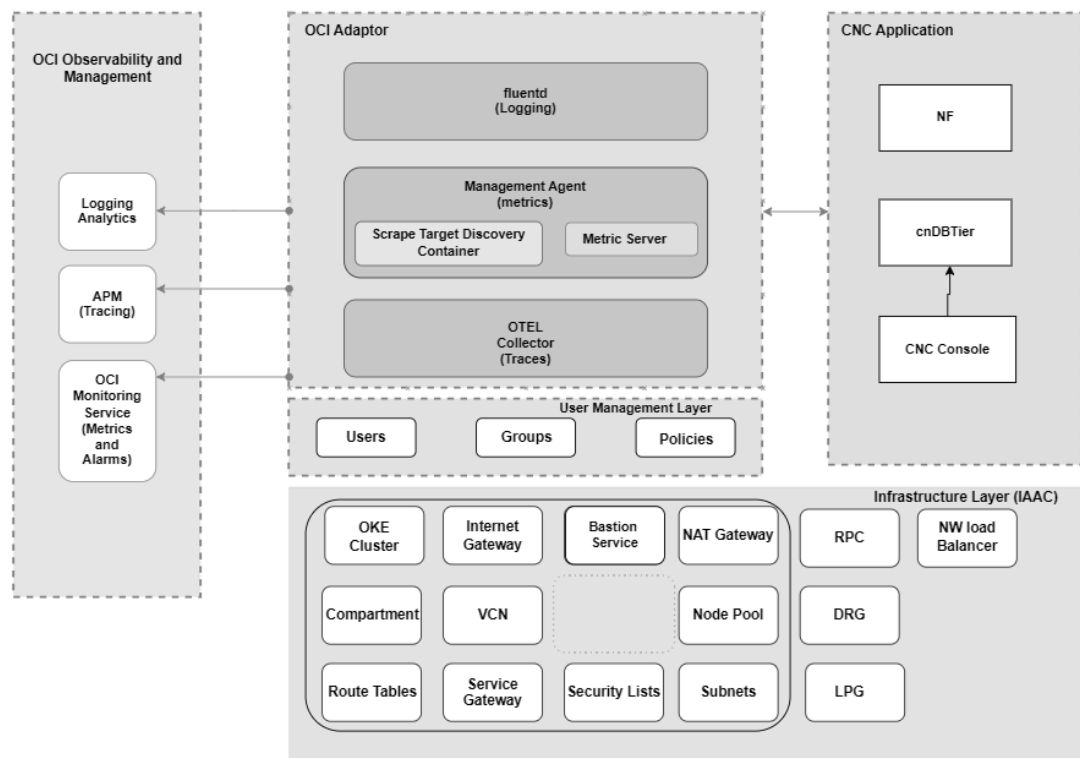
> ⓘ **Note**
>
> • OCI Adaptor and CNC Applications (NFs, cnDBTier, and CNC Console) are deployed on a single OKE cluster.
>
> • The user must have OCI tenancy and CNC Applications with OCI Adaptor to deploy the CNC NF on the OCI.
>
> • The user can create a new compartment or use an existing compartment to deploy the OCI components.

OCI Adaptor

The OCI Adaptor acts as a channel to transfer information between the application and OCI observability management. It assists CNC NFs in achieving observability and monitoring functions on the public cloud OCI platform.

The OCI Adaptor components include:

• Fluentd: It is a open source data collector software to collect log data.

• Management Agent: Management Agent, Scrape Target Discovery Container, and Metric Server are the components used to derive metrics.
  This component consists of the following subcomponents:

  – OCI Management Agent

  – Scrape Target Discovery Container

  – Metric server

• OpenTelemetry (OTEL) Collector: OpenTelemetry (OTEL) Collector is the component to collect traces.
  For more information about deploying the components, see the "Deploying OCI Adaptor" section in *Oracle Communications Cloud Native Core, OCI Deployment Guide*.

Fluentd

OCIs customized Fluentd is responsible for collecting the log data from the input sources, transforming the logs, and routing the log data to the OCI Logging Analytics (LA) service.

For more information about the Fluentd, see the "Introduction" section in *Ingest logs to OCI Logging Analytics using Fluentd Documentation*.

A dynamic group is created, allowing Fluentd to upload the logs to the OCI LA service (part of the OCI Observability Management layer) using the terraform scripts based on the following matching rule and policy.

For more information about deploying Fluentd, see the "Terraform Scripts for OCI Deployment" section in *Oracle Communications Cloud Native Core, OCI Deployment Guide*.

```
Matching Rule:
ALL {instance.compartment.id = '<CNC NF's compartment_ocid>'}
Policy:
Allow dynamic-group DYNAMIC_GROUP_NAME to
{LOG_ANALYTICS_LOG_GROUP_UPLOAD_LOGS} in compartment COMPARTMENT_NAME
```

For more information, see the "Managing Dynamic Groups" section in *Oracle Cloud Infrastructure Documentation*.

Management Agent, Scrape Target Discovery Container, and Metric Server

This section provides the information about the management agent, scrape target discovery container, and metric server. This is used for metrics.

Management Agent

The Management Agent is the central entity that collects metrics data from different services and sources that need to be monitored and uploads it into the OCI Monitoring Service over the REST interface.

For more information, see the "Management Agents" section in *Oracle Cloud Infrastructure Documentation*.

A dynamic group is created that allows the Management Agent to upload metrics data to OCI Monitoring Service using the terraform scripts based on the following matching rule and policy.

For more information, see the "Terraform Scripts for OCI Deployment" section in *Oracle Communications Cloud Native Core, OCI Deployment Guide*.

```
Matching Rule:
ALL {resource.type='managementagent',
resource.compartment.id='<AGENT_COMPARTMENT_OCID>'}
Policy:
allow dynamic-group DYNAMIC_GROUP_NAME to use metrics in compartment
<PROMETHEUS_METRIC_COMPARTMENT_NAME>
```

Scrape Target Discovery Container

Management Agent collects the metrics data as per the information of different sources provided to it.

The job of Scrape Target Discovery Container is to dynamically update the metrics source information for the Management Agent. Accordingly, the Management Agent can fetch the correct metric data.

Metric Server

It is required to collect `cAdvisor` metrices. `cAdvisor` provides various insights into containers.

CNC requires these details to monitor the health and performance of the containers running on a given node. For example, `container_memory_usage_bytes, container_tasks_state`.

OTEL Collector

OpenTelemetry (OTEL) is an observability framework which is used to capture and export traces. CNC applications use OpenTelemetry to create spans or traces. A trace is a collection of spans connected in a parent child relationship.

The traces specify how APIs or requests are propagated through the microservices and other components. OpenTelemetry Collector is used to collect traces from the NFs and send them to OCI Application Performance Monitoring (APM).

For more information, see the [Application Performance Monitoring (APM) (Tracing)](link).

User Management Layer

The user management layer allows you to create user groups and policies in a domain. There can be multiple identity domains in a tenancy:

- Default domain: Available by default and should only have limited number of admins who have administrative privileges on the tenancy level.

- Non-Default domains: All other application users.

The user needs to set up user groups in a domain.

There are three types of user groups:

- Tenancy Admin

- Compartment Admin

- Non-Admin Users

User or Group Management

The user can setup either of the following user groups to create OCI Infrastructure:

- Tenancy Admin

- Compartment Admin

Only `Tenancy Admin` can install OCI Adaptor using terraform scripts.

A `Tenancy Admin` can create two user groups. One group is assigned with compartment administrative privileges, and the other is assigned non-administrative privileges.

Dynamic Group Management

OCI compute instances can be grouped using dynamic groups to better access OCI Platform as a service (PaaS) services.

For more information, see the "Managing Dynamic Groups" section in *Oracle Cloud Infrastructure Documentation*.

OCI Observability and Management

The OCI Observability and Management service provides dashboards and explorers for observing the metrics, logs, and traces of CNC Applications. These dashboards and explorers allow you to view, query, and study metrics, logs, and traces.

The Observability and Management module includes:

- OCI Monitoring Service (Metrics and Alarms)

- Application Performance Monitoring (APM) (Tracing)

- Logging Analytics

Integration of OCIs observability and management with NF applications is achieved using OCI Adaptor.

OCI Monitoring Service (Metrics and Alarms)

Management Agent of OCI Adaptor fetches metrics from the scrape targets and publishes them towards the OCI Monitoring Service.

The following diagram briefly outlines the integration and use of Management Agent in fetching the metrics from the targets and publishing the same towards Monitoring Service.

Figure 2-2   OKE Cluster Monitoring with Management Agent
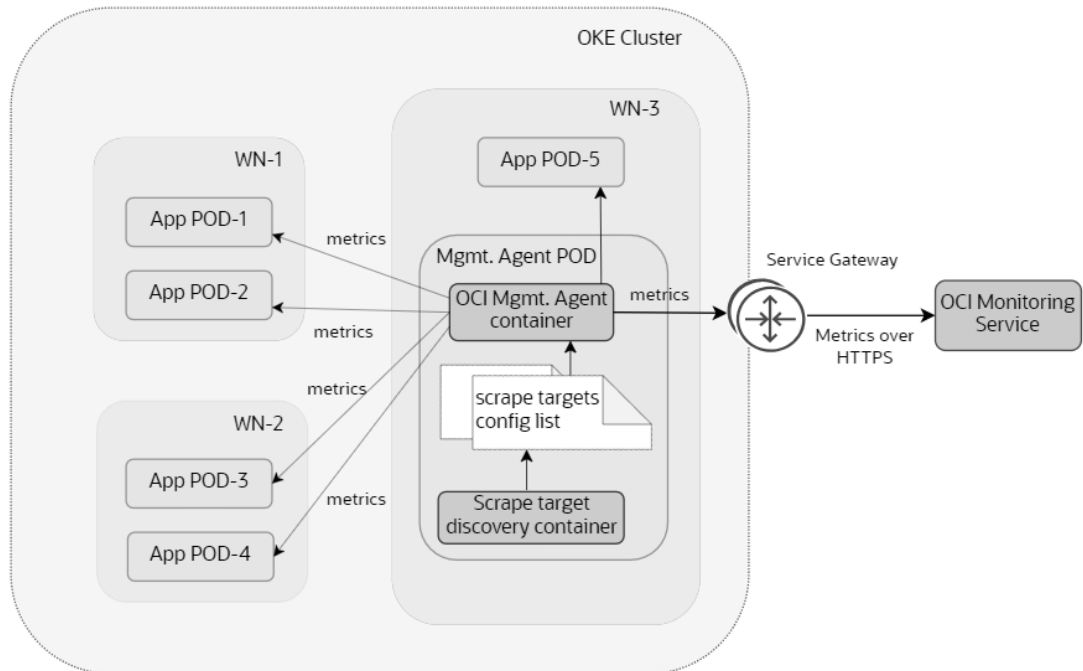
The Oracle Cloud Infrastructure (OCI) Monitoring service actively and passively monitors NF applications and cloud resources using the metrics and alarms features. CNC shares metrics with the OCI Monitoring service. These metrics help render various dashboards on OCI that eventually help monitor the CNC Applications. These metrics also assist in generating alerts for the customer. The management agent collects the metrics data from different services and sources that have to be monitored and uploads it to the OCI Monitoring Service.

For more information, see the "Monitoring" section in *Oracle Cloud Infrastructure Documentation*.

> ⓘ **Note**
>
> To view metrics and create the alarms, write the metrics queries using the Monitoring Query Language (MQL) syntax.
> For more information, see the "Monitoring Query Language (MQL) Reference" section in *Oracle Cloud Infrastructure Documentation*.

Viewing the Metrics

The user can view the metrics using the Metrics Explorer option in the OCI Monitoring Service.

To access the metrics explorer:

1. Log in to **OCI Console**.
   For more information, see "Signing In to the OCI Console" section in *Oracle Cloud Infrastructure Documentation*.

2. Open the navigation menu and click **Observability and Management**.

3. Under **Monitoring**, click **Metrics Explorer**. The **Metrics Explorer** screen appears.

4. The user can customize the metrics view by choosing the customizations options.

For more information, see the "Creating a Basic Query" section in *Oracle Cloud Infrastructure Documentation*.

Creating the Alarms or Alerts

The user can create the alarms using the OCI Monitoring Service.

For more information, see the "Configuring NF Alerts on OCI" section in *Oracle Communications Cloud Native Core, OCI Deployment Guide*.

Creating the Alarms

The user can create the alarms in the OCI Monitoring Service.

To create the alarms:

1. Log in to **OCI Console**.
   For more information, see the "Signing In to the OCI Console" section in *Oracle Cloud Infrastructure Documentation*.

2. Open the navigation menu and click **Observability and Management**.

3. Under **Monitoring**, click **Alarm Definitions**.

4. Click **Create Alarm**. The **Create Alarm** page opens in **Basic mode** (the default view). The following steps describe how to create an alarm in **Basic mode**:

   a. Enter a user-friendly name for the alarm. This name is sent as the title for notifications related to this alarm.

   b. For **Alarm severity**, select the perceived type of response required when the alarm is in the firing state.

   c. For **Alarm body**, provide user-readable notification content.

5. The user can customize the alarms by choosing the provided customizations options.

For more information, see the "Managing Alarms" section in *Oracle Cloud Infrastructure Documentation*.

Application Performance Monitoring (APM) (Tracing)

The OCI Application Performance Monitoring (APM) is a service that provides deep visibility into the performance of applications and offers the ability to diagnose issues quickly. APM collects and processes the transaction instance trace data using Application Performance Monitoring data sources, open source tracers, or directly using API. It accepts OpenTelemetry

spans and combines them into traces. CNC Applications microservices share spans or traces towards the OCI APM that help troubleshoot the application.

CNC Applications use OpenTelemetry to create spans or traces. A trace is a collection or list of spans connected in a parent or child relationship.

Traces specify how APIs or requests are propagated through the microservices and other components, assisting operations and developers in troubleshooting issues in a customer deployment.

OCI Application Performance Monitoring (APM) provides a comprehensive set of features to monitor applications and diagnose performance issues.

For more information, see the "Application Performance Monitoring" section in *Oracle Cloud Infrastructure Documentation*.

Figure 2-3   Trace Collection towards OCI APM Data Collector
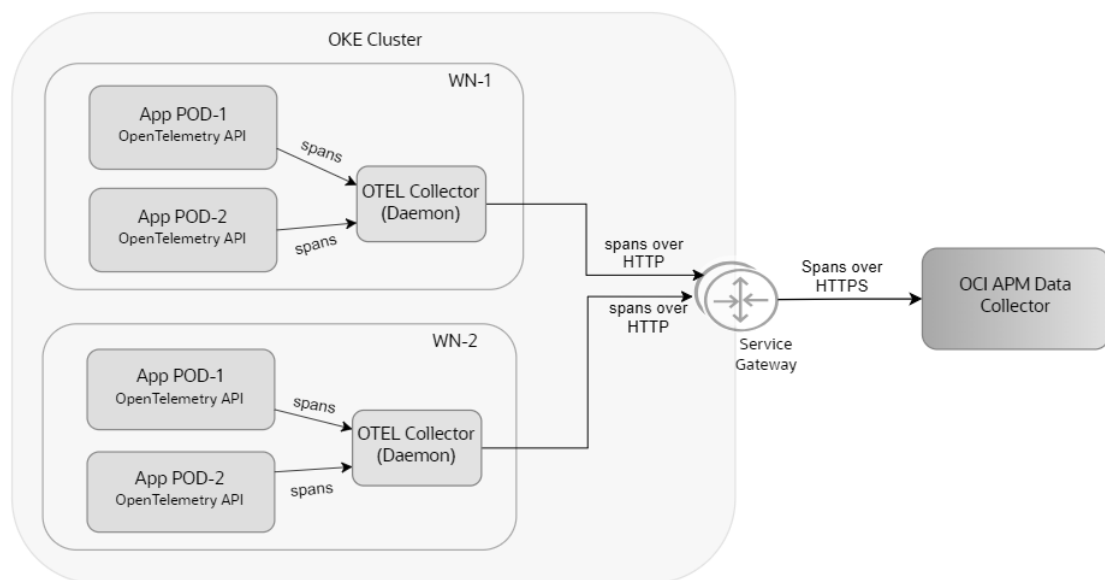
Monitoring the Traces

The user can monitor the traces using the **Trace Explorer** option in the Application Performance Monitoring service.

To access the Trace Explorer:

1. Log in to **OCI Console**.
   For more information, see the "Signing In to the OCI Console" in *Oracle Cloud Infrastructure Documentation*.

2. Open the navigation menu and click **Observability and Management**.

3. Under **Application Performance Monitoring**, click **Trace Explorer**.

4. The **Tracing Explorer** screen appears.

5. The user can customize the view of traces view by choosing the different options available.

For more information, see the "Monitor Traces in Trace Explorer" section in *Oracle Cloud Infrastructure Documentation*.

Logging Analytics

OKE cluster environment on a OCI tenancy comprises of three tiers from which logs are generated:

- Infrastructure tier comprising of worker nodes, networking resources.
- Kubernetes platform tier comprising of Kubelet, Core DNS and so on.
- Application tier comprising of applications and DB applications.

Oracle Cloud Infrastructure (OCI) provides two logging services, namely OCI Logging and OCI Logging Analytics.

Oracle Cloud Logging Analytics (LA), a cloud solution in OCI assists in indexing, aggregating, visualizing, searching and also monitoring all log data from your applications and system infrastructure.

OCI Logging Analytics supports integration or log ingestion by applications using either Management Agent or Fluentd. However, OCI recommends log ingestion using Fluentd.

Figure 2-4   Fluentd Plugin Architecture
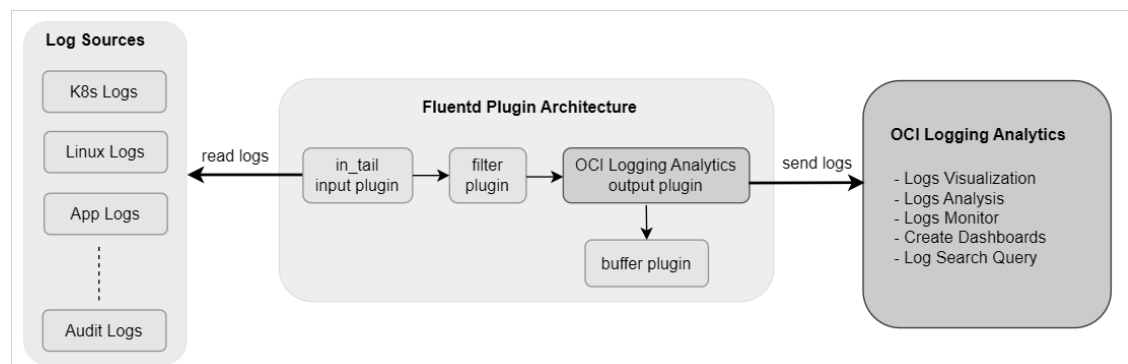
The CNC shares logs towards OCI LA using Fluentd, the OCI LA is used to perform log analysis.

For more information, see the "Logging Analytics" section in *Oracle Cloud Infrastructure Documentation*.

Visualizing the Logs

The user can view the logs using the **Log Explorer** option in the Logging Analytics.

To access the Log Explorer:

1. Log in to **OCI Console**.
   For more information, see the "Signing In to the OCI Console" section in *Oracle Cloud Infrastructure Documentation*.

2. Open the navigation menu and click **Observability and Management**.

3. Under **Logging Analytics**, Click **Log Explorer**. The **Log Explorer** screen appears.

4. The user can customize the **Logs** view by choosing the **Visualizations** options.

For more information, see the "Visualize Data Using Charts and Controls" section in *Oracle Cloud Infrastructure Documentation*.

Infrastructure Layer (IAAC)

Oracle Cloud Infrastructure (OCI) provides high-performance computing capabilities (such as physical hardware instances) and storage capacity in a flexible overlay virtual network that is securely accessible from your on-premise network.

Oracle Cloud Infrastructure (OCI) consists of Compartments, a Network Load balancer, a Bastion Host, a Dynamic Routing Gateway (DRG), a Remote Peering Connection (RPC), a Service Gateway, an Internet Gateway, and an OKE cluster.

OKE Cluster

Oracle Kubernetes Engine (OKE) offers a carrier-grade container orchestration platform (based on Kubernetes) that supports the deployment requirements of cloud-native, containerized applications.

OKE is a fully managed and scalable service designed for deploying containerized applications in the cloud. It ensures high availability and reliability of cloud native applications. OKE utilizes Kubernetes to automate the deployment, scaling, and management of containerized applications.

OKE provides two types of clusters:

- Basic cluster: Basic clusters support all the core functionalities provided by Kubernetes and Container Engine for Kubernetes, but none of the enhanced features that Container Engine for Kubernetes provides.

- Enhanced cluster: Enhanced clusters support all the available features, such as virtual nodes, cluster add-on management, workload identity, and additional worker nodes per cluster.

CNC Applications can be deployed on basic or enhanced OKE clusters (through the terraform scripts) with managed nodes. Since the NFs cannot be deployed on virtual nodes and virtual node pools, enhanced cluster is not required for CNC NF deployment. However, the user can always upgrade any basic cluster to an enhanced cluster but enhanced cluster cannot be downgraded.

For more information, see the "Working with Enhanced Clusters and Basic Clusters" section in *Oracle Cloud Infrastructure Documentation*.

Compartment

A compartment is a logical resource that helps achieve security isolation and controls resource access. CNC utilizes this resource to group CNC NF and its corresponding OCI resources and assigns users or groups with policies to access these resources, thus isolating access from the rest of the OCI tenancy's users or groups.

When a tenancy is created in Oracle Cloud Infrastructure (OCI), a root compartment is automatically provisioned, which holds all the cloud resources. The root compartment is the top-level folder for organizing and managing resources within the tenancy, similar to a root folder in a file system. Compartments are also tenancy-wide resources used to help organize and isolate cloud resources.

They enable a hierarchical structure for the tenancy resources and allow better management and control of access. The tenancy explorer provides a complete view of all cloud resources in a specific compartment across all regions.

For more information, see the "Viewing All Resources in a Compartment" section in *Oracle Cloud Infrastructure Documentation*.

When a compartment is created, it is assigned a unique identifier called an Oracle Cloud ID (OCID).

For more information, see the "OCID Resource Identifiers" section in *Oracle Cloud Infrastructure Documentation*.

The subcompartments inherit access permissions from the compartments higher up in the hierarchy.

For more information, see the "Managing Compartments" section in *Oracle Cloud Infrastructure Documentation*.

Virtual Cloud Network

Virtual Cloud Network (VCN) is a networking service that helps create and manage a customizable and private network in the OCI cloud. A VCN service includes subnets, route tables, and security lists. CNC NF on OCI uses separate subnets for Bastion Service, CLI Server, Worker Nodes, and so on. Security lists define the Ingress and Egress rules for a subnet. Route tables assist in routing to a different network through gateways, such as sending CNC NF metrics data to the OCI Monitoring service.

For more information, see the "Networking Overview" in *Oracle Cloud Infrastructure Documentation*.

Network Load Balancer

The network load balancer enables creating network load balancers in Virtual Cloud Network (VCN) for reliable traffic balancing. Load balancers can be public or private. Multiple listeners can balance Layer 4 traffic. Both public and private load balancers can route traffic to any backend server within the Virtual Cloud Network (VCN).

To enhance the security of the network load balancer and make it less vulnerable to external threats, you can create a private network load balancer. By doing so, the network load balancer will be assigned a private IP address which will act as the entry point for all incoming traffic.

This private network load balancer can only be accessed within the VCN that includes the host regional subnet, or as per the restrictions set by your security rules.

> ⓘ **Note**
>
> CNC NF integration with OCI utilize the Private Network Load Balancer.

For more information, see the "Introduction to Network Load Balancer" section in *Oracle Cloud Infrastructure Documentation*.

The OCI Network Load Balancer (NLB) provides flexible automated traffic distribution from one entry point to multiple backend servers in your VCN. It operates at the connection level and load balances incoming client connections to healthy backend servers based on Layer 3 or Layer 4 (IP protocol) data.

CNC uses the following LB annotations to route the incoming request to the appropriate backend servers:

- `oci-network-load-balancer.oraclecloud.com/internal: "false"`

- `oci-network-load-balancer.oraclecloud.com/security-list-management-mode: All`

- `oci-network-load-balancer.oraclecloud.com/subnet: <your-lb-subnet-ocid>`

- `oci.oraclecloud.com/load-balancer-type: nlb`

For more information, see the "Introduction to Network Load Balancer" section in *Oracle Cloud Infrastructure Documentation*.

Dynamic Routing Gateway

A Dynamic Routing Gateway (DRG) acts as a virtual router, providing a path for traffic to flow between your on-premise networks and VCNs and can also route traffic between VCNs. CNC NF uses OCI DRG to send and receive messages to NFs residing on the customer's on-premise network and towards NFs residing in a different OCI Region. Each DRG attachment has an associated route table to route packets entering the DRG to their next hop.

For more information, see the "Dynamic Routing Gateways" section in *Oracle Cloud Infrastructure Documentation*.

Bastion Service

Bastions let authorized users to connect from specific IP addresses to target resources using Secure Shell (SSH) sessions. When connected, users can interact with the target resource by using any software or protocol supported by SSH. For example, you can use the Remote Desktop Protocol (RDP) to connect to a Windows host, or use Oracle Net Services to connect to a database.

The Bastion Service allows users to reach the OKE cluster using the CLI Server. Using the Bastion Session, User can SSH into the CLI Server.

> ⓘ **Note**
>
> Copy all the CNC software to CLI Server for installation.

Creating the Bastion Service Session

Bastion service is deployed as a part of the infrastructure terraform. User must create a `Bastion Service Session` to login to `CLI Server`.

Following are the steps to create the Bastion Service Session:

1. Log in to the **OCI Console**.
   For more information, see the "Signing In to the OCI Console" section in *Oracle Cloud Infrastructure Documentation*.

2. Open the navigation menu and select **Identity and Security** from the hamburger menu and select **Bastion**.
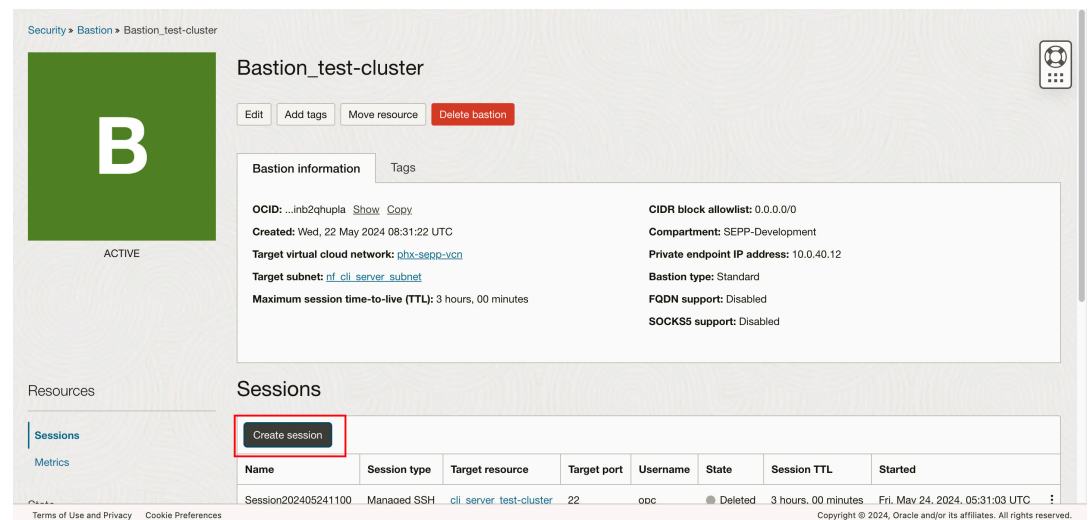
Figure 2-5   Identity and Security
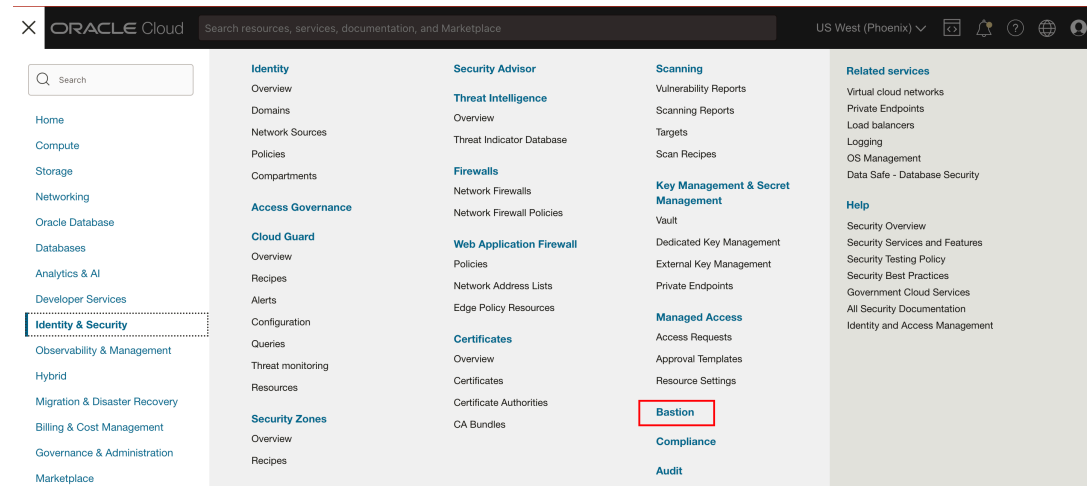
3. Select the **Bastion**.

Figure 2-6   Bastion
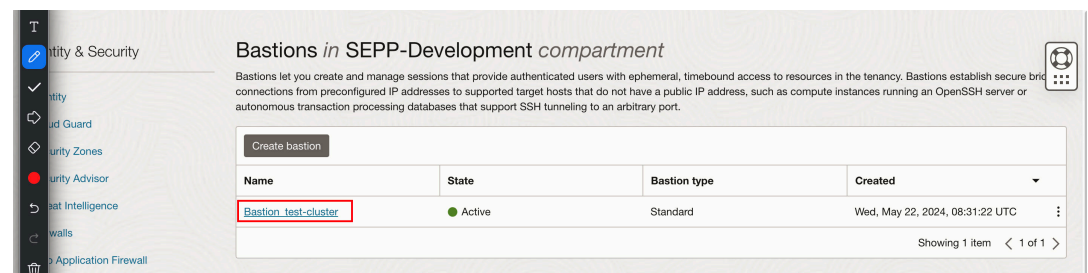
4. Click **Create session** to create the session.

Figure 2-7   Create Session

5. On the **Create session** page, enter the value for the following fields:

a. Session type as **Managed SSH session**.

b. Enter the **Session name** as per user's choice.

c. Type the **Username** as `opc`.

d. Select **Compute instance** as your CLI Server.

e. Enter the public key that will be used to login to the session.

f. Click the **Create session**.

Figure 2-8   Create Session

6. Click the three dots of the session created and copy the SSH Command using the option "**Copy SSH command**". The SSH Command copied above will take us directly to the CLI Server.

Figure 2-9   Copy SSH command

> ⓘ **Note**
>
> After Deploying OCI Adaptor using the steps mentioned below, User needs to perform the same steps of creating `Bastion Session` and copy SSH Command to jump to the CLI Server.

From the CLI Server, all the Kubectl commands can be run.

> ⓘ **Note**
>
> The maximum life span of a `Bastion Session` is three hours. After that (If Required) end user will have to create a new `Bastion Session`.

For more information, see the "Bastion Overview" section in *Oracle Cloud Infrastructure Documentation*.

CLI Server

The CLI Server is a private VM that connects to the OKE Cluster. It allows you to perform user operations and run all `kubectl` commands. The user can log into the CLI Server only through the Bastion Session. The CLI Server uses OCI's instance principle mechanism to authenticate itself with the OKE cluster.

For more information, see the "Calling Services from an Instance" section in *Oracle Cloud Infrastructure Documentation*.

The terraform scripts create a dynamic group, which allows the CLI Server to use the **Instance Principal Mechanism** to authenticate itself with the following matching rule and policy.

```
Matching Rule
All {instance.compartment.id='COMPARTMENT_ID', tag.cli_server_tag='true'}
Policy
Allow dynamic-group DYNAMIC_GROUP_NAME to manage cluster-family in
compartment COMPARTMENT_NAME
```

Node Pool

An OCI node pool is a group of nodes within a cluster that have the same configuration, such as a `Linux image, number of OCPUs, RAM per node, boot volume, the applicable Kubernetes version`, and so on. CNC uses this node pool to create compute for the NFs. The advantage of this node pool is that a new compute is created with the same attributes when it is resized. OKE worker nodes are created on this node pool.

For more information, see the "Node Pool Management" section in *Oracle Cloud Infrastructure Documentation*.

Service Gateway

The service gateway provides access from a VCN to other OCI services, such as Oracle Cloud Infrastructure (OCI) Object Storage. CNC uses Service Gateway to share metrics towards OCI Monitoring Services, share logs towards OCI Logging and Analytics and traces towards OCI

APM data collector. The traffic from the VCN to the Oracle service does not use the internet. It uses the Oracle network fabric.

For more information, see the "Overview of Service Gateways" section in *Oracle Cloud Infrastructure Documentation*.

3

OCI Adaptor Resource Requirements

This section lists the resource requirements to install and run OCI Adaptor.

Table 3-1   Consolidated Resource Profile for OCI Deployment

Resource Requirement	Service	
	Min	Max
CPU	500m	800m
Memory	1.5Gi	2.5Gi
Ephemeral Storage	400Mi	400Mi

The following table lists resource requirement for OCI Adaptor Services:

Table 3-2   Resource Profile for OCI Adaptor

Service Name	CPU		Memory (GB)		POD		Ephemeral Storage	
	Min	Max	Min	Max	Min	Max	Min	Max
oci-Adaptor-opentelemetry-collector-agent	100m	100m	500Mi	500Mi	1	*	100Mi	100Mi
oci-onm-mgmt-agent-0	200m	500m	500Mi	1Gi	1	1	100Mi	100Mi
<cnc_nf>-oci-Adaptor-logan-8g9fl	100m	100m	250Mi	500Mi	1	*	100Mi	100Mi
metric-server	100m	100m	250Mi	500Mi	1	1	100Mi	100Mi
Total	500m	800m	1.5Gi	2.5Gi	4	NA	400Mi	400Mi

> ⓘ **Note**
>
> * Both OTEL Collector and Fluentd are deployed as a daemonset. Thus, the maximum number of pods depends on the number of worker nodes in any cluster.

> ⓘ **Note**
>
> The listed OCI resources are essential for full deployment.

4

Uninstalling OCI Adaptor

This section provides information about uninstalling the OCI Adaptor.

Prerequisite:

The user must purge (delete) the logs manually.

To delete the logs, perform the below steps:

1. Log in to the **OCI Console**.
   For more information, see the "Signing In to the OCI Console" section in *Oracle Cloud Infrastructure Documentation*.

2. Open the navigation menu and select **Observability and Management** and click **Logging Analytics**.

3. Then in the list click on **Administration** and the click on **Storage**.

4. Click **Purge Logs** and then, click **Purge**.

Uninstalling the OCI Adaptor

To uninstall OCI Adaptor, perform the following procedure:

1. Log in to the **OCI Console**.
   For more information, see the "Signing In to the OCI Console" section in *Oracle Cloud Infrastructure Documentation*.

2. Open the navigation menu and select **Developer Services**. The Developer Services window appears in the right pane.

3. Under the Developer Services, select **Stacks** which is displayed under **Resource Manager**.

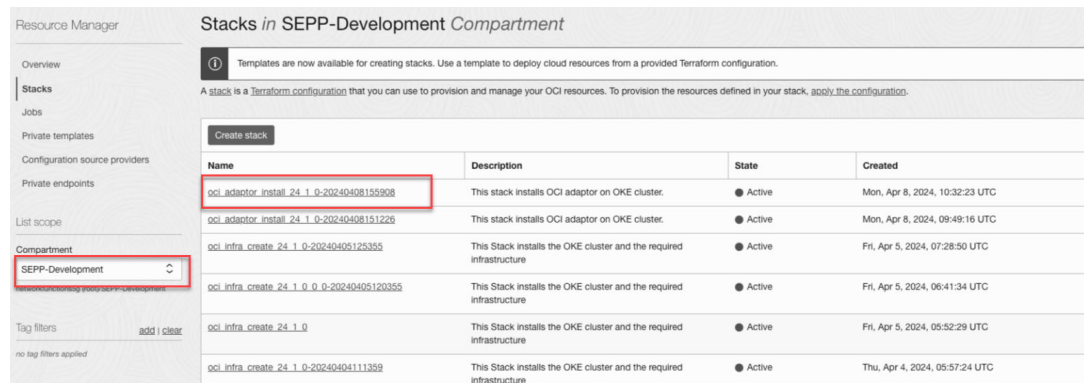4. The **Stacks** window appears.

Figure 4-1   Developer Services
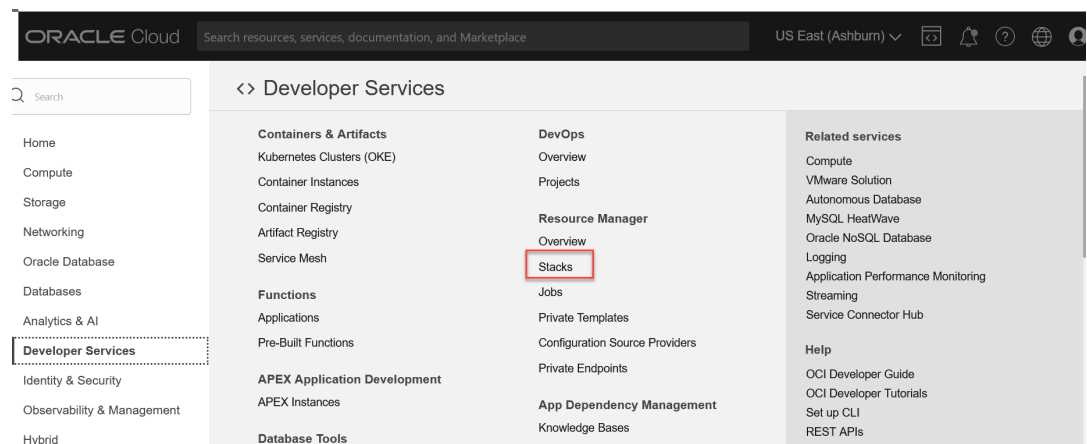
5. Select **Compartment** from the **Compartment drop-down** list.
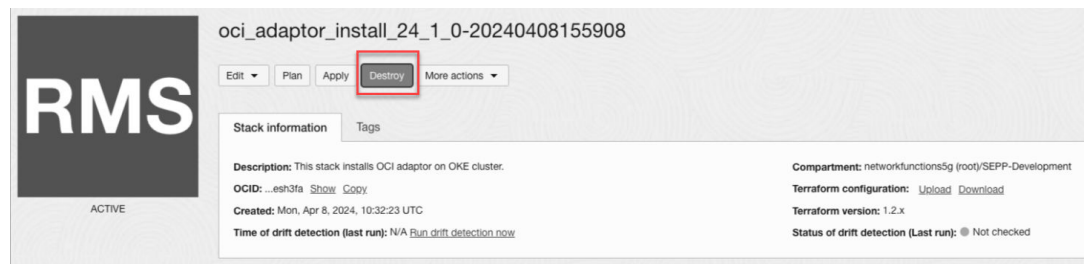
Figure 4-2   Compartment
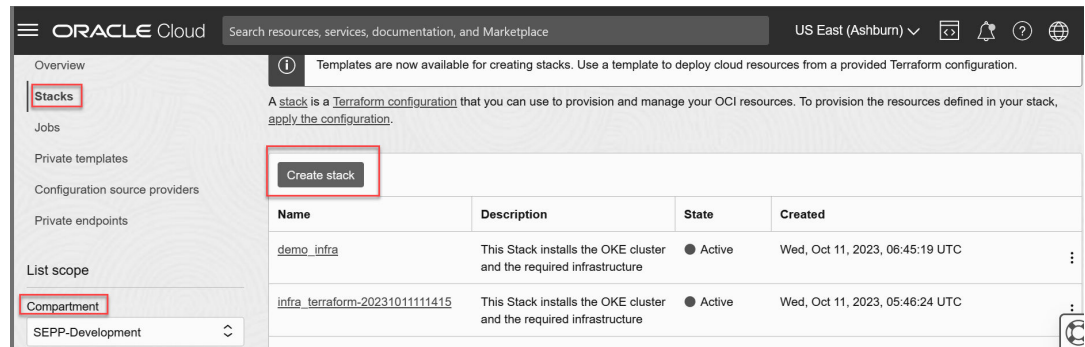
6. Select the **Stack** created during OCI deployment.

Figure 4-3   Select Stack

7. Click **Destroy**.

Figure 4-4   Destroy Stack

8. The confirmation page appears. Click **Destroy** to uninstall the OCI Adaptor.

> ⓘ **Note**
>
> The sample dashboards created automatically during OCI Adaptor installation are not deleted during OCI Adaptor uninstallation. It is recommended that you delete them manually.

5

Troubleshooting Scenarios

This section provides information on troubleshooting the common errors encountered during the deployment and creation of the OCI Adaptor infrastructure.

- Problem: The resource limit exceeded and the following error displayed:

  `"Error: 400-LimitExceeded, The following service limits were exceeded: <OCI Resource name>. Request a service limit increase from the service limits page in the console".`

  Solution: The user must request for the additional resources.

- Problem:

  – Scenario 1: While manually creating any OCI Resource or Component from the OCI Console.

  – Scenario 2: While running the apply step on the terraform stack.

  The user gets the following error:

  `"NotAuthorizedOrNotFound, Authorization failed or requested resource not found"`

  Solution: Verify the following:

  – The user is a `Tenancy Admin`.

  – To create the OCI infrastructure, it is recommended that first-level subcompartments be created in the root compartment. The creation of OCI infrastructure at the second-level subcompartment is not supported.