# Oracle® Communications

# Cloud Native Unified Data Repository User's Guide

Release 1.8

F34861-01

September 2020

Oracle Communications Cloud Native Unified Data Repository User's Guide, Release 1.8

F34861-01

# Contents

# My Oracle Support

My Oracle Support 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 MOS, 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.

# What's New in This Guide

This section shares the list of new features introduced in every OCUDR release. For more release specific information, you can refer to its release notes.

**Release 1.8**

The following new features are supported in this release:

- UDR deployment with service mesh like Aspen. This helps in controlling and monitoring the data flow within UDR microservices and outside as well.
- OAUTH2 token validation for ingress traffic.
- Bulk Import of subscriber data using import tool.
- Migration of 4G policy data from 4G UDR to 5G UDR using migration tool.
- Consolidated provisioning APIs for PCF data.
- Diameter Sh support including Quota and all other entities.
- Audit of two SLF segments using ProvGw.
- Supports UDR Alerts.
- Diameter configurations on CNC-Console GUI.

# 1 Overview

The 5G **Unified Data Repository (UDR)** is one of the main key component of the 5G Service Based Architecture. It is implemented as a cloud native function and offers a unified database for storing application, subscription, authentication, service authorization, policy data, session binding and Application state information. It provides a HTTP2 based RESTful interface for other NF's and provisioning clients to access the stored data.

Oracle's 5G UDR:

- Leverages a common Oracle Communications Cloud Native Framework
- Is compliant to 3GPP 29.505 Release 15 specification UDM
- Is compliant to 3GPP 29.519 Release 16 (backward compatible with Release 15) specification for PCF
- Has tiered architecture providing separation between the connectivity, business logic and data layers
- Uses Oracle MySQL NDB Cluster CGE Edition as backend database in the Data Tier
- Registers with NRF in the 5G network so that the other NFs in the network can discover UDR through NRF
- Registers UDR with services like DR-SERVICE and GROUP-ID-MAP

As per 3GPP, UDR supports following functionality:

- Storage and retrieval of subscription data by the UDM
- Storage and retrieval of policy data by the PCF
- Storage and retrieval of structured data for exposure
- Storage and retrieval of SLF information, consumed by NRF
- Application data (including Packet Flow Descriptions (PFDs) for application detection, AF request information for multiple UEs), by the NEF
- Subscription and Notification feature

**Unstructured Data Storage Function (UDSF)** is a part of Oracle's 5G UDR solution. It supports storage and retrieval of unstructured data by any 5G NF. The specifications of UDSF are presently not defined by 3GPP.

**5G SLF** functionality is also a part of Oracle's 5G UDR solution. It:

- Supports Nudr-groupid-map service as defined by 3GPP
- Registers with NRF for Nudr-groupid-map service
- Is complaint with 3GPP Release 16 for APIs to be consumed by 5G NRF
- Supports REST/JSON based provisioning APIs for SLF data

# Architecture

The Cloud Native Unified Data Repository architecture has following three tiers:

**Connectivity Tier**

- Ingress API Gateway (Spring Cloud Gateway [SCG] based) is used as an API gateway that receives all requests and forwards them to the Nudr-drservice service of Business Tier.
- It load balances the traffic and provides required authentication using Oauth2.
- It provides TLS support.
- It runs on Kubernetes/OCCNE as a microservice.
- It uses Egress API Gateway for Egress traffic arising from UDR (notifications and NRF management APIs).

**Business Tier**

- Provides the business logic of 5G Unified Data Repository.
- It has following micro services:
  - **nudr-drservice:** The core service that handles flexible URI support, runtime schema validation and connects to Data Tier for DB operations. It provides SLF lookup functionality.
  - **nudr-nrf-client-service:** Handles registration, heartbeat, update and deregistration with Network Repository Function (NRF).
  - **nudr-notify-service:** Handles notification messages to Policy Control Function (PCF) and Unified Data Management (UDM) for data subscriptions.
  - **nudr-config:** Handles all request from CNC-Console and redirects all requests to appropriate REST API of the config server. It allows users to configure UDR for all micro services.
  - **nudr-config-server:** Handles all the requests from nudr-config and updates the database.
  - **nudr-diameterproxy service:** Supports Diameter Sh interface for 4G policy data for the subscriber profile.

**Data Tier**

- Uses Oracle MySQL NDB Cluster, CGE edition as backend database in the DB tier. This provides HA and geo-redundcancy capabilities.
- Users can build database on either Bare metal, virtualized or on kubernetes platform (kubevirt based).

# Unified Data Repository Features

This section provides list of the features supported in Oracle Communications Unified Data Repository.

- **Flexible URI support:** A user can define a new URI for any resource at runtime for basic CRUD operations on the resource.

- **Supports multi-keys:** UDR supports multiple keys for a subscriber and provides flexibility to define new ones.
- **Runtime schema validation:** Allows users to modify and validate the schema in use for data storage without service restart.
- **Supports 5G PCF's AM**, **SM**, **UEPolicySet** and **UsageMonitoring data APIs**. It is also compliant with 3gpp spec v29.519, v16.2 (backward compatible with v15.3.0). See Rollback to v15.3.0 version of PCF Data
- **Provisioning support via REST/JSON:** UDR provides provisioning APIs for creating subscribers and provisioning different types of NF data (PCF, UDM, SLF).
- **Schema versioning:** Using this feature, UDR maintains different versions of schema, starting from default version v0 (software version shipped to customers) and as customer upgrades schema, it creates new versions like v0, v1, v2.
- **Subscription/Notification feature:** The "nudr-drservice" service receives and processes the subscription request to subscribe the subscriber's resources. Whenever it receives any update or delete request for subscribers (subscribed for notifications), the "nudr-drservice" service sends an internal signal (HTTP2 POST request) to "nudr-notify-service". The "nudr-notify-service" is a new service, which is responsible to send out the notification requests to the target.
- **UDSF API support:** UDR supports basic CRUD operations for UDSF API.
- **UDM API support:** UDR is compliant with 3GPP spec 29.505 v15.4.0. It supports encryption of stored data for certain types of UDM data. For this, it utilizes the DB Tier (MySQL NDB Cluster) encryption technology.
- **Integrated with CNE services:** Like Prometheus/Grafana for metrics, EFK/ Kibana for logging and Jaeger for tracing.
- **5G SLF functionality:** UDR provides Nudr-groupid-map service towards 5G NRF. It is compliant with 3GPP release 16, spec 29.504 v16.2.0. It provides provisioning APIs based on SLF Groupname and Nudr-group-id-map.
- Supports **TLS** with Ingress and Egress API gateway.
- **Integrated with ProvGwy** for receiving provisioning updates to SLF. See ProvGwy documentation
- **Integrated with CNC-C** for manual subscriber provisioning on the GUI and configuration of UDR services. The CNC-C GUI allows UDR users to configure SLF groupname.
- **Supports customized lables**, **annotations** and **naming conventions of kubernetes resources** in the Helm charts.
- **Supports Helm test** to validate the NF deployment.
- Exposes Diameter Sh interface for 4G Policy data, which PCRF uses. It also generates PNR notifications for profile updates.

### List of Operations Supported

Unified Data Repository supports the following operations.

#### Subscriber Profile Related Operations for Provisioning

- **Create Subscriber**: Provisions a subscriber with the given set of keys.
- **Update Subscriber**: Updates the subscriber information.
- **Get subscriber**: Retrieves the subscriber information completely.

- **Delete Subscriber**: Deletes the subscriber information and related data.

**NF Data Related Operations as per 3GPP**

- **Insert Data**: Creates a subscriber if not present and inserts the specific data (policy_data/udsf_data/udm_data/SLF Data).
- **Update Data**: Creates a subscriber if not present and updates the complete data for a particular subscriber as given in request.
- **Patch Data**: Supports patch update of a specific data or parts of it. If not present, it adds the data. This operation is valid only when subscriber is already present.
- **Get Data**: Retrieves the requested data for a particular subscriber.
- **Delete Data**: Deletes the requested data for a particular subscriber.
- **Subscription Request**: Subscribes to a subscriber resources and get notified in case of updates.
- **Notification**: UDR generates notifications when there are updates to subscriber resources subscribed in previous request. These notifications are sent to notification URI received in subscription request.

# 2 Configuring User Parameters

The UDR micro services have configuration options. The user should be able to configure them via deployment values.yaml.

> **Note:**
>
> The default value of some of the settings may change.

> **Note:**
>
> - **NAME**: is the release name used in helm install command
> - **NAMESPACE**: is the namespace used in helm install command
> - **K8S_DOMAIN**: is the default kubernetes domain (svc.cluster.local)

**Default Helm Release Name**:- ocudr

**Global Configuration:** These values are suffixed to all the container names of OCUDR. These values are useful to add custom annotation(s) to all non-Load Balancer Type Services that OCUDR helm chart creates.

Following table provides the parameters for **global configurations**.

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| dockerRegistry | Docker registry from where the images will be pulled | ocudr-registry.us.oracle.com:5000 | Not applicable | |
| mysql.dbServiceName | DB service to connect | mysql-connectivity-service.occne-infra | Not applicable | This is a CNE service used for db connection. Default name used on CNE is the same as configured. |
| mysql.port | Port for DB Service Connection | 3306 | Not applicable | |
| udrTracing.enable | Flag to enable udr tracing on Jaeger | false | true/false | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| udrTracing.host | Jaegar Service Name installed in CNE | occne-tracer-jaeger-collector.occne-infra | Not applicable | |
| udrTracing.port | Jaegar Service Port installed in CNE | 14268 | Not applicable | |
| dbenc.shavalue | Encryption Key size | 256 | 256 or 512 | |
| serviceAccountName | Service account name | null | Not Applicable | The serviceaccount, role and rolebindings required for deployment should be done prior installation. Use the created serviceaccountname here. |
| egress.enabled | Flag to enable outgoing traffic through egress gateway | true | true/false | |
| configServerEnable | Flag to enable config-server | true | true/false | |
| initContainerEnable | Flag to disable init container for config-server. This is not required because the pre install hooks take care of DB tables creation and connectivity is also verified | false | true/false | |
| dbCredSecretName | DB Credentioal Secret Name | ocudr-secrets | Not Applicable | |
| configServerFullNameOverride | Config Server Full Name Override | nudr-config-server | Not Applicable | |
| udrServices | Services supported on the UDR deployment, This config decides the schema execution on the udrdb which is done by the nudr-preinstall hook pod. | All | All/nudr-dr/nudr-group-id-map | For SLF, set udrServices values as nudr-group-id-map. |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| udsfEnable | Flag to enable UDSF services on the deployment | false | true/false | |
| publicHttpSignalingPort | Port on which ingressgateway listens for incoming http requests. | 80 | Valid Port | |
| publicHttpsSignallingPort | Port on which ingressgateway listens for incoming https requests. | 443 | Valid Port | |
| nfInstanceId | Nf Instance ID for UDR (same is registered with NRF) | 5a7bd676-ceeb-44bb-95e0-f6a55a328b03 | Valid uuid | A valid UUID is a 128-bit unique number that helps to identify information in computer systems. |
| test.nfName | NF name on which the helm test is performed. For UDR the default value is UDR. Will be used in container name as suffix | ocudr | Not applicable | |
| test.image.name | Image name for the helm test container image | ocudr/nf_test | Not Applicable | |
| test.image.tag | Image version tag for helm test | 1.8.0 | Not Applicable | |
| test.config.logLevel | Log level for helm test pod | WARN | Possible Values -<br>WARN<br>INFO<br>DEBUG | |
| test.config.timeout | Timeout value for the helm test operation. If exceeded helm test will be considered as failure | 120 | Range: 1-300<br>Unit:seconds | |
| preinstall.image.name | Image name for the nudr-prehook pod which will take care of DB and table creation for UDR deployment. | ocudr/prehook | Not Applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| preinstall.image.tag | Image version for nudr-prehook pod image | 1.8.0 | Not Applicable | |
| preinstall.config.logLevel | Log level for preinstall hook pod | WARN | Possible Values -<br>WARN<br>INFO<br>DEBUG | |
| hookJobResources.limits.cpu | CPU limit for pods created kubernetes hooks/jobs created as part of UDR installation. Applicable for helm test job as well. | 2 | Not Applicable | |
| hookJobResources.limits.memory | Memory limit for pods created kubernetes hooks/jobs created as part of UDR installation. Applicable for helm test job as well. | 2Gi | Not Applicable | |
| hookJobResources.requests.cpu | CPU requests for pods created kubernetes hooks/jobs created as part of UDR installation. Applicable for helm test job as well. | 1 | Not Applicable | The cpu to be allocated for hooks during deployment |
| hookJobResources.requests.memory | Memory requests for pods created k8s hooks/jobs created as part of UDR installation. Applicable for helm test job as well. | 1Gi | Not Applicable | The memory to be allocated for hooks during deployment |
| customExtension.allResources.labels | Custom Labels that needs to be added to all the OCUDR kubernetes resources | null | Not Applicable | This can be used to add custom label(s) to all k8s resources that will be created by OCUDR helm chart. |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| customExtension.allResources.annotations | Custom Annotations that needs to be added to all the OCUDR kubernetes resources | null | Not Applicable<br>**Note:** ASM related annotations needs to be added under ASM Specific Configuration section | This can be used to add custom annotation(s) to all k8s resources that will be created by OCUDR helm chart. |
| customExtension.lbServices.labels | Custom Labels that needs to be added to OCUDR Services that are considered as Load Balancer type | null | Not Applicable | This can be used to add custom label(s) to all Load Balancer Type Services that will be created by OCUDR helm chart. |
| customExtension.lbServices.annotations | Custom Annotations that needs to be added to OCUDR Services that are considered as Load Balancer type | null | Not Applicable | This can be used to add custom annotation(s) to all Load Balancer Type Services that will be created by OCUDR helm chart. |
| customExtension.lbDeployments.labels | Custom Labels that needs to be added to OCUDR Deployments that are associated to a Service which is of Load Balancer type | null | Not Applicable | This can be used to add custom label(s) to all Deployments that will be created by OCUDR helm chart which are associated to a Service which if of Load Balancer Type. |
| customExtension.lbDeployments.annotations | Custom Annotations that needs to be added to OCUDR Deployments that are associated to a Service which is of Load Balancer type | null | Not Applicable<br>**Note:** ASM related annotations needs to be added under ASM Specific Configuration section | This can be used to add custom annotation(s) to all Deployments that will be created by OCUDR helm chart which are associated to a Service which if of Load Balancer Type. |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| customExtension.nonlbServices.labels | Custom Labels that needs to be added to OCUDR Services that are considered as not Load Balancer type | null | Not Applicable | This can be used to add custom label(s) to all non-Load Balancer Type Services that will be created by OCUDR helm chart. |
| customExtension.nonlbServices.annotations | Custom Annotations that needs to be added to OCUDR Services that are considered as not Load Balancer type | null | Not Applicable | This can be used to add custom annotation(s) to all non-Load Balancer Type Services that will be created by OCUDR helm chart. |
| customExtension.nonlbDeployments.labels | Custom Labels that needs to be added to OCUDR Deployments that are associated to a Service which is not of Load Balancer type | null | Not Applicable | This can be used to add custom label(s) to all Deployments that will be created by OCUDR helm chart which are associated to a Service which if not of Load Balancer Type. |
| customExtension.nonlbDeployments.annotations | Custom Annotations that needs to be added to OCUDR Deployments that are associated to a Service which is not of Load Balancer type | null | Not Applicable<br>**Note:** ASM related annotations to be added under ASM Specific Configuration section | This can be used to add custom annotation(s) to all Deployments that will be created by OCUDR helm chart which are associated to a Service which if not of Load Balancer Type. |
| k8sResource.container.prefix | Value that will be prefixed to all the container names of OCUDR. | null | Not Applicable | This value will be used to prefix to all the container names of OCUDR. |
| k8sResource.container.suffix | Value that will be suffixed to all the container names of OCUDR. | null | Not Applicable | This value will be used to prefix to all the container names of OCUDR. |

Following table provides the parameters for **nudr-drservice micro service**.

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| image.name | Docker Image name | ocudr/<br>nudr_datarepository_service | Not applicable | |
| image.tag | Tag of Image | 1.8.0 | Not applicable | |
| image.pullPolicy | This setting signifies whether image needs to be pulled or not | Always | Possible Values -<br>Always<br>IfNotPresent<br>Never | |
| subscriber.autocreate | Flag to enable auto creation of subscriber | true | true/false | This flag enables auto creation of subscriber when creating data for a non existent subscriber. |
| validate.smdata | Flag to enable correlation feature for smdata | false | true/false | This flag controls the correlation feature for smdata. This flag must be false if using v16.2.0 for PCF data. |
| logging.level.root | Log Level | WARN | Possible Values -<br>WARN<br>INFO<br>DEBUG | Log level of the nudr-drservice pod |
| deployment.replicaCount | Replicas of nudr-drservice pod | 2 | Not applicable | Number of nudr-drservice pods to be maintained by replica set created with deployment |
| minReplicas | Minimum Replicas | 2 | Not applicable | Minimum number of pods |
| maxReplicas | Maximum Replicas | 8 | Not applicable | Maximum number of pods |
| service.http2enabled | Enabled HTTP2 support flag for rest server | true | true/false | Enable/Disable HTTP2 support for rest server |
| service.type | UDR service type | ClusterIP | Possbile Values-<br>ClusterIP<br>NodePort<br>LoadBalancer | The kubernetes service type for exposing UDR deployment<br>Note: Suggested to be set as ClusterIP (default value) always |
| service.port.http | HTTP port | 5001 | Not applicable | The http port to be used in nudr-drservice service |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| service.port.https | HTTPS port | 5002 | Not applicable | The https port to be used for nudr-drservice service |
| service.port.management | Management port | 9000 | Not applicable | The actuator management port to be used for nudr-drservice service |
| resources.requests.cpu | Cpu Allotment for nudr-drservice pod | 3 | Not applicable | The cpu to be allocated for nudr-drservice pod during deployment |
| resources.requests.memory | Memory allotment for nudr-drservice pod | 4Gi | Not applicable | The memory to be allocated for nudr-drservice pod during deployment |
| resources.limits.cpu | Cpu allotment limitation | 3 | Not applicable | |
| resources.limits.memory | Memory allotment limitation | 4Gi | Not applicable | |
| resources.target.averageCpuUtil | CPU utilization limit for autoscaling | 80 | Not Applicable | CPU utilization limit for creating HPA |
| notify.port.http | HTTP port on which notify service is running | 5001 | Not applicable | |
| notify.port.https | HTTPS port on which notify service is running | 5002 | Not applicable | |
| hikari.poolsize | Mysql Connection pool size | 25 | Not applicable | The hikari pool connection size to be created at start up |
| vsaLevel | The data level where the vsa which holds the 4G Policy data is added. | smpolicy | Not applicable | |
| vsaBillingDay | The Billing day value | 0 | Not applicable | |
| tracingEnabled | Flag to enable/disable jaeger tracing for nudr-drservice | false | true/false | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| service.customExtension.labels | Custom Labels that needs to be added to nudr-drservice specific Service. | null | Not Applicable | This can be used to add custom label(s) to nudr-drservice Service. |
| service.customExtension.annotations | Custom Annotations that needs to be added to nudr-drservice specific Services. | null | Not Applicable | This can be used to add custom annotation(s) to nudr-drservice Service. |
| deployment.customExtension.labels | Custom Labels that needs to be added to nudr-drservice specific deployment. | null | Not Applicable | This can be used to add custom label(s) to nudr-drservice Deployment. |
| deployment.customExtension.annotations | Custom Annotations that needs to be added to nudr-drservice specific deployment. | null | Not Applicable | This can be used to add custom annotation(s) to nudr-drservice deployment. |
| readinessProbe.initialDelaySeconds | Configurable wait time before performing the first readiness probe by the kubelet<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 70 | Not Applicable<br>Unit: Seconds | |
| readinessProbe.periodSeconds | Time interval for every readiness probe check.<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 10 | Not Applicable<br>Unit: Seconds | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| livenessProbe.initialDelaySeconds | Configurable wait time before performing the first liveness probe by the kubelet.<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 70 | Not Applicable<br>Unit: Seconds | |
| livenessProbe.periodSeconds | Time interval for every liveness probe check.<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 10 | Not Applicable<br>Unit: Seconds | |

Following table provides the parameters for **nudr-notify-service micro service**.

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| enabled | flag for enabling or disabling nudr-notify-service | true | true or false | For SLF deployment, this micro service must be disabled. |
| image.name | Docker Image name | ocudr/nudr_notify_service | Not applicable | |
| image.tag | Tag of Image | 1.8.0 | Not applicable | |
| image.pullPolicy | This setting will tell if image need to be pulled or not | Always | Possible Values -<br>Always<br>IfNotPresent<br>Never | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| notification.retrycount | Number of notifications to be attempted | 3 | Range: 1 - 10 | Number of notification attempts to be done in case of notification failures.<br>Whether retry should be done will be based on notification.retryerrorcodes configuration. |
| notification.retryinterval |  | 5 | Range: 1 - 60<br>Unit: Seconds | The retry interval for notifications in case of failure. Unit is in seconds.<br>Whether retry should be done will be based on notification.retryerrorcodes configuration. |
| notification.retryerrorcodes | Notification failures eligible for retry | "400,429,500,503" | Valid HTTP status codes comma seperated | Comma separated error code should be given. These error codes will be eligible for retry notifications in case of failures. |
| hikari.poolsize | Mysql Connection pool size | 10 | Not applicable | The hikari pool connection size to be created at start up |
| tracingEnabled | Flag to enable/disable jaeger tracing for nudr-notify-service | false | true/false |  |
| http.proxy.port | Port to connect to egress gateway | 8080 | Not applicable |  |
| logging.level.root | Log Level | WARN | Possible Values -<br>WARN<br>INFO<br>DEBUG | Log level of the notify service pod |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| deployment.replicaCount | Replicas of nudr-notify-service pod | 2 | Not applicable | Number of nudr-notify-service pods to be maintained by replica set created with deployment |
| minReplicas | Minimum Replicas | 2 | Not applicable | Minimum number of pods |
| maxReplicas | Maximum Replicas | 4 | Not applicable | Maximum number of pods |
| service.http2enabled | Enabled HTTP2 support flag | true | true/false | This is a read only parameter. Do not change this value |
| service.type | UDR service type | ClusterIP | Possbile Values-<br>ClusterIP<br>NodePort<br>LoadBalancer | The kubernetes service type for exposing UDR deployment<br>Note: Suggested to be set as ClusterIP (default value) always |
| service.port.http | HTTP port | 5001 | Not applicable | The http port to be used in notify service to receive signals from nudr-notify-service pod. |
| service.port.https | HTTPS port | 5002 | Not applicable | The https port to be used in notify service to receive signals from nudr-notify-service pod. |
| service.port.management | Management port | 9000 | Not applicable | The actuator management port to be used for notify service. |
| resources.requests.cpu | Cpu Allotment for nudr-notify-service pod | 3 | Not applicable | The cpu to be allocated for notify service pod during deployment |
| resources.requests.memory | Memory allotment for nudr-notify-service pod | 3Gi | Not applicable | The memory to be allocated for nudr-notify-service pod during deployment |
| resources.limits.cpu | Cpu allotment limitation | 3 | Not applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| resources.limits.memory | Memory allotment limitation | 3Gi | Not applicable | |
| resources.target.averageCpuUtil | CPU utilization limit for autoscaling | 80 | Not Applicable | CPU utilization limit for creating HPA |
| service.customExtension.labels | Custom Labels that needs to be added to nudr-notify-service specific service. | null | Not Applicable | This can be used to add custom label(s) tonudr-notify-service Service. |
| service.customExtension.annotations | Custom Annotations that needs to be added to nudr-notify-service specific services. | null | Not Applicable | This can be used to add custom annotation(s) to nudr-notify-service Service. |
| deployment.customExtension.labels | Custom Labels that needs to be added to nudr-notify-service specific deployment. | null | Not Applicable | This can be used to add custom label(s) to nudr-notify-service deployment. |
| deployment.customExtension.annotations | Custom Annotations that needs to be added to nudr-notify-service specific deployment. | null | Not Applicable | This can be used to add custom annotation(s) to nudr-notify-service deployment. |
| readinessProbe.initialDelaySeconds | Configurable wait time before performing the first readiness probe by the kubelet<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 80 | Not Applicable<br>Unit: Seconds | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| readinessProbe.periodSeconds | Time interval for every readiness probe check.<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 5 | Not Applicable<br>Unit: Seconds | |
| livenessProbe.initialDelaySeconds | Configurable wait time before performing the first liveness probe by the kubelet.<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 80 | Not Applicable<br>Unit: Seconds | |
| livenessProbe.periodSeconds | Time interval for every liveness probe check.<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 20 | Not Applicable<br>Unit: Seconds | |

Following table provides the parameters for **nudr-nrf-client-service micro service**.

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| enabled | flag for enabling or disabling nudr-nrf-client-service | true | true/false | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| host.baseurl | NRF url for registration | http://ocnrf-ingressgateway.mynrf.svc.cluster.local/nnrf-nfm/v1/nf-instances | Not applicable | Url used for udr to connect and register with NRF |
| host.proxy | Proxy Setting | NULL | nrfClient.host | Proxy setting if required to connect to NRF |
| ssl | SSL flag | false | true/false | SSL flag to enable SSL with udr nrf client pod |
| logging.level.root | Log Level | WARN | Possible Values -<br>WARN<br>INFO<br>DEBUG | Log level of the UDR nrf client pod |
| image.name | Docker Image name | ocudr/nudr_nrf_client_service | Not applicable | |
| image.tag | Tag of Image | 1.8.0 | Not applicable | |
| image.pullPolicy | This setting will tell if image need to be pulled or not | Always | Possible Values -<br>Always<br>IfNotPresent<br>Never | |
| heartBeatTimer | Heart beat timer | 90 | Unit: Seconds | |
| udrGroupId | Group ID of UDR | udr-1 | Not applicable | |
| capacityMultiplier | Capacity of UDR | 500 | Not applicable | Capacity multiplier of UDR based on number of UDR pods running |
| supirange | Supi Range supported with UDR | [{\"start\": \"10000000000\", \"end\": \"20000000000\"}] | Valid start and end supi range | |
| priority | Priority | 10 | Priority to be sent in registration request | Priority to be sent in registration request |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| fqdn | UDR FQDN | ocudr-ingressgateway.myudr.svc.cluster.local | Not Applicable | FQDN to used for registering in NRF for other NFs to connect to UDR.<br>Note: Be cautious in updating this value. Should consider helm release name, namespace used for udr deployment and name resolution setting in k8s. |
| gpsirange | Gpsi Range supported with UDR | [{\"start\": \"10000000000\", \"end\": \"20000000000\"}] | Valid start and end gpsi range | |
| livenessProbeMaxRetry | Max retries of liveness proble failed | 5 | This should be changed based on how many times do you want to retry | This should be changed based on how many times do you want to retry if liveness fails |
| udrMasterIpv4 | Master IP of which we deployed | 10.0.0.0 | This should be changed with the master ip which we deployed | udrMasterIpv4 is used to send the ipv4 address to the nrf while registration. |
| plmnvalues | Plmn values range that it supports | [{\"mnc\": \"14\", \"mcc\": \"310\"}] | This values can be changed that the range it supports | Plmn values are sent to nrf during regisration from UDR. |
| scheme | scheme in which udr supports | http | This can be changed to https. | scheme which we send to NRF during registration |
| resources.requests.cpu | Cpu Allotment for nudr-notify-service pod | 1 | Not applicable | The cpu to be allocated for nrf client service pod during deployment |
| resources.requests.memory | Memory allotment for nudr-notify-service pod | 2Gi | Not applicable | The memory to be allocated for nrf client service pod during deployment |
| resources.limits.cpu | Cpu allotment limitation | 1 | Not applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| resources.limits.memory | Memory allotment limitation | 2Gi | Not applicable | |
| http.proxy.port | Port to connect egress gateway | 8080 | Not applicable | |
| service.customExtension.labels | Custom Labels that needs to be added to nudr-nrf-client specific service. | null | Not Applicable | This can be used to add custom label(s) to nudr-nrf-client service. |
| service.customExtension.annotations | Custom Annotations that needs to be added to nudr-nrf-client specific services. | null | Not Applicable | This can be used to add custom annotation(s) to nudr-nrf-client service. |
| deployment.customExtension.labels | Custom Labels that needs to be added to nudr-nrf-client specific deployment. | null | Not Applicable | This can be used to add custom label(s) to nudr-nrf-client deployment. |
| deployment.customExtension.annotations | Custom Annotations that needs to be added to nudr-nrf-client specific deployment. | null | Not Applicable<br>**Note:** ASM related annotations to be added under ASM Specific Configuration section | This can be used to add custom annotation(s) to nudr-nrf-client deployment. |

Following table provides the parameters for **nudr-config micro service**.

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| enabled | flag for enabling or disabling nudr-config service | true | true/false | |
| logging.level.root | Log Level | WARN | Possible Values -<br>WARN<br>INFO<br>DEBUG | Log level of the nudr-config pod |
| service.http2enabled | Enabled HTTP2 support flag for rest server | true | true/false | Enable/Disable HTTP2 support for rest server |
| image.name | Docker Image name | ocudr/nudr_config | Not applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| service.customExtension.labels | Custom Labels that needs to be added to nudr-config specific Service. | null | Not applicable | This can be used to add custom label(s) to nudr-config Service. |
| service.customExtension.annotations | Custom Annotations that needs to be added to nudr-config specific Services. | null | Not applicable | This can be used to add custom annotation(s) to nudr-config Service. |
| deployment.customExtension.labels | Custom Labels that needs to be added to nudr-config specific Deployment. | null | Not applicable | This can be used to add custom label(s) to nudr-config Deployment. |
| deployment.customExtension.annotations | Custom Annotations that needs to be added to nudr-config specific Deployment. | null | Not applicable | This can be used to add custom annotation(s) to nudr-config Deployment. |
| service.type | UDR service type | ClusterIP | Possbile Values-<br>ClusterIP<br>NodePort<br>LoadBalancer | The kubernetes service type for exposing UDR deployment<br>Note: Suggested to be set as ClusterIP (default value) always |
| image.pullPolicy | This setting will tell if image need to be pulled or not | Always | Possible Values -<br>Always<br>IfNotPresent<br>Never | |
| service.port.management | Management port | 9000 | Not applicable | The actuator management port to be used for nudr-config service |
| service.port.https | HTTPS port | 5002 | Not applicable | The https port to be used for nudr-config service |
| service.port.http | HTTP port | 5001 | Not applicable | The http port to be used in nudr-config service |
| resources.target.averageCpuUtil | CPU utilization limit for autoscaling | 80 | Not Applicable | CPU utilization limit for creating HPA |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| resources.requests.memory | Memory allotment for nudr-drservice pod | 2Gi | Not applicable | The memory to be allocated for nudr-config pod during deployment |
| resources.limits.memory | Memory allotment limitation | 2Gi | Not applicable | |
| resources.requests.cpu | Cpu Allotment for nudr-drservice pod | 2 | Not applicable | The cpu to be allocated for nudr-config pod during deployment |
| resources.limits.cpu | Cpu allotment limitation | 2 | Not applicable | |
| image.tag | Tag of Image | 1.8.0 | Not applicable | |
| deployment.replicaCount | Replicas of nudr-config pod | 1 | Not applicable | Number of nudr-config pods to be maintained by replica set created with deployment |
| minReplicas | Minimum Replicas | 1 | Not applicable | Minimum number of pods |
| maxReplicas | Maximum Replicas | 1 | Not applicable | Maximum number of pods |
| readinessProbe.initialDelaySeconds | Configurable wait time before performing the first readiness probe by the kubelet<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 30 | Not Applicable<br>Unit: Seconds | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| readinessProbe.periodSeconds | Time interval for every readiness probe check.<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 5 | Not Applicable<br>Unit: Seconds | |
| livenessProbe.initialDelaySeconds | Configurable wait time before performing the first liveness probe by the kubelet.<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 40 | Not Applicable<br>Unit: Seconds | |
| livenessProbe.periodSeconds | Time interval for every liveness probe check.<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 10 | Not Applicable<br>Unit: Seconds | |

Following table provides the parameters for **nudr-config-server Micro service**.

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| enabled | Flag to enable/ disable nudr-config-server service | true | true/false | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| global.nfName | It is NF name used to add with config server service name. | nudr | Not applicable | |
| global.imageServiceDetector | Image Service Detector for config-server init container | ocudr/readiness-detector:1.7.1 | Not Applicable | |
| global.envJaegerAgentHost | Host FQDN for Jaeger agent service for config-server tracing | ' ' | Not Applicable | |
| global.envJaegerAgentPort | Port for Connection to Jaeger agent for config-server tracing | 6831 | Valid Port | |
| envLoggingLevelApp | Log Level | WARN | Possible Values -<br>WARN<br>INFO<br>DEBUG | Log level of the nudr-config-server pod |
| replicas | Replicas of nudr-config-server pod | 1 | Not applicable | Number of nudr-config-server pods to be maintained by replica set created with deployment |
| service.type | UDR service type | ClusterIP | Possbile Values-<br>ClusterIP<br>NodePort<br>LoadBalancer | The kubernetes service type for exposing UDR deployment<br>Note: Suggested to be set as ClusterIP (default value) always |
| resources.requests.cpu | Cpu Allotment for nudr-drservice pod | 2 | Not applicable | The cpu to be allocated for nudr-config-server pod during deployment |
| resources.requests.memory | Memory allotment for nudr-drservice pod | 512Mi | Not applicable | The memory to be allocated for nudr-config-server pod during deployment |
| resources.limits.cpu | Cpu allotment limitation | 2 | Not applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| resources.limits.memory | Memory allotment limitation | 2Gi | Not applicable | |
| readinessProbe.initialDelaySeconds | Configurable wait time before performing the first readiness probe by the kubelet<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 70 | Not Applicable<br>Unit: Seconds | |
| readinessProbe.periodSeconds | Time interval for every readiness probe check.<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 10 | Not Applicable<br>Unit: Seconds | |
| readinessProbe.timeoutSeconds | Number of seconds after which the probe times out<br>**Note:** Do not change this default value. | 3 | Not Applicable | |
| readinessProbe.successThreshold | Minimum consecutive successes for the probe to be considered successful after having failed<br>**Note:** Do not change this default value. | 1 | Not Applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| readinessProbe.failureThreshold | When a Pod starts and the probe fails, Kubernetes tries failureThreshold times before giving up<br>**Note:** Do not change this default value. | 3 | Not Applicable | |
| livenessProbe.initialDelaySeconds | Configurable wait time before performing the first liveness probe by the kubelet.<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 60 | Not Applicable<br>Unit: Seconds | |
| livenessProbe.periodSeconds | Time interval for every liveness probe check.<br>**Note:** Do not change this value. If there is any delay in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 15 | Not Applicable<br>Unit: Seconds | |
| livenessProbe.timeoutSeconds | Number of seconds after which the probe times out<br>Note: Do not change this default value. | 3 | Not Applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| livenessProbe.successThreshold | Minimum consecutive successes for the probe to be considered successful after having failed<br>Note: Do not change this default value. | 1 | Not Applicable | |
| livenessProbe.failureThreshold | When a Pod starts and the probe fails, Kubernetes will try failureThreshold times before giving up<br>Note: Do not change this default value. | 3 | Not Applicable | |

Following table provides parameters for **nudr-diameterproxy micro service**.

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| enabled | To enable service. | true | Not applicable | Used to enable or disable service. |
| image.name | Docker Image name | ocudr/ nudr_diameterproxy | Not applicable | |
| image.tag | Tag of Image | 1.8.0 | Not applicable | |
| image.pullPolicy | This setting will tell if image need to be pulled or not | Always | Possible Values -<br>Always<br>IfNotPresent<br>Never | |
| logging.level.root | Log Level | WARN | Possible Values -<br>WARN<br>INFO<br>DEBUG | The log level of the nudr-diameterproxy server pod |
| deployment.replicaCount | Replicas of the nudr-diameterproxy pod | 2 | Not applicable | Number of nudr-config-server pods to be maintained by replica set created with deployment |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| minReplicas | min replicas of nudr-diameterproxy | 2 | Not applicable | Minimum number of pods |
| maxReplicas | max replicas of nudr-diameterproxy | 4 | Not applicable | Maximum number of pods |
| service.http2enabled | Enabled HTTP2 support flag for rest server | true | true/false | Enable/Disable HTTP2 support for rest server |
| service.type | UDR service type | ClusterIP | Possible Values-<br>ClusterIP<br>NodePort<br>LoadBalancer | The Kubernetes service type for exposing UDR deployment<br>Note: Suggested to be set as ClusterIP (default value) always |
| service.diameter.type | Diameter service type | LoadBalancer | Possible Values-<br>ClusterIP<br>NodePort<br>LoadBalancer | The Kubernetes service type for exposing UDR deploymentdiameter traffic goes via diameter-endpoint, not via ingress-gateway |
| service.port.http | HTTP port | 5001 | Not applicable | The HTTP port to be used in nudr-diameterproxy service |
| service.port.https | HTTPS port | 5002 | Not applicable | The https port to be used for nudr-diameterproxy service |
| service.port.management | Management port | 9000 | Not applicable | The actuator management port to be used for nudr-diameterproxy service |
| service.port.diameter | Diameter port | 6000 | Not applicable | The diameter port to be used for nudr-diameterproxy service |
| resources.requests.cpu | Cpu Allotment for nudr-diameterproxy pod | 3 | Not applicable | The CPU to be allocated for nudr-diameterproxy pod during deployment |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| resources.requests.memory | Memory allotment for nudr-diameterproxy pod | `4Gi` | Not applicable | The memory to be allocated for nudr-diameterproxy pod during deployment |
| resources.limits.cpu | Cpu allotment limitation | 3 | Not applicable | The CPU to be max allocated for nudr-diameterproxy pod |
| resources.limits.memory | Memory allotment limitation | 4Gi | Not applicable | The memory to be max allocated for nudr-diameterproxy pod |
| resources.target.averageCpuUtil | CPU utilization limit for autoscaling | 80 | Not Applicable | CPU utilization limit for creating HPA |
| drservice.port.http | HTTP port on which dr service is running | 5001 | Not Applicable | dr-service port is required in diameterproxy application |
| drservice.port.https | HTTPS port on which dr service is running | 5002 | Not Applicable | dr-service port is required in diameterproxy application |
| diameter.realm | Realm of the diameterproxy microservice | oracle.com | String value | Host realm of diameterproxy |
| diameter.identity | FQDN of the diameterproxy in diameter messages | nudr.oracle.com | String value | identity of the diameterproxy |
| diameter.strictParsing | Strict parsing of Diameter AVP and Messages | false | Not Applicable | strict parsing |
| diameter.IO.threadCount | Number of thread for IO operation | 0 | 0 to 2* CPU | Number of threads to handle IO operations in diameterproxy pod<br>if threadcount is 0 then application choose the threadCount based on pod profile size |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| diameter.IO.queueSize | Queue size for IO | 0 | 2048 to 8192 | the count should be the power of 2<br>if queueSize is 0 then application choose the queueSize based on pod profile size |
| diameter.messageBuffer.threadCount | Number of threads for process the message | 0 | 0 to 2* CPU | Number of threads to handle meassages in diameterproxy pod<br>if threadcount is 0 then application choose the threadCount based on pod profile size |
| diameter.peer.setting | Diameter peer setting | reconnectDelay: 3<br>responseTimeout: 4<br>connectionTimeOut: 3<br>watchdogInterval: 6<br>transport: 'TCP'<br>reconnectLimit: 50 | Not Applicable | 1. reconnect delay for diameter reonnect (in seconds).<br>2. total turnaround time for process the diameter messages. (in sec)<br>3. TCP connection timeout time. (in sec)<br>4. DWR and DWA messages every number of time (in sec)<br>5. Transport layer<br>6. reconnect the number of time if diameter peer is down |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| diameter.peer.nodes | diameter server peer nodes list | - name: 'seagull'<br>responseOnly: false<br>namespace: 'seagull1'<br>host: '10.75.185.158'<br>domain: 'svc.cluster.local'<br>port: 4096<br>realm: 'seagull1.com'<br>identity: 'seagull1a.seagull1.com' | Not applicable | the diameter server peer node information<br>*it should be yaml list<br>*default values are template , how to add peer nodes. |
| diameter.peer.clientNodes | diameter client peers | - identity: 'seagull1a.seagull1.com'<br>realm: 'seagull1.com'<br>- identity: 'seagull1.com'<br>realm: 'seagull1.com' | Not applicable | the diameter client node information<br>*it should be yaml list<br>*default values is template, how to add peer nodes. |
| service.customExtension.labels | Custom Labels that needs to be added to nudr-diameterproxy specific Service. | null | Not applicable | This can be used to add custom label(s) to nudr-diameterproxy Service. |
| service.customExtension.annotations | Custom Annotations that needs to be added to nudr-diameterproxy specific Services. | null | Not applicable | This can be used to add custom annotation(s) to nudr-diameterproxy Service. |
| deployment.customExtension.labels | Custom Labels that needs to be added to nudr-diameterproxy specific Deployment. | null | Not applicable | This can be used to add custom label(s) to nudr-diameterproxy Deployment. |
| deployment.customExtension.annotations | Custom Annotations that needs to be added to nudr-diameterproxy specific Deployment. | null | Not applicable | This can be used to add custom annotation(s) to nudr-diameterproxy Deployment. |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| readinessProbe.initialDelaySeconds | Configurable wait time before performing the first readiness probe by the kubeletNote: Do not change this value. If you see delays in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 80 | Not Applicable<br>Unit: Seconds | |
| readinessProbe.periodSeconds | Time interval for every readiness probe check.Note: Do not change this value. If you see delays in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 5 | Not Applicable<br>Unit: Seconds | |
| livenessProbe.initialDelaySeconds | Configurable wait time before performing the first liveness probe by the kubelet.<br>Note: Do not change this value. If you see delays in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 80 | Not Applicable<br>Unit: Seconds | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| livenessProbe.periodSeconds | Time interval for every liveness probe check.<br>Note: Do not change this value. If you see delays in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 20 | Not Applicable<br>Unit: Seconds | |

Following table provides parameters for **ocudr-ingressgateway micro service (API Gateway)**

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| global.type | ocudr-ingressgateway service type | LoadBalancer | Possbile Values-<br>ClusterIP<br>NodePort<br>LoadBalancer | |
| global.metalLbIpAllocationEnabled | Enable or disable Address Pool for Metallb | true | true/false | |
| global.metalLbIpAllocationAnnotation | Address Pool for Metallb | metallb.universe.tf/address-pool: signaling | Not applicable | |
| global.staticNodePortEnabled | If Static node port needs to be set, then set staticNodePortEnabled flag to true and provide value for staticNodePort | false | Not applicable | |
| global.istioIngressTlsSupport.ingressGateway | Supports clear text traffic from outside of the cluster when enabled to try in case of Service Mesh Enabled. | false | true/false | |
| image.name | Docker image name | ocudr/ocingress_gateway | Not applicable | |
| image.tag | Image version tag | 1.8.1 | Not applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| image.pullPolicy | This setting will tell if image need to be pulled or not | Always | Possible Values -<br>Always<br>IfNotPresent<br>Never | |
| initContainersImage.name | Docker Image name | ocudr/configurationinit | Not applicable | |
| initContainersImage.tag | Image version tag | 1.4.0 | Not applicable | |
| initContainersImage.pullPolicy | This setting will tell if image need to be pulled or not | Always | Possible Values -<br>Always<br>IfNotPresent<br>Never | |
| updateContainersImage.name | Docker Image name | ocudr/configurationupdate | Not applicable | |
| updateContainersImage.tag | Image version tag | 1.4.0 | Not applicable | |
| updateContainersImage.pullPolicy | This setting will tell if image need to be pulled or not | Always | Possible Values -<br>Always<br>IfNotPresent<br>Never | |
| service.ssl.tlsVersion | Configuration to take TLS version to be used | TLSv1.2 | Valid TLS version | These are service fixed parameters |
| service.ssl.privateKey.k8SecretName | name of the secret which stores keys and certificates | ocudr-gateway-secret | Not applicable | |
| service.ssl.privateKey.k8NameSpace | namespace in which secret is created | ocudr | Not applicable | |
| service.ssl.privateKey.rsa.fileName | rsa private key stored in the secret | rsa_private_key_pkcs1.pem | Not applicable | |
| service.ssl.privateKey.ecdsa.fileName | ecdsa private key stored in the secret | ecdsa_private_key_pkcs8.pem | Not applicable | |
| service.ssl.certificate.k8SecretName | name of the secret which stores keys and certificates | ocudr-gateway-secret | Not applicable | |
| service.ssl.certificate.k8NameSpace | namespace in which secret is created | ocudr | Not applicable | |
| service.ssl.certificate.rsa.fileName | rsa certificate stored in the secret | apigatewayrsa.cer | Not applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| service.ssl.certificate.ecdsa.fileName | ecdsa certificate stored in the secret | apigatewayecdsa.cer | Not applicable | |
| service.ssl.caBundle.k8SecretName | name of the secret which stores keys and certificates | ocudr-gateway-secret | Not applicable | |
| service.ssl.caBundle.k8NameSpace | namespace in which secret is created | ocudr | Not applicable | |
| service.ssl.caBundle.fileName | ca Bundle stored in the secret | caroot.cer | Not applicable | |
| service.ssl.keyStorePassword.k8SecretName | name of the secret which stores keys and certificates | ocudr-gateway-secret | Not applicable | |
| service.ssl.keyStorePassword.k8NameSpace | namespace in which secret is created | ocudr | Not applicable | |
| service.ssl.keyStorePassword.fileName | keyStore password stored in the secret | key.txt | Not applicable | |
| service.ssl.trustStorePassword.k8SecretName | name of the secret which stores keys and certificates | ocudr-gateway-secret | Not applicable | |
| service.ssl.trustStorePassword.k8NameSpace | namespace in which secret is created | ocudr | Not applicable | |
| service.ssl.trustStorePassword.fileName | trustStore password stored in the secret | trust.txt | Not applicable | |
| service.initialAlgorithm | Algorithm to be used<br>ES256 can also be used, but corresponding certificates need to be used. | RSA256 | RSA256/ES256 | |
| resources.limits.cpu | Cpu allotment limitation | 5 | Not applicable | |
| resources.limits.memory | Memory allotment limitation | 4Gi | Not applicable | |
| resources.limits.initServiceCpu | Maximum amount of CPU that Kubernetes will allow the ingress-gateway init container to use. | 1 | Not Applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| resources.limits.initServiceMemory | Memory Limit for ingress-gateway init container | 1Gi | Not Applicable | |
| resources.limits.updateServiceCpu | Maximum amount of CPU that Kubernetes will allow the ingress-gateway update container to use. | 1 | Not Applicable | |
| resources.limits.updateServiceMemory | Memory Limit for ingress-gateway update container | 1Gi | Not Applicable | |
| resources.requests.cpu | Cpu allotment for ocudr-endpoint pod | 5 | Not Applicable | |
| resources.requests.memory | Memory allotment for ocudr-endpoint pod | 4Gi | Not Applicable | |
| resources.requests.initServiceCpu | The amount of CPU that the system guarantees for the ingress-gateway init container, and Kubernetes uses this value to decide on which node to place the pod. | | Not Applicable | |
| resources.requests.initServiceMemory | The amount of memory that the system will guarantee for the ingress-gateway init container, and Kubernetes will use this value to decide on which node to place the pod | | Not Applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| resources.requests.updateServiceCpu | The amount of CPU that the system will guarantee for the ingress-gateway update container, and Kubernetes will use this value to decide on which node to place the pod. | | Not Applicable | |
| resources.requests.updateServiceMemory | The amount of memory that the system will guarantee for the ingress-gateway update container, and Kubernetes will use this value to decide on which node to place the pod. | | Not Applicable | |
| resources.target.averageCpuUtil | CPU utilization limit for autoscaling | 80 | Not Applicable | |
| minAvailable | Number of pods always running | 2 | Not Applicable | |
| minReplicas | Min replicas to scale to maintain an average CPU utilization | 2 | Not applicable | |
| maxReplicas | Max replicas to scale to maintain an average CPU utilization | 5 | Not applicable | |
| log.level.root | Logs to be shown on ocudr-endpoint pod | WARN | valid level | |
| log.level.ingress | Logs to be shown on ocudr-ingressgateway pod for ingress related flows | INFO | valid level | |
| log.level.oauth | Logs to be shown on ocudr-ingressgateway pod for oauth related flows | INFO | valid level | |
| initssl | To Initialize SSL related infrastructure in init/update container | false | Not Applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| jaegerTracingEnabled | Enable/Disable Jaeger Tracing | false | true/false | |
| openTracing.jaeger.udpSender.host | Jaeger agent service FQDN | occne-tracer-jaeger-agent.occne-infra | Valid FQDN | |
| openTracing.jaeger.udpSender.port | Jaeger agent service UDP port | 6831 | Valid Port | |
| openTracing.jaeger.probabilisticSampler | Probablistic Sampler on Jaeger | 0.5 | Range: 0.0 - 1.0 | Sampler makes a random sampling decision with the probability of sampling. For example, if the value set is 0.1, approximately 1 in 10 traces will be sampled |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| | Supported cipher suites for ssl | - TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384<br>- TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384<br>- TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256<br>- TLS_DHE_RSA_WITH_AES_256_GCM_SHA384<br>- TLS_DHE_RSA_WITH_AES_256_CCM<br>- TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256<br>- TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 | Not applicable | |
| oauthValidatorEnabled | OAUTH Configuration | false | Not Applicable | |
| nfType | NFType of service producer | UDR | Not Applicable | Mandatory when oauthValidatorEna ebled is true |
| producerScope | Comma-seperated list of services hosted by service producer. | nudr-dr,nudr-group-id-map | Valid service list | Mandatory when oauthValidatorEna ebled is true |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| allowedClockSkewSeconds | Set this value if clock on the parsing NF (producer) is not perfectly in sync with the clock on the NF (consumer) that created the JWT. | 0 | Unit: Seconds | Mandatory when oauthValidatorEna ebled is true |
| nrfPublicKeyKubeSecret | Name of the secret which stores the public key(s) of NRF. | oauthsecret | Not Applicable | Mandatory when oauthValidatorEna ebled is true |
| nrfPublicKeyKubeNamespace | Namespace of the NRF publicKey Secret | ocudr | Not Applicable | Mandatory when oauthValidatorEna ebled is true |
| validationType | Values can be "strict" or "relaxed"."strict" means that incoming requests without "Authorization"(Access Token) header are rejected."relaxed" means that if incoming request contains "Authorization" header, it is validated. If incoming request does not contain "Authorization" header, validation is ignored. | strict | strict/relaxed | Mandatory when oauthValidatorEna ebled is true |
| producerPlmnMNC | MNC of service producer | 14 | Valid MNC | |
| producerPlmnMCC | MCC of service producer | 310 | Valid MCC | |
| enableIncomingHttp | Enabling for accepting http requests | true | Not Applicable | |
| enableIncomingHttps | Enabling for accepting https requests | false | true or false | |
| enableOutgoingHttps | Enabling for sending https requests | false | true or false | |
| maxRequestsQueuedPerDestination | Queue Size at the ocudr-endpoint pod | 5000 | Not Applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| maxConnectionsPerIp | Connections from endpoint to other microServices | 10 | Not Applicable | |
| serviceMeshCheck | Load balancing will be handled by Ingress gateway, if true it would be handled by serviceMesh | false | true/false | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| routesConfig | Routes configured to connect to different micro services of UDR | - id: traffic_mapping_http<br>uri: http://{{ .Release.Name }}-nudr-drservice:5001<br>path: /nudr-dr/**<br>order: 1<br>- id: traffic_mapping_http_prov<br>uri: http://{{ .Release.Name }}-nudr-drservice:5001<br>path: /nudr-dr-prov/**<br>order: 2<br>- id: traffic_mapping_http_mgmt<br>uri: http://{{ .Release.Name }}-nudr-drservice:5001<br>path: /nudr-dr-mgm/**<br>order: 3<br>- id: traffic_mapping_http_udsf<br>uri: http://{{ .Release.Name }}-nudr- | Not Applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| | | drservice:5001<br>path: /nudsf-dr/**<br>order: 4<br>- id: traffic_mapping_http_group<br>uri: http://{{ .Release.Name }}-nudr-drservice:5001<br>path: /nudr-group-id-map/**<br>order: 5<br>- id: traffic_mapping_http_group_prov<br>uri: http://{{ .Release.Name }}-nudr-drservice:5001<br>path: /nudr-group-id-map-prov/**<br>order: 6<br>- id: traffic_mapping_http_slf_group_prov<br>uri: http://{{ .Release.Name }}-nudr-drservice:5001<br>path: /slf-group-prov/**<br>order: 7 | | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| service.customExtension.labels | Custom Labels that needs to be added to ingressgateway specific service. | null | Not Applicable | This can be used to add custom label(s) to ingressgateway service. |
| service.customExtension.annotations | Custom Annotations that needs to be added to ingressgateway specific services. | null | Not Applicable | This can be used to add custom annotation(s) to ingressgateway service. |
| deployment.customExtension.labels | Custom Labels that needs to be added to ingressgateway specific deployment. | null | Not Applicable | This can be used to add custom label(s) to ingressgateway deployment. |
| deployment.customExtension.annotations | Custom Annotations that needs to be added to ingressgateway specific deployment. | null | Not Applicable | This can be used to add custom annotation(s) to ingressgateway deployment. |
| readinessProbe.initialDelaySeconds | Configurable wait time before performing the first readiness probe by the kubelet<br>Note: Do not change this value. If you see delays in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 30 | Not Applicable<br>Unit: Seconds | |
| readinessProbe.periodSeconds | Time interval for every readiness probe check.<br>Note: Do not change this value. If you see delays in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 10 | Not Applicable<br>Unit: Seconds | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| readinessProbe.timeoutSeconds | Number of seconds after which the probe times out<br>Note: Do not change this default value. | 3 | Not Applicable | |
| readinessProbe.successThreshold | Minimum consecutive successes for the probe to be considered successful after having failed<br>Note: Do not change this default value. | 1 | Not Applicable | |
| readinessProbe.failureThreshold | When a Pod starts and the probe fails, Kubernetes will try failureThreshold times before giving up<br>Note: Do not change this default value. | 3 | Not Applicable | |
| livenessProbe.initialDelaySeconds | Configurable wait time before performing the first liveness probe by the kubelet.<br>Note: Do not change this value. If you see delays in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 30 | Not Applicable<br>Unit: Seconds | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| livenessProbe.periodSeconds | Time interval for every liveness probe check.<br>Note: Do not change this value. If you see delays in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 15 | Not Applicable<br>Unit: Seconds | |
| livenessProbe.timeoutSeconds | Number of seconds after which the probe times out<br>Note: Do not change this default value. | 3 | Not Applicable | |
| livenessProbe.successThreshold | Minimum consecutive successes for the probe to be considered successful after having failed<br>Note: Do not change this default value. | 1 | Not Applicable | |
| livenessProbe.failureThreshold | When a Pod starts and the probe fails, Kubernetes will try failureThreshold times before giving up<br>Note: Do not change this default value. | 3 | Not Applicable | |

Following table provides parameters for **ocudr-egressgateway micro service (API Gateway)**

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| enabled | Configuration flag to enable/disable egress gateway | true | true/false | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| image.name | Docker image name | ocudr/ ocegress_gateway | Not applicable | |
| image.tag | Image version tag | 1.8.1 | Not applicable | |
| image.pullPolicy | This setting will tell if image need to be pulled or not | Always | Possible Values -<br>Always<br>IfNotPresent<br>Never | |
| initContainersImage.name | Docker Image name | ocudr/ configurationinit | Not applicable | |
| initContainersImage.tag | Image version tag | 1.4.0 | Not applicable | |
| initContainersImage.pullPolicy | This setting will tell if image need to be pulled or not | Always | Possible Values -<br>Always<br>IfNotPresent<br>Never | |
| updateContainersImage.name | Docker Image name | ocudr/ configurationupdate | Not applicable | |
| updateContainersImage.tag | Image version tag | 1.4.0 | Not applicable | |
| updateContainersImage.pullPolicy | This setting will tell if image need to be pulled or not | Always | Possible Values -<br>Always<br>IfNotPresent<br>Never | |
| resources.limits.cpu | Cpu allotment limitation | 3 | Not applicable | |
| resources.limits. memory | Memory allotment limitation | 4Gi | Not applicable | |
| resources.limits.initServiceCpu | Maximum amount of CPU that Kubernetes will allow the egress-gateway init container to use. | 1 | Not applicable | |
| resources.limits.initServiceMemory | Memory Limit for egress-gateway init container | 1Gi | Not applicable | |
| resources.limits.updateServiceCpu | Maximum amount of CPU that Kubernetes will allow the egress-gateway update container to use. | 1 | Not applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| resources.limits.updateServiceMemory | Memory Limit for egress-gateway update container | 1Gi | Not applicable | |
| resources.requests.cpu | Cpu allotment for ocudr-egressgateway pod | 3 | Not applicable | |
| resources.requests.memory | Memory allotment for ocudr-egressgatewaypod | 4Gi | Not applicable | |
| resources.requests.initServiceCpu | The amount of CPU that the system will guarantee for the egress-gateway init container, and Kubernetes will use this value to decide on which node to place the pod | | Not Applicable | |
| resources.requests.initServiceMemory | The amount of memory that the system will guarantee for the egress-gateway init container, and Kubernetes will use this value to decide on which node to place the pod | | Not Applicable | |
| resources.requests.updateServiceCpu | The amount of CPU that the system will guarantee for the egress-gateway update container, and Kubernetes will use this value to decide on which node to place the pod. | | Not Applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| resources.requests.updateServiceMemory | The amount of memory that the system will guarantee for the egress-gateway update container, and Kubernetes will use this value to decide on which node to place the pod. | | Not Applicable | |
| resources.target.averageCpuUtil | CPU utilization limit for autoscaling | 80 | Not applicable | |
| service.ssl.tlsVersion | Configuration to take TLS version to be used | TLSv1.2 | Valid TLS version | These are service fixed parameters |
| service.initialAlgorithm | Algorithm to be used<br>ES256 can also be used, but corresponding certificates need to be used. | RSA256 | RSA256/ES256 | |
| service.ssl.privateKey.k8SecretName | name of the secret which stores keys and certificates | ocudr-gateway-secret | Not applicable | |
| service.ssl.privateKey.k8NameSpace | namespace in which secret is created | ocudr | Not applicable | |
| service.ssl.privateKey.rsa.fileName | rsa private key stored in the secret | rsa_private_key_pkcs1.pem | Not applicable | |
| service.ssl.privateKey.ecdsa.fileName | ecdsa private key stored in the secret | ecdsa_private_key_pkcs8.pem | Not applicable | |
| service.ssl.certificate.k8SecretName | name of the secret which stores keys and certificates | ocudr-gateway-secret | Not applicable | |
| service.ssl.certificate.k8NameSpace | namespace in which secret is created | ocudr | Not applicable | |
| service.ssl.certificate.rsa.fileName | rsa certificate stored in the secret | apigatewayrsa.cer | Not applicable | |
| service.ssl.certificate.ecdsa.fileName | ecdsa certificate stored in the secret | apigatewayecdsa.cer | Not applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| service.ssl.caBundle.k8SecretName | name of the secret which stores keys and certificates | ocudr-gateway-secret | Not applicable | |
| service.ssl.caBundle.k8NameSpace | namespace in which secret is created | ocudr | Not applicable | |
| service.ssl.caBundle.fileName | ca Bundle stored in the secret | caroot.cer | Not applicable | |
| service.ssl.keyStorePassword.k8SecretName | name of the secret which stores keys and certificates | ocudr-gateway-secret | Not applicable | |
| service.ssl.keyStorePassword.k8NameSpace | namespace in which secret is created | ocudr | Not applicable | |
| service.ssl.keyStorePassword.fileName | keyStore password stored in the secret | key.txt | Not applicable | |
| service.ssl.trustStorePassword.k8SecretName | name of the secret which stores keys and certificates | ocudr-gateway-secret | Not applicable | |
| service.ssl.trustStorePassword.k8NameSpace | namespace in which secret is created | ocudr | Not applicable | |
| service.ssl.trustStorePassword.fileName | trustStore password stored in the secret | trust.txt | Not applicable | |
| minAvailable | Number of pods always running | 1 | Not Applicable | |
| minReplicas | Min replicas to scale to maintain an average CPU utilization | 1 | Not applicable | |
| maxReplicas | Max replicas to scale to maintain an average CPU utilization | 4 | Not applicable | |
| log.level.root | Logs to be shown on ocudr-egressgateway pod | WARN | valid level | |
| log.level.egress | Logs to be shown on ocudr-egressgateway pod for egress related flows | INFO | valid level | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| log.level.oauth | Logs to be shown on ocudr-egressgateway pod for oauth related flows | INFO | valid level | |
| fullnameOverride | Name to be used for deployment | ocudr-egressgateway | Not applicable | This config is commented by default. |
| initssl | To Initialize SSL related infrastructure in init/update container | false | Not Applicable | |
| jaegerTracingEnabled | Enable/Disable Jaeger Tracing | false | true/false | |
| openTracing.jaeger.udpSender.host | Jaeger agent service FQDN | occne-tracer-jaeger-agent.occne-infra | Valid FQDN | |
| openTracing.jaeger.udpSender.port | Jaeger agent service UDP port | 6831 | Valid Port | |
| openTracing.jaeger.probabilisticSampler | Probablistic Sampler on Jaeger | 0.5 | Range: 0.0 - 1.0 | Sampler makes a random sampling decision with the probability of sampling. For example if the value set is 0.1, approximately 1 in 10 traces will be sampled. |
| enableOutgoingHttps | Enabling for sending https requests | false | true or false | |
| oauthClient.enabled | Enable if oauth is required | false | true or false | Enable based on Oauth configuration |
| oauthClient.dnsSrvEnabled | DNS SRV Enabled for oAuth | false | true/false | |
| oauthClient.httpsEnabled | Determine if https support is enabled or not which is a deciding factor for oauth request scheme and search query parameter in dns-srv request | false | true/false | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| oauthClient.virtualFqdn | virtualFqdn value which needs to be populated and sent in the dns-srv query. | localhost:port | | Mandatory if oauthClient.dnsSrvEnabled is true |
| oauthClient.staticNrfList | List of Static NRF's | - localhost:port | | Mandatory if oauthClient.enabled is true |
| oauthClient.nfType | NFType of service consumer. | UDR | Not Applicable | Mandatory if oauthClient.enabled is true |
| oauthClient.consumerPlmnMNC | MNC of service Consumer. | 14 | Valid MNC | |
| oauthClient.consumerPlmnMCC | MCC of service Consumer. | 310 | Valid MCC | |
| oauthClient.maxRetry | Maximum number of retry that need to be performed to other NRF Fqdn’s in case of failure response from first contacted NRF based on the errorCodeSeries configured. | 2 | Valid Number | Mandatory if oauthClient.enabled is true |
| oauthClient.apiPrefix | apiPrefix that needs to be appended in the Oauth request flow. | "" | Valid String | Mandatory if oauthClient.enabled is true |
| oauthClient.errorCodeSeries | Determines the fallback condition to other NRF in case of failure response from currently contacted NRF. | 4XX | Valid series | Mandatory if oauthClient.enabled is true and requires different error code series |
| oauthClient.retryAfter | RetryAfter value in milliseconds that needs to be set for a particular NRF Fqdn, if the error matched the configured errorCodeSeries. | 5000 | Unit: Milliseconds | Mandatory if oauthClient.enabled is true |
| maxConcurrentPushedStreams | Jetty client configuration | 1000 | Valid Number | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| maxRequestsQueuedPerDestination | Jetty client configuration | 1024 | Valid Number | |
| maxConnectionsPerIp | Max Connections allowed per Ip | 4 | Valid Number | |
| connectionTimeout | Connection timeout in milli seconds | 10000 | Unit: Milliseconds | |
| requestTimeout | Request Timeout in milli seconds | 1000 | Unit: Milliseconds | |
| jettyIdleTimeout | Jetty Idle Timeout in milli seconds | 0 | Unit: Milliseconds<br>#(ms,<=0 -> to make timeout infinite) | |
| k8sServiceCheck | Enable this if loadbalancing is to be done by egress instead of K8s | false | true/false | |
| service.customExtension.labels | Custom Labels that needs to be added to egressgateway specific Service. | null | Not applicable | This can be used to add custom label(s) to egressgateway Service. |
| service.customExtension.annotations | Custom Annotations that needs to be added to egressgateway specific Services. | null | Not applicable | This can be used to add custom annotation(s) to egressgateway Service. |
| deployment.customExtension.labels | Custom Labels that needs to be added to egressgateway specific Deployment. | null | Not applicable | This can be used to add custom label(s) to egressgateway Deployment. |
| deployment.customExtension.annotations | Custom Annotations that needs to be added to egressgateway specific Deployment. | null | Not applicable | This can be used to add custom annotation(s) to egressgateway deployment. |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| readinessProbe.initialDelaySeconds | Configurable wait time before performing the first readiness probe by the kubelet<br>Note: Do not change this value. If you see delays in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 30 | Not Applicable<br>Unit: Seconds | |
| readinessProbe.periodSeconds | Time interval for every readiness probe check.<br>Note: Do not change this value. If you see delays in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 10 | Not Applicable<br>Unit: Seconds | |
| readinessProbe.timeoutSeconds | Number of seconds after which the probe times out<br>Note: Do not change this default value. | 3 | Not Applicable | |
| readinessProbe.successThreshold | Minimum consecutive successes for the probe to be considered successful after having failed<br>Note: Do not change this default value. | 1 | Not Applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| readinessProbe.failureThreshold | When a Pod starts and the probe fails, Kubernetes will failureThreshold times before giving up<br>Note: Do not change this default value. | 3 | Not Applicable | |
| livenessProbe.initialDelaySeconds | Configurable wait time before performing the first liveness probe by the kubelet.<br>Note: Do not change this value. If you see delays in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 30 | Not Applicable<br>Unit: Seconds | |
| livenessProbe.periodSeconds | Time interval for every liveness probe check.<br>Note: Do not change this value. If you see delays in pod coming up and probe is killing the pod then you should consider tuning these parameters. | 15 | Not Applicable<br>Unit: Seconds | |
| livenessProbe.timeoutSeconds | Number of seconds after which the probe times out<br>Note: Do not change this default value. | 3 | Not Applicable | |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| livenessProbe.successThreshold | Minimum consecutive successes for the probe to be considered successful after having failed<br>Note: Do not change this default value. | 1 | Not Applicable | |
| livenessProbe.failureThreshold | When a Pod starts and the probe fails, Kubernetes will try failureThreshold times before giving upNote: Do not change this default value. | 3 | Not Applicable | |

# Configuring User Parameters - Bulk Import Tool

Following table provides parameters for creating PVC and template for installing **Bulk Import Tool**.

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| storageClassName | storage class used for creation of PVC | standard | Not applicable | Used for creation of PVC |
| storage | Volume for PVC | 10Gi | Not applicable | Volume assigned for creation of PVC. Value for the volume should be more than set of CSV files required for import operaiton. |
| name | name assigned to the bulkimport pod under metadata section in template yaml file | ocudr-nudr-bulk-import | Not applicable | Use `releaseName-nudr-bulk-import` |
| MYSQL_DATABASE | Uses the default secret name from the UDR deployment | ocudr-secrets | Not Applicable | Uses ocudr-secrets of the UDR deployment |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| DB_SERVICE_NAME | Defines the mysql-connectivity service used for connecting to mysql database | mysql-connectivity-service.occne-infra | Possible Values:-<br>mysql-connectivity-service.<namespace> | mysql-connectivity service yaml file can be used for connecting to sql nodes. |
| DB_SERVICE_PORT | Port for the mysql-connectivity service | 3306 | Not Applicable | mysql-connectivity service port for connecting to mysql database. |
| LOGGING_LEVEL_ROOT | Log Level | INFO | Possible Values -<br>WARN<br>INFO<br>DEBUG | The log level of the ocudr-nudr-bulk-import pod |
| POLLING_TIME_INTERVAL | Defines the time interval in milliseconds for pooling the import directory | 5000 milliseconds | Not applicable | Time interval to be specified in milliseconds |
| HTTP_RETRY_CONFIGURE | retrycount for the okhttp3 client to send request to the udr in case of service unavailable error scenarios | 2 | Not applicable | Retry Count to be specified for the okhttp3 client to send request to udr. |
| HIKARI_POOL_SIZE | Mysql Connection pool size | 10 | Not Applicable | The hikari pool connection size to be created at start up |
| DATA_SERVICE_BASEURL | FQDN | ocudr-ingressgateway.myudr.svc.cluster.local | Not Applicable | FQDN used by the bulk import tool to send reques to UDR.<br>Note: Be cautious in updating this value. Should consider helm release name, namespace used for udr deployment and name resolution setting in k8s. |
| image | image from the docker registry along with image name and tag | ocudr-registry.us.oracle.com:5000/ocudr/nudr_bulk_import:1.8.0 | Not applicable | docker-registry / imageName:<imagetag> |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| imagePullPolicy | This setting will tell if image need to be pulled or not | Always | Possible Values -<br>WARN<br>INFO<br>DEBUG | |
| MANAGEMENT_SERVER_PORT | management server port | 9000 | Not Applicable | The actuator management port to be used for nudr-bulk-import pod |
| fsGroup | Docker Id given in the Docker file | 1002 | Not Applicable | Use the same docker id given in the docker file |
| resources.requests.cpu | Cpu Allotment for nudr-bulk-import pod | 4 | Not applicable | The CPU to be allocated for nudr-bulk-import pod during install of tool |
| resources.requests.memory | Memory allotment for nudr-bulk-import pod | 6Gi | Not applicable | The memory to be allocated for nudr-bulk-import pod during install of tool |
| resources.limits.cpu | Cpu allotment limitation | 4 | Not applicable | The CPU to be max allocated for nudr-bulk-import pod |
| resources.limits.memory | Memory allotment limitation | 6Gi | Not applicable | The memory to be max allocated for nudr-bulk-import pod |

# Configuring User Parameters for Migration Tool

Following table provides parameters for installing **Migration Tool**.

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| name | name assigned to the migration pod under metadata section in template yaml file | ocudr-nudr-migration | Not applicable | Use `releaseName-nudr-migration` |
| nodeSelectorTerms (values) | Update the value as node name in which you want to run | 5g-udr-dev-1-k8s-node-1 | Not applicable | update this value to k8 worker node name |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| MYSQL_DATABASE | Uses the default secret name from the UDR deployment | ocudr-secrets | Not Applicable | Uses ocudr-secrets of the UDR deployment |
| DB_SERVICE_NAME | Defines the mysql-connectivity service used for connecting to mysql database | mysql-connectivity-service.occne-infra | Possible Values:-<br>mysql-connectivity-service.<namespace> | mysql-connectivity service yaml file can be used for connecting to sql nodes. |
| DB_SERVICE_PORT | Port for the mysql-connectivity service | 3306 | Not Applicable | mysql-connectivity service port for connecting to mysql database. |
| LOGGING_LEVEL_ROOT | Log Level | WARN | Possible Values -<br>WARN<br>INFO<br>DEBUG | The log level of the `ocudr-nudr-migration` pod |
| K8S_HOST_IP | HOST IP of the pod running node external ip |  | Not applicable | pick the node external ip of pod running based on the node affinity rules. |
| START_RANGE | Defines the range of subscriber data in which migration should start. |  | Not applicable |  |
| END_RANGE | Defines the range of subscriber data in which migration should end. |  | Range |  |
| KEY_TYPE | Defines the key type of which data migration should happen.Ex:MSISDN,IMSI... |  | Range |  |
| HTTP_RETRY_CONFIGURE | retrycount for the okhttp3 client to send request to the udr in case of service unavailable error scenarios | 2 | Not applicable | Retry Count to be specified for the okhttp3 client to send request to udr. |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| HIKARI_POOL_SIZE | Mysql Connection pool size | 10 | Not Applicable | The hikari pool connection size to be created at start up |
| UDR_SERVICE_BASEURL | FQDN | ocudr-ingressgateway.myudr.svc.cluster.local | Not Applicable | FQDN used by the migration tool to send reques to UDR.<br>Note: Be cautious in updating this value. Should consider helm release name, namespace used for udr deployment and name resolution setting in k8s. |
| image | image from the docker registry along with image name and tag | ocudr-registry.us.oracle.com:5000/ocudr/nudr_migration:1.7.30 | Not applicable | docker-registry / imageName:<imagetag> |
| imagePullPolicy | This setting will tell if image need to be pulled or not | Always | Possible Values -<br>WARN<br>INFO<br>DEBUG | |
| MANAGEMENT_SERVER_PORT | management server port | 9000 | Not Applicable | The actuator management port to be used for nudr-bulk-import pod |
| resources.requests.cpu | Cpu Allotment for nudr-bulk-import pod | 4 | Not applicable | The CPU to be allocated for nudr-bulk-import pod during install of tool |
| resources.requests.memory | Memory allotment for nudr-bulk-import pod | 5Gi | Not applicable | The memory to be allocated for nudr-bulk-import pod during install of tool |
| resources.limits.cpu | Cpu allotment limitation | 4 | Not applicable | The CPU to be max allocated for nudr-bulk-import pod |
| resources.limits.memory | Memory allotment limitation | 5Gi | Not applicable | The memory to be max allocated for nudr-bulk-import pod |

| Parameter | Description | Default value | Range or Possible Values (If applicable) | Notes |
|---|---|---|---|---|
| DIAMETER_REALM | Dimeter Realm to be configured from 4G side | udr.oracle.com | Not applicable | Its a client Realm and needs to be configured in 4G setup |
| DIAMETER_IDENTITY | Dimeter Identity to be configured from 4G side | udr.migration.oracle.com | Not applicable | Its a client Identithy and needs to be configured in 4G setup |
| DIAMETER_SETTING_NUM_OF_CONNECTIONS | Number of connections required to connect | 3 | Not applicable | Number of connections |
| DIAMETER_NODES_HOST | 4G server Host IP Address | 10.75.214.207 | Not applicable | This IP configures from 4G side |
| DIAMETER_NODES_PORT | 4G server Host Listener Port | 3868 | Not applicable | This port configures from 4G side |
| DIAMETER_NODES_REALM | 4G server Realm | tekelec.com | Not applicable | This Realm configures from 4G side |
| DIAMETER_NODES_IDENTITY | 4G server Identity | local.tekelec.com | Not applicable | This Identity configures from 4G side |
| APPLICATION_NAME | Application name of nudr-migration | ocudr | Not Applicable | Application name of nudr-migration |
| ENGINEERING_VERSION | Release version of ocudr | 1.8.0 | Not Applicable | Release version of ocudr |
| MARKETING_VERSION | Marketing version of ocudr | 1.8.0.0 | Not Applicable | Marketing version of ocudr |
| MICROSERVICE_NAME | MIcroservice name of migration tool | ocudr-nudr-migration | Not Applicable | MIcroservice name of migration tool |
| K8S_CLUSTER_NAME | Cluster name of migration tool | ocudr | Not Applicable | Cluster name of migration tool |
| K8S_NAMESPACE | Namespace in which you deployed | ocudr | Not Applicable | Namespace in which you deployed |
| K8S_NODE | Node name in which you deployed this tool | 5g-udr-dev-1-k8s-node-1 | Not Applicable | Node name in which you deployed this tool |

# 3 Bulk Import Provisioning

In this section, you will learn to install and use Bulk Import tool.

## Understanding Bulk Import Tool

With the help of **Bulk Import** tool, you can provision 5G UDR subscribers data in bulk. This tool:

- Reads the subscriber data from a text file in the CSV format.
- Must be run manually as it reads the CSV file (pre-determined format) from a specified directory path present on the pod.
- Sends requests to the UDR for provisioning the subscribers.
- Supports PCF subscribers data and NRF(SLF) subscribers data.
- Supports different type of operations: CREATE, UPDATE, DELETE.

> **Note:**
>
> - The Bulk Import Tool uses PVC (Persistent Volume Claim) and PV (Persistent Volume) that you must create before deploying the tool. For more details on creating PVC and PV, refer to UDR installation document.
> - The user/operator needs to manually copy all the CSV files to the PV mountpath inside the container, which is /home/udruser/import/.

## Viewing CSV File Format

The CSV file format is as follows:

| Key Type | Key value | OpType | Type | Subs_data(JSON) |
|---|---|---|---|---|
| MSISDN | 1234 | CREATE | PCF | { .. } |
| IMSI | 6786677 | MODIFY | PCF | { .. } |
| NAI | abcd@oracle.com | DELETE | PCF | { .. } |

> **Note:**
>
> The **subs_data** column must have JSON body enclosed in double quotes(").

To import CSV, you can use the following coding:

```
keyType,keyValue,operationType,nfType,jsonPayload
msisdn,1111111114,CREATE,PCF,
"{
  "profile-data": {
    "MSISDN": [
      "9111111112",
      "9211111112",
      "1111111114"
    ]
  },
  "policy-data": {
    "am-data": {
      "praInfos": {
        "p1": {
          "praId": "p1",
          "trackingAreaList": [
            {
              "plmnId": {
                "mcc": "976",
                "mnc": "32"
              },
              "tac": "5CB6"
            },
            {
              "plmnId": {
                "mcc": "977",
                "mnc": "33"
              },
              "tac": "5CB7"
            }
          ],
          "ecgiList": [
            {
              "plmnId": {
                "mcc": "976",
                "mnc": "32"
              },
              "eutraCellId": "92FFdBE"
            },
            {
              "plmnId": {
                "mcc": "977",
                "mnc": "33"
              },
              "eutraCellId": "8F868C4"
            }
          ],
          "ncgiList": [
            {
              "plmnId": {
                "mcc": "976",
                "mnc": "32"
              },
              "nrCellId": "b2fB6fE9D"
            },
            {
              "plmnId": {
                "mcc": "977",
                "mnc": "33"
              },
              "nrCellId": "5d1B4127b"
            }
          ],
          "globalRanNodeIdList": [
            {
              "plmnId": {
                "mcc": "965",
                "mnc": "235"
              },
              "n3IwfId": "fFf0f2AFbFa16CEfE7"
            },
            {
              "plmnId": {
                "mcc": "967",
                "mnc": "238"
              },
              "gNbId": {
                "bitLength": 25,
                "gNBValue": "1A8F1D"
              }
            }
          ]
        }
      },
      "subscCats": [
        "cat1",
        "cat2"
      ]
    }
  }
}"

accountID,12345678912345678912345678,MODIFY,SLF,"{
    "profile-data": {
        "imsi": [
            "2222222221",
            "2222222222"
        ],
        "nai": [
            "test@vzw.com"
        ],
        "accountID": [
            "12345678912345678912345678"
        ],
        "msisdn": [
            "19195225555",
            "19195225556"
        ],
        "extid": [
            "user@vzw.com"
        ]
    },
    "slfGroupName": "IMSGrp1"
}"
```

# Features Supported

The Bulk Import tool supports following operation types:

- **CREATE**: Allows to create subscribers with all keys mentioned in the payload. The PCF create and SLF create request payloads are available in the REST Guide.
- **UPDATE**: Allows to update the existing subscribers with new data mentioned in the **subs_data** column.

  > **Note:**
  >
  > You should provide complete subscriber information while using this operation as it replaces the old information with new data.

- **DELETE**: Allows to delete complete subscriber information with all keys attached to the subscriber.

The Bulk Import tool supports following NF Types:

- **PCF**: Using this tool, you can provision all 5G PCF data in bulk for all resource types: AM data, SM Data and UEPolicySet. You can refer to: Operations supported for PCF data and Management URI#Provisioningsystemasconsumer
- **SLF**: You can create SLF Subscribers with all keys and SLFGroupName in the payload. For more details, you can refer to: SLF lookup support on UDR#SubscriberProvisioningAPIsonUDR(SLF)

> **Note:**
>
> The Bulk Import tool does not support SLFGroupName provisioning. You can provision this using REST APIs or CNC-Console GUI.

# Installing Bulk Import Tool

You can implement the Bulk Import tool as a kubernetes job that comes up as a pod when you deploy it using a template yaml file. Before installing the Bulk Import tool, you must create a PVC.

**Creating Persistent Volume Claim (PVC)**

To create a PVC:

1. Create a namespace and then create a PVC using below yaml file:

   ```
   apiVersion: v1
   kind: PersistentVolumeClaim
   metadata:
     name: bulkimportpersistentclaim
   spec:
     storageClassName: #<Please Provide your StorageClass  Name>
     accessModes:
       - ReadWriteOnce
     resources:
       requests:
         storage: 10Gi
   ```

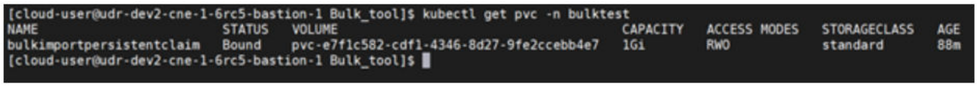
   > **Note:**
   >
   > You must name the PVC as 'bulkimportpersistentclaim'.

2. After creating a yaml file, execute the following command to create a PVC under the namespace:
   `kubectl create -f <yaml file for PVC formed using the above code block> -n <namespace>`

   **Example:** `kubectl create -f bulk_pvc.yaml -n bulktest`

3. Execute the following command to verify whether PVC is created or not:
   `kubectl get pvc -n <namespace>`

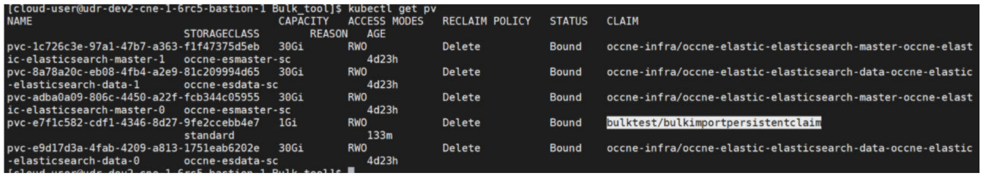
   **Figure 3-1 Verifying PVC Creation**

   ```
   [cloud-user@udr-dev2-cne-1-6rc5-bastion-1 Bulk_tool]$ kubectl get pvc -n bulktest
   NAME                        STATUS   VOLUME                                     CAPACITY   ACCESS MODES   STORAGECLASS   AGE
   bulkimportpersistentclaim   Bound    pvc-e7f1c582-cdf1-4346-8d27-9fe2ccebb4e7   1Gi        RWO            standard       88m
   [cloud-user@udr-dev2-cne-1-6rc5-bastion-1 Bulk_tool]$ 
   ```

4. Execute the following command to verify whether PV is created or not:
   `kubectl get pv`

   **Figure 3-2 Verifying PV Creation**

   ```
   [cloud-user@udr-dev2-cne-1-6rc5-bastion-1 Bulk_tool]$ kubectl get pv
   NAME                                       CAPACITY   ACCESS MODES   RECLAIM POLICY   STATUS   CLAIM
                              STORAGECLASS         REASON   AGE
   pvc-1c726c3e-97a1-47b7-a363-f1f47375d5eb   30Gi       RWO            Delete           Bound    occne-infra/occne-elastic-elasticsearch-master-occne-elast
   ic-elasticsearch-master-1   occne-esmaster-sc             4d23h
   pvc-8a78a20c-eb08-4fb4-a2e9-81c209994d65   30Gi       RWO            Delete           Bound    occne-infra/occne-elastic-elasticsearch-data-occne-elastic
   -elasticsearch-data-1       occne-esdata-sc               4d23h
   pvc-adba0a09-806c-4450-a22f-fcb344c05955   30Gi       RWO            Delete           Bound    occne-infra/occne-elastic-elasticsearch-master-occne-elast
   ic-elasticsearch-master-0   occne-esmaster-sc             4d23h
   pvc-e7f1c582-cdf1-4346-8d27-9fe2ccebb4e7   1Gi        RWO            Delete           Bound    bulktest/bulkimportpersistentclaim
                              standard                      133m
   pvc-e9d17d3a-4fab-4209-a813-1751eab6202e   30Gi       RWO            Delete           Bound    occne-infra/occne-elastic-elasticsearch-data-occne-elastic
   -elasticsearch-data-0       occne-esdata-sc               4d23h
   [cloud-user@udr-dev2-cne-1-6rc5-bastion-1 Bulk_tool]$ 
   ```

### Installing Bulk Import Tool as a Job

To install Bulk Import tool as a job:

1. Use the below template yaml file to install Bulk Import tool as a job:

   ```
   apiVersion: batch/v1
   kind: Job
   ```

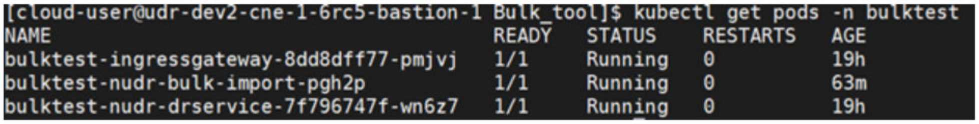
```
metadata:
  name: ocudr-nudr-bulk-import     # <Please use releaseName-nudr-
bulk-import>
  namespace: ocudr                 # <Use the namespace created>
spec:
  backoffLimit: 0
  template:
    metadata:
      name: ocudr-nudr-bulk-import  # <Please use releaseName-nudr-
bulk-import>
      annotations:
        "prometheus.io/port": "9000"
        "prometheus.io/path": "/actuator/prometheus"
        "prometheus.io/scrape": "true"
    spec:
      restartPolicy: Never
      securityContext:
        fsGroup: 1002
      volumes:
        - name: bulkimportpersistentvolume
          persistentVolumeClaim:
            claimName: bulkimportpersistentclaim
      containers:
      - env:
        - name: MYSQL_DATABASE
          valueFrom:
            secretKeyRef:
              name: ocudr-secrets
              key: dbname
        - name: DATASOURCE_USERNAME
          valueFrom:
            secretKeyRef:
              name: ocudr-secrets
# <Please use secrets created under UDR deployment >
              key: dsusername
 # < key for username mentioned in secrets yaml file under UDR
deployment >
        - name: DATASOURCE_PASSWORD
          valueFrom:
            secretKeyRef:
              name: ocudr-secrets
              key: dspassword
 # < key for password mentioned in secrets yaml file under UDR
deployment >
        - name: DB_SERVICE_NAME
          value: "mysql-connectivity-service.occne-infra"
 # <Default mysql-connectivity service used under occne-infra
namespace >
        - name: DB_SERVICE_PORT
          value: "3306"
 # <Default port for mysql-connectivity service >
        - name: HIKARI_POOL_SIZE
          value: "10"
        - name: LOGGING_LEVEL_ROOT
          value: "INFO"
        - name: MANAGEMENT_SERVER_PORT
          value: "9000"
        - name: DATA_SERVICE_BASEURL
          value: "http://ocudr-ingressgateway.ocudr:80"
 # <Please use servicename.namespace:portNumber of
ingressgateway>
        - name: POLLING_TIME_INTERVAL
# <Time interval in milliseconds after which the csv file is polled
>
          value: "5000"
        - name: HTTP_RETRY_CONFIGURE
# <Number of retries made to dr-service for failed request>
          value: "2"
        - name: APPLICATION_NAME
          value: "ocudr-nudr-bulk-import"
        - name: ENGINEERING_VERSION
          value: "1.8.0"
        - name: MARKETING_VERSION
          value: "1.8.0.0.0"
        - name: MICROSERVICE_NAME
          value: "ocudr-nudr-bulk-import"
        - name: K8S_CLUSTER_NAME
          value: "ocudr"
        - name: K8S_NAMESPACE
          value: "ocudr"
        - name: K8S_NODE
          value: "5g-udr-dev-1-k8s-
node-2"
        name: ocudr-nudr-bulk-import
        image: "ocudr-registry.us.oracle.com:5000/ocudr/
nudr_bulk_import:1.8.0"
# <Use the dockerregistry path for image:image tag>
        imagePullPolicy: Always
        volumeMounts:
          - name: bulkimportpersistentvolume
            mountPath: /home/udruser/import
# <Path where csv files needs to be placed for polling>
        ports:
        - containerPort: 9000
          name: management
        resources:
          requests:
            cpu: "4"
            memory: "6Gi"
          limits:
            cpu: "4"
            memory: "6Gi"
```

2. Execute the following command to install Bulk Import tool as a job:
   `kubectl create -f <template yaml >-n <namespace>`

   **Example:** `kubectl create -f bulk_import_tool.yaml -n bulktest`

   **Output:** Container comes up as shown below:

**Figure 3-3    Bulk Import Install Output**



```
[cloud-user@udr-dev2-cne-1-6rc5-bastion-1 Bulk_tool]$ kubectl get pods -n bulktest
NAME                                       READY   STATUS    RESTARTS   AGE
bulktest-ingressgateway-8dd8dff77-pmjvj    1/1     Running   0          19h
bulktest-nudr-bulk-import-pgh2p            1/1     Running   0          63m
bulktest-nudr-drservice-7f796747f-wn6z7    1/1     Running   0          19h
```

3. Execute the following command to check the pod details:
   `kubectl describe pods <nudr-bulk-import pod name> -n <namespace>`

   **Example:** `kubectl describe pods bulktest-nudr-bulk-import-pgh2p -n bulktest`

4. After the pod is up and running, copy the csv file. The Bulk Import tool starts processing the records.

## Using Bulk Import Tool

By default, the Bulk Import tool polls the **/home/udruser/import** path every 5000 ms. You can use the Bulk Import tool when:

- The **nudr-bulk-import** pod is up and running.
- Execute the following command to place the file inside the container:
  `kubectl cp <csv file> <namespace>/<nudr-bulk-import pod name>:/home/udruser/import`

  `kubectl cp create_pcf.csv bulktest/bulktest-nudr-bulk-import-pgh2p:/home/udruser/import`
- Login to the bash of container and execute the following command to check whether file is present inside the container or not:
  `kubectl exec -it <nudrbulk-import pod name>-n <namespace> -- bash`

  **Example:** `kubectl exec -it bulktest-nudr-bulk-import-pgh2p -n bulktest -- bash`

  > **Note:**
  >
  > After login to bash, you can find the file under **/home/udruser/import** location.

  ```
  bash-4.2$ pwd
    /home/udruser/import
  bash-4.2$ ls -l
     -rw-r--r--. 1 root udruser 28471 Aug 19 09:33 create_pcf.csv
  ```
- The Bulk Import tool processes each CSV file at a time and provisions the subscribers on 5G UDR. It's summary report is available in the **/home/udruser/import** path with filename (<CSV_filename>.log). You can monitor its metrics on Prometheus/Grafana.
  **Sample: create_pcf.log**

  ```
  bash-4.2$ pwd
    /home/udruser/import
  ```

```
bash-4.2$ ls -l
   -rw-r--r--. 1 root udruser 28471 Aug 19 09:33 create_pcf.csv
create_pcf.log
```

- The Bulk Import container logs contains the summary report, when the INFO logging level is enabled.
- Execute the following command to copy the report from pod container:
  `kubectl cp <pod name>:home/udruser/import/modify_slf.log <file name> -n <namespace>`

  **Example:** `kubectl cp ocudr-nudr-bulk-import-8g8c8:home/udruser/import/modify_slf.log modify_slf.log -n myudr`

# 4

# Migrating Subscriber Data from 4G UDR to 5G UDR

### Overview

The **Migration Tool** is an external tool that you can use to migrate the subscriber data from source UDR (4G UDR) to target UDR (5G UDR). It is not part of default 5G UDR charts and hence, you need to install it manually. This tool:

- Takes configurations for the range of subscribers, key type from chart.
- Uses Diameter interface of source 4G UDR to connect and read the subscriber information.
- Sends the provisioning requests to target 5G UDR via Ingress Gateway to provision the subscriber information read from source 4G UDR.
- Adds a report having details of the subscribers migrated, to the logs at the INFO Log level when the migration completes.

### Prerequisites

The prerequisites to use the Migration tool are as follows:

- 4G UDR and 5G UDR should be in active and running state. It is required to start the migration process when you install the yaml file.
- The XSI IP network of 4G MP nodes must be accessible from kubernetes nodes, where you are deploying.
- Enable subscriber auto create as true in the nudr-dr section of the values yaml file.

## Using the Migration Tool

You can use the Migration tool only if:

- There exists an active source i.e. 4G UDR and destination source i.e. 5G UDR and they are up and running.
- It is installed as per the installation process shared at Installing the Migration Tool.
- The Diameter connections on 4G UDR are confiigured. It means that the peer node information and connections are created to identify the Migration tool as a known Diameter client.
- You can verify in 5G UDR that all 4G UDR records are migrated in a specified range when the migration job completes.
- You can observe its metrics that are pegged in the Grafana dashboard.

Below sample report shows the success and failure records count in the INFO logs of nudr-migration job.

```
"Migration Tool Summary Report.
 Time taken for processing: 0min 1sec | Total Number of records: 10 |
 Total Number of source UDR success: 10 |  Total Number of source UDR 
subscriber not found: 0 |
 Total Number of target UDR keys already exists: 0 |Total Number of 
source UDR failure: 0 |
 Total Number of target UDR success: 10 | Total Number of target UDR 
failure: 0 |
 List of failed source UDR keys:  |  List of failed subscriber not 
exist keys at source:  |
 List of keys aleady exists in target UDR:  | List of failed target UDR 
keys: "
```

## Installing the Migration Tool

You can deploy the Migration tool as a separate Kubernetes job. To install the Migration tool, you need to edit a yaml file that has configuration details of the migration tool.

A sample **nudr-migration yaml** file is given below:

```
apiVersion: batch/v1
kind: Job
metadata:
  name: ocudr-nudr-migration      # <Please use releaseName-nudr-
migration>
  namespace: ocudr                # <Use the namespace created>
spec:
  backoffLimit: 0
  template:
    metadata:
      name: ocudr-nudr-migration   # <Please use releaseName-nudr-
migration>
      annotations:
        "prometheus.io/port": "9000"
        "prometheus.io/path": "/actuator/prometheus"
        "prometheus.io/scrape": "true"
    spec:
      affinity:
        nodeAffinity:
          requiredDuringSchedulingIgnoredDuringExecution:
            nodeSelectorTerms:
              - matchExpressions:
                - key: kubernetes.io/hostname
                  operator: In
                  values:
                  - 5g-udr-dev-1-k8s-node-2
        #<Update this value as nodename of which you want to run this 
migration tool>
      restartPolicy: Never
      containers:
        - name: nudr-migration
          image: "ocudr-registry.us.oracle.com:5000/ocudr/
nudr_migration:1.7.40"
        # <Use the dockerregistry path for image:image tag>
          imagePullPolicy: Always
          resources:
            requests:
              cpu: "3"
              memory: "5Gi"
            limits:
              cpu: "3"
              memory: "5Gi"
          env:
            - name: MYSQL_DATABASE
              valueFrom:
                secretKeyRef:
                  name: ocudr-secrets
                  key: dbname
            - name: DATASOURCE_USERNAME
              valueFrom:
                secretKeyRef:
                  name: ocudr-secrets
         # <Please use secrets created under UDR deployment >
                  key: dsusername
         # < key for username mentioned in secrets yaml file under UDR
deployment >
            - name: DATASOURCE_PASSWORD
              valueFrom:
                secretKeyRef:
                  name: ocudr-secrets
                  key: dspassword
            # < key for password mentioned in secrets yaml file under
UDR deployment >
            - name: DB_SERVICE_NAME
              value: "mysql-connectivity-service.occne-infra"
            # <Default mysql-connectivity service used under occne-
infra namespace>
            - name: DB_SERVICE_PORT
              value: "3306"
            # <Default port for mysql-connectivity service>
            - name: HIKARI_POOL_SIZE
              value: "10"
            - name: LOGGING_LEVEL_ROOT
              value: "INFO"
            - name: K8S_HOST_IP
              value: "10.75.229.65"
            #< Update this host ip as node external ip in which you
should run >
            - name: START_RANGE
              value: "72111100000001"
            - name: END_RANGE
              value: "72111100000100"
            - name: KEY_TYPE
              value: "msisdn"
            - name: DELETE_SOURCE_UDR_USER
              value: "false"
            - name: UDR_SERVICE_BASEURL
              value: "http://ocudr-ingressgateway:80"
            # <Please use namespace-servicename:80 of ingressgateway>
            - name: HTTP_FAILED_RETRY_COUNT
```

```
              value: "5"                            # Target udr http
retry count
            - name: DIAMETER_REALM
              value: "udr.oracle.com"               # Diameter Client
Realm
            - name: DIAMETER_IDENTITY
              value: "udr.migration.oracle.com"     # Diameter Client
Identity
            - name: DIAMETER_SETTING_NUM_OF_CONNECTIONS
              value: "3"
            # Diameter Client Number of Connections
            - name: DIAMETER_NODE_HOST
              value: "10.75.214.207"                # Diameter Server
Host
            - name: DIAMETER_NODE_PORT
              value: "3868"                         # Diameter Server
Port
            - name: DIAMETER_NODE_REALM
              value: "tekelec.com"                  # Diameter Server
Realm
            - name: DIAMETER_NODE_IDENTITY
              value: "local.tekelec.com"            # Diameter Server
Identity
            - name: APPLICATION_NAME
              value: "ocudr"
            - name: ENGINEERING_VERSION
              value: "1.8.0"
            - name: MARKETING_VERSION
              value: "1.8.0.0.0"
            - name: MICROSERVICE_NAME
              value: "ocudr-nudr-migration"
            - name: K8S_CLUSTER_NAME
              value: "ocudr"
            - name: K8S_NAMESPACE
              value: "ocudr"
            - name: K8S_NODE
              value: "5g-udr-dev-1-k8s-node-2"
```

To install the Migration tool:

1. Modify the values of **START_RANGE**, **END_RANGE** and **KEY_TYPE** in the template yaml file. These values should be of subscriber ranges whose data you need to migrate from 4G UDR to 5G UDR.
2. Set the value of **K8S_HOST_IP** parameter as an external IP Address of the node, where you want to run this tool and update the node name in affinity rules.
3. Change the **UDR_SERVICE_BASEURL** with the Ingress Gateway URL using which 5G UDR is running.
4. Ensure that the value of **DB_SERVICE_NAME** is same as UDR. These details are available in the Installation Process of UDR.
5. Use client configuration details for **DIAMETER_REALM** and **DIAMETER_IDENTITY** parameters as configured in 4G UDR.
6. Enter the number of connections you want to establish from Diameter client to 4G UDR in the **DIAMETER_SETTING_NUM_OF_CONNECTIONS** parameter.

7. Enter the **DIAMETER_NODES_HOST**, **DIAMETER_NODES_PORT**, **REALM** and **IDENTITY** parameters values as 4G UDR server details.

8. Execute the following command to create a yaml file.

   ```
   kubectl create -f <template yaml >-n <namespace>
   ```

   where, **template yaml** is nudr_migration.yaml and **namespace** is the one that 5G UDR uses.

   **Example:** `kubectl create -f nudr_migration.yaml -n ocudr`

9. Execute the following command to check whether pod is in running state without any error:

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

   If the pod is up and running, it means the migration process has begun as a job and subscribers are getting processed.

5

# Using Unified Data Repository (UDR) Console

In this chapter, you will learn to login to CNC Console application and configure global and service parameters on UDR. The service configuration includes data repository service, notify service, NRF client service and diameter service. The CNC Console also allows you to perform provisioning operations for profile data, PCF data, SLF data, UDM data and schema management.

## Logging into CNC Console

A user can use UDR integrated with CNC Console only after logging successfully into the CNC Console application. To login successfully into the CNC Console, the user needs to make the following updates to the hosts file available at the **C:\Windows\System32\drivers\etc** location.

In the Windows system, user needs to open the **hosts** file in the notepad as an Administrator and append the following set of lines at the end:

**Example:**

```
10.75.212.88 cncc-iam-ingress-gateway.cncc.svc.cluster.local
```

```
10.75.212.88 cncc-core-ingress-gateway.cncc.svc.cluster.local
```

> **Note:**
>
> The IP Address in the above lines may change when deployment cluster changes.

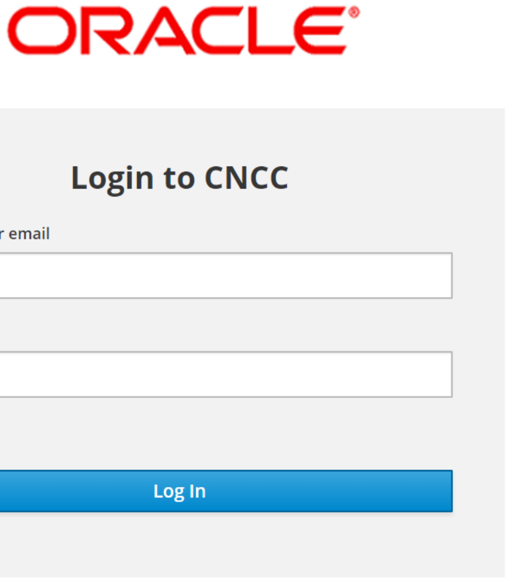
Save and close the notepad.

> **Note:**
>
> Before logging to CNC Console, it is important to create a CNC user. Using this user details, you can login to the CNC Console application. For information on creating a CNC Console user, you can refer to its user guide.

To login to CNC Console:

1. Type **http://cncc-core-ingress-gateway.cncc.svc.cluster.local:<PortNumber>/** in the web browser and press Enter. Following screen appears:

**Figure 5-1 CNC Console Login Screen**



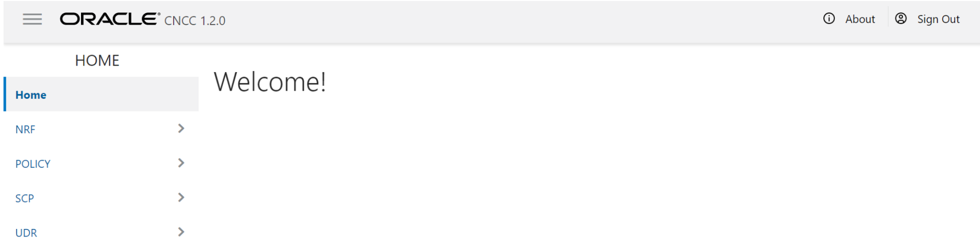
Login to CNCC

Username or email

Password

2. Enter the **Username** and **Password** and click **Log In**. Following screen appears:

**Figure 5-2 CNC Console Home Page**



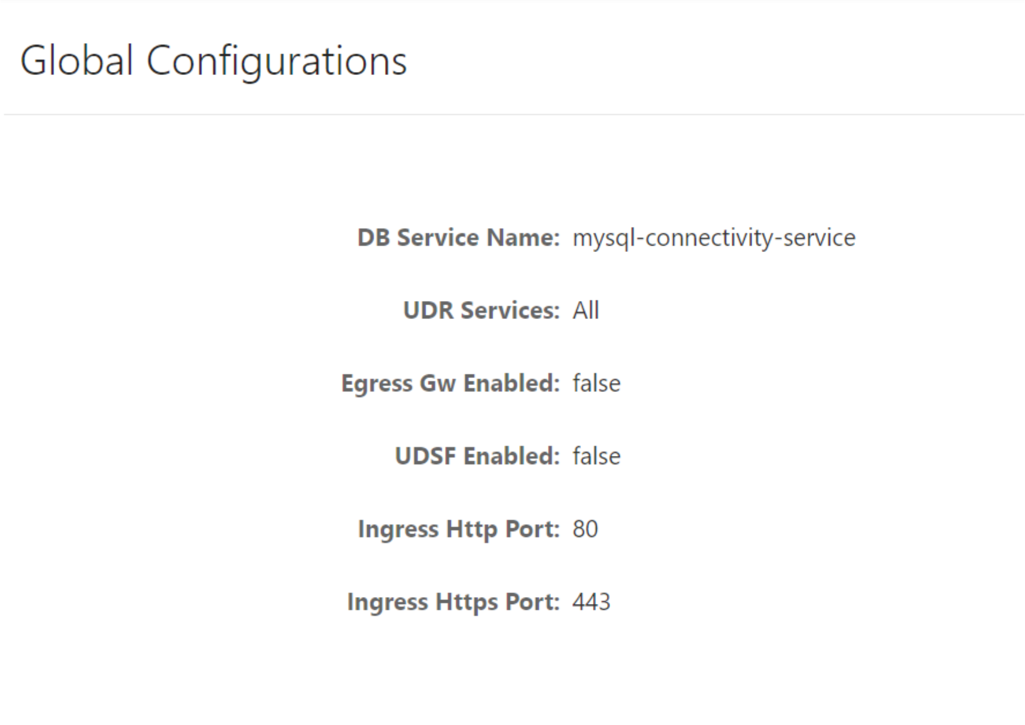
Welcome!

This is the CNC Console Home Page from where a user can navigate to different NF services. To use UDR services integrated with CNC Console, click **UDR** in the left navigation pane.

# Global Configurations

To access the **Global Configurations** screen, click **UDR** → **Global Configurations** in the left navigation pane of the CNC Console application. The following screen appears:

**Figure 5-3 Global Configurations**



Global Configurations

DB Service Name: mysql-connectivity-service

UDR Services: All

Egress Gw Enabled: false

UDSF Enabled: false

Ingress Http Port: 80

Ingress Https Port: 443

The Global Configurations screen displays value of some global parameters like:

- **DB Service Name:** displays the name of the DB Service
- **UDR Services:** displays 'All' when all the UDR services are engaged
- **Egress Gw Enabled:** displays 'true' when egress gateway is enabled and 'false' when egress gateway is not enabled.
- **UDSF Enabled:** displays 'true' when UDSF is enabled and 'false' when UDSF is not enabled.
- **Ingress Http Port:** is a read-only field and its default value is 80. It defines the Ingress Gateway signalling port for http.
- **Ingress Https Port:** is a read-only field and its default value is 443. It defines the Ingress Gateway signalling port for https.

These values are extracted from the database and on every Helm upgrade, users need to click the **Refresh** icon to view the latest values.

## Service Configurations

To configure UDR services using CNC Console, you can use Service Configurations. In the left navigation pane of the CNC Console application, click **UDR** → **Service Configurations**.

Using **Service Configurations**, UDR users can configure following services:

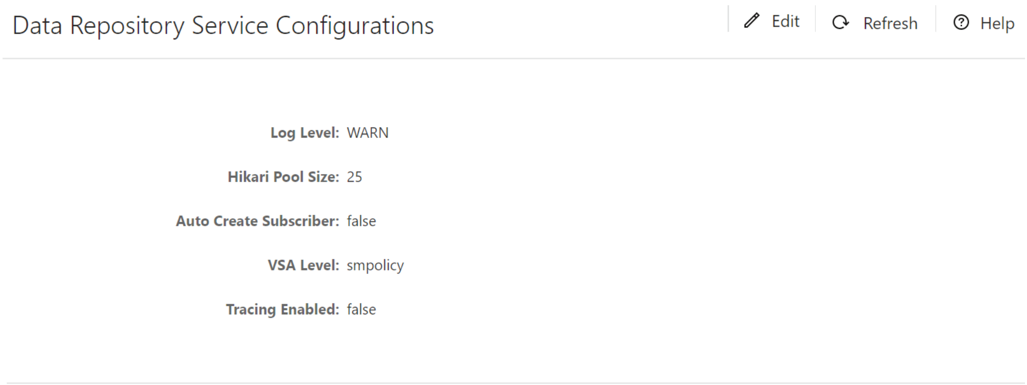
- Data Repository Service
- Notify Service
- NRF Client Service

- Diameter Service

## Data Repository Service

Click **UDR** → **Service Configurations** → **Data Repository Service**. The **Data Repository Service Configurations** screen appears as shown below:

**Figure 5-4 Data Repository Service Configurations**



Data Repository Service Configurations

Log Level: WARN

Hikari Pool Size: 25

Auto Create Subscriber: false

VSA Level: smpolicy

Tracing Enabled: false

The following table explains the Data Repository Service Configurations screen in detail:

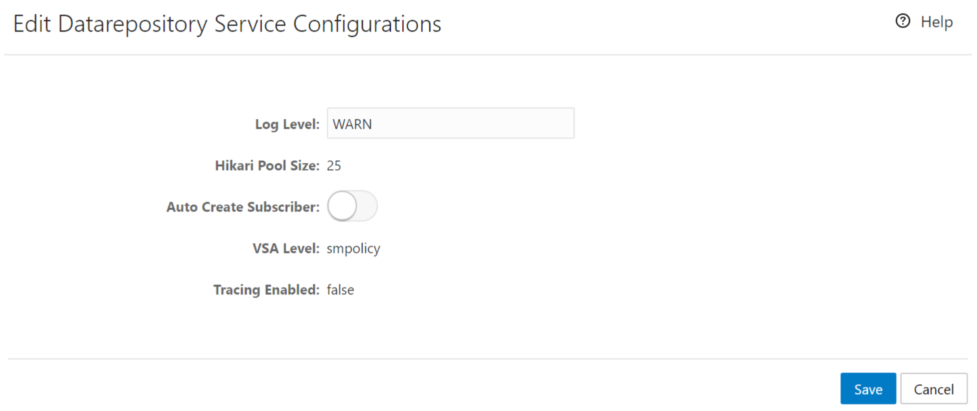
| Field Name | Default Value | Attribute | Description |
|---|---|---|---|
| Log Level | WARN | Editable | This field shows the log level of the data repository service. Its possible values are:<br>• WARN<br>• INFO<br>• DEBUG |
| Hikari Pool Size | 25 | Read-only | The hikari pool connection size to be created at start uP |
| Auto Create Subscriber | True | Editable | This field enables auto creation of a subscriber when creating data for a non existent subscriber. |
| VSA Level | smpolicy | Read-only | This field displays the VSA level. It is a read-only value. |
| Tracing Enabled | False | Read-only | By default, tracing is not enabled. |

### Editing Data Repository Service Configurations

To update or modify any configuration information related to data repository:

1. Click **Edit**. The **Edit Datarepository Service Configurations** screen appears.

**Figure 5-5 Edit Datarepository Service Configurations**



Edit Datarepository Service Configurations

Log Level: WARN

Hikari Pool Size: 25

Auto Create Subscriber:

VSA Level: smpolicy

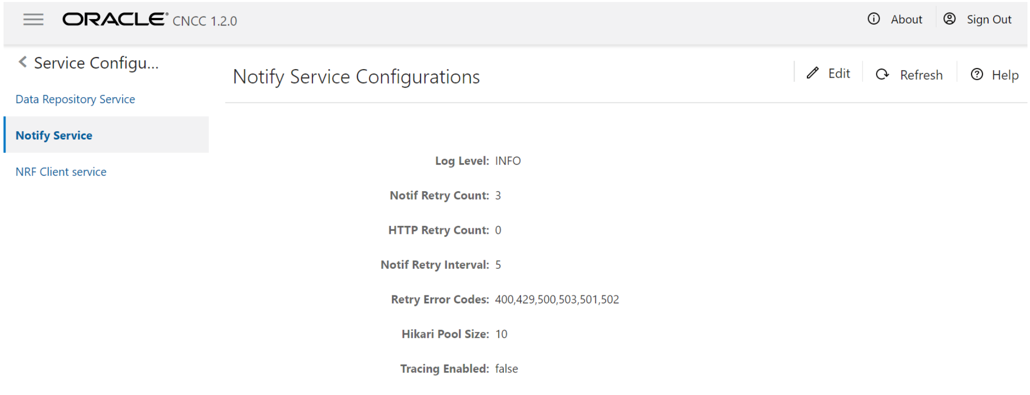
Tracing Enabled: false

2. Update the configuration details as required and click **Save**. A confirmation message, "Save successfully." appears.
3. If you do not want to modify any configuration, click **Cancel**. You navigate back to the Data Repository Service Configurations screen.
4. Click **Refresh** to reload the Data Repository Service Configurations screen.

## Notify Service

Click **UDR** → **Service Configurations** → **Notify Service**. The **Notify Service Configurations** screen appears as shown below:

**Figure 5-6 Notify Service Configurations**



Notify Service Configurations

Log Level: INFO

Notif Retry Count: 3

HTTP Retry Count: 0

Notif Retry Interval: 5

Retry Error Codes: 400,429,500,503,501,502

Hikari Pool Size: 10

Tracing Enabled: false

The following table explains the Notify Service Configurations screen in detail:

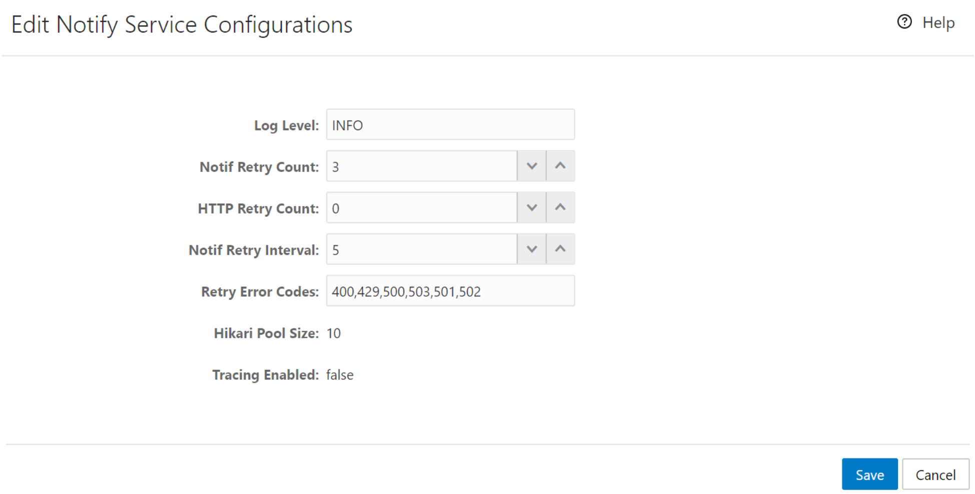
| Field Name | Default Value | Attribute | Description |
|---|---|---|---|
| Log Level | WARN | Editable | This field shows the log level of the data repository service. Its possible values are:<br>• WARN<br>• INFO<br>• DEBUG |
| Notif Retry Count | 3 | Editable | Number of notification attempts to be done in case of notification failures.<br>Retries are based on notification.retryerrorcodes configuration. |
| HTTP Retry Count | 2 | Editable | Number of retries when there is failure of connection to other services. |
| Notif Retry Interval | 5 | Editable | The retry interval for notifications in case of failure. It is measured in seconds.<br>Retries are based on notification.retryerrorcodes configuration. |
| Retry Error Codes | 400,429,500,503 | Editable | This field displays comma separated error codes. These error codes are eligible for retry notifications in case of failures. |
| Hikari Pool Size | 25 | Read-only | The hikari pool connection size to be created at start up. |
| Tracing Enabled | False | Read-only | By default, tracing is not enabled. |

### Editing Notify Service Configurations

To update or modify any configuration information related to notify service:

1. Click **Edit**. The **Edit Notify Service Configurations** screen appears.

**Figure 5-7 Edit Notify Service Configurations**



Edit Notify Service Configurations

Log Level: INFO

Notif Retry Count: 3

HTTP Retry Count: 0

Notif Retry Interval: 5

Retry Error Codes: 400,429,500,503,501,502

Hikari Pool Size: 10

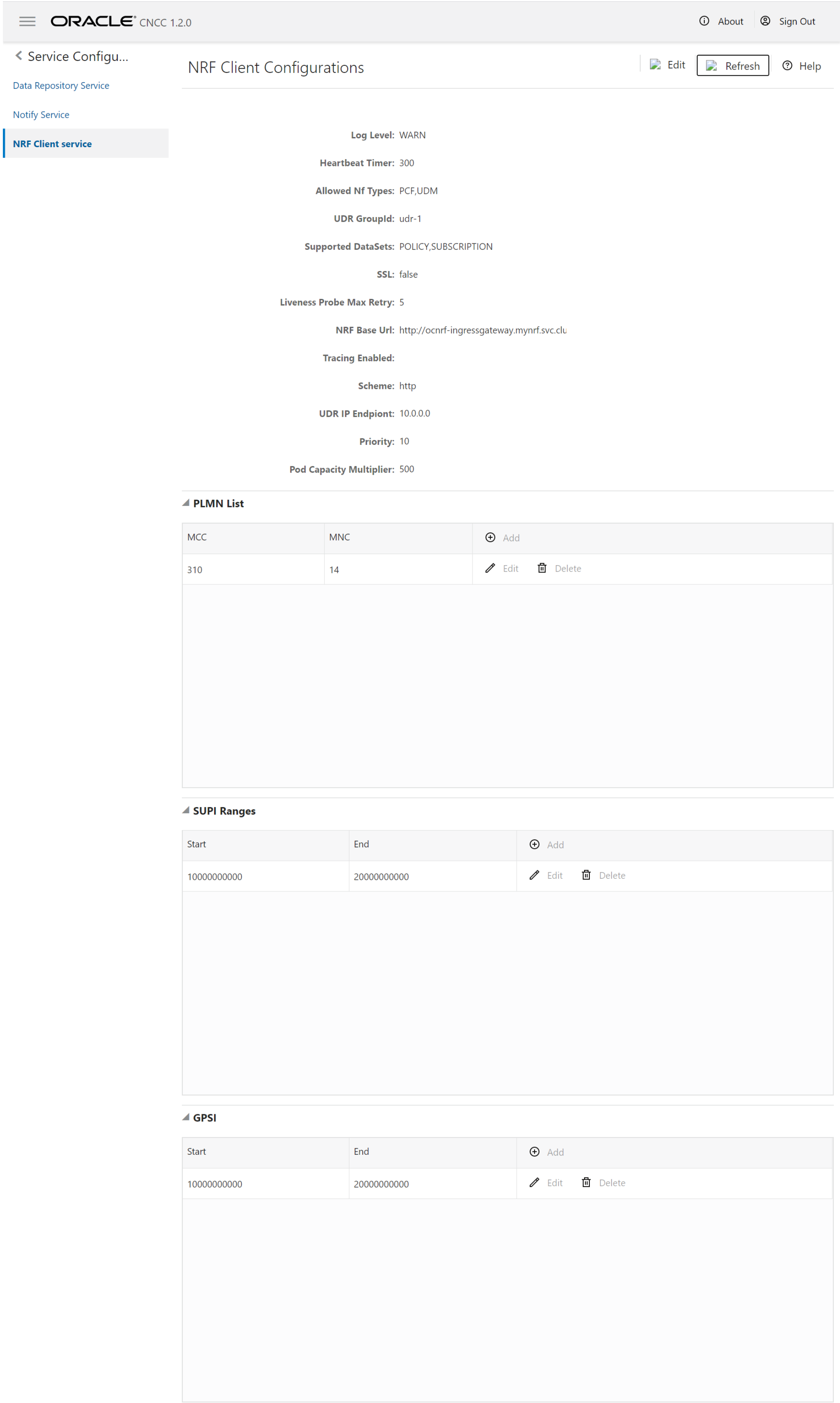
Tracing Enabled: false

2. Update the configuration details as required and click **Save**. A confirmation message, "Save successfully." appears.
3. If you do not want to modify any configuration, click **Cancel**. You are navigated back to the Notify Service Configurations screen.
4. Click **Refresh** to reload the Notify Service Configurations screen.

# NRF Client Service

Click **UDR** → **Service Configurations** → **NRF Client Service**. The **NRF Client Configurations** screen appears as shown below:

### Figure 5-8 NRF Client Configurations



Log Level: WARN

Heartbeat Timer: 300

Allowed Nf Types: PCF,UDM

UDR GroupId: udr-1

Supported DataSets: POLICY,SUBSCRIPTION

SSL: false

Liveness Probe Max Retry: 5

NRF Base Url: http://ocnrf-ingressgateway.mynrf.svc.clu

Tracing Enabled:

Scheme: http

UDR IP Endpiont: 10.0.0.0

Priority: 10

Pod Capacity Multiplier: 500

PLMN List

| MCC | MNC | Add |
|---|---|---|
| 310 | 14 | Edit Delete |

SUPI Ranges

| Start | End | Add |
|---|---|---|
| 10000000000 | 20000000000 | Edit Delete |

GPSI

| Start | End | Add |
|---|---|---|
| 10000000000 | 20000000000 | Edit Delete |

The following table explains the NRF Client Configurations screen in detail:

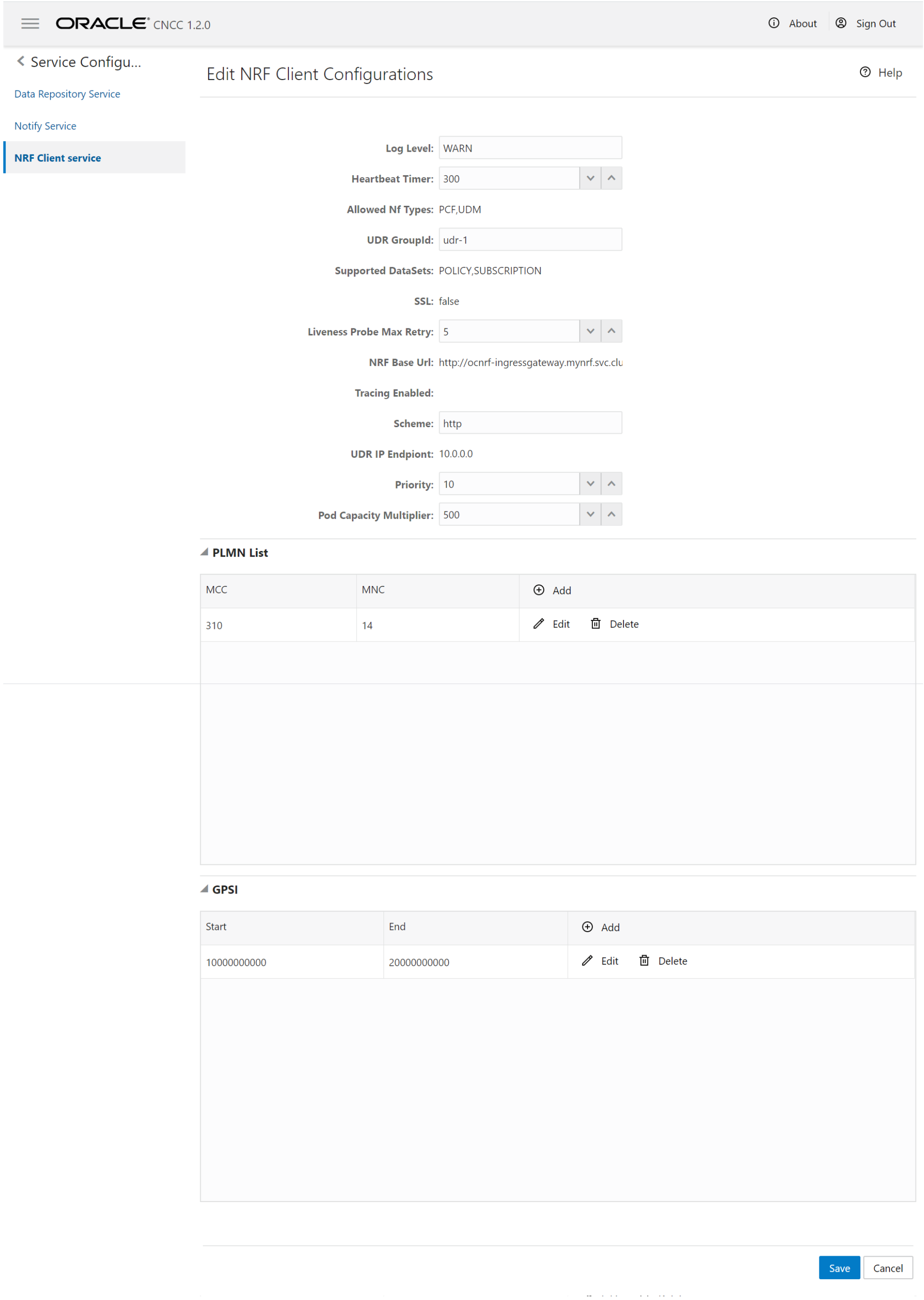
| Field Name | Default Value | Attribute | Description |
|---|---|---|---|
| Log Level | WARN | Editable | This field shows the log level of the data repository service. Its possible values are:<br>• WARN<br>• INFO<br>• DEBUG |
| Heartbeat Timer | 90 | Editable | |
| Allowed Nf Types | PCF,UDM | Read-only | The types of NF that are allowed to use this NRF client service. |
| UDR GroupId | udr-1 | Editable | Group ID of UDR |
| Supported DataSets | POLICY,SUBSCRIPTION | Read-only | The types of datasets that are allowed. |
| SSL | False | Read-only | SSL flag to enable SSL with udr nrf client pod |
| Liveness Probe Max Retry | 5 | Editable | The maximum number of times to retry liveness probe of NRF client service. |
| NRF Base Url | "http://ocnrf-ingressgateway.mynrf.svc.cluster.local/nnrf-nfm/v1/nf-instances" | Read-only | NRF URL for registration |
| Tracing Enabled | False | Read-only | By default, tracing is not enabled. |
| Scheme | Http | Read-only | scheme in which udr supports |
| UDR IP Endpoint | udr-1 | Editable | IP alotted to ocudr-endpoint pod |
| Priority | 10 | Editable | Priority to be sent in registration request |
| Pod Capacity Multiplier | 500 | Editable | Capacity multiplier of UDR based on number of UDR pods running |
| PLMN List | "[{mnc": "14", "mcc": "310"}]" | Editable | Plmn values range that it supports. Plmn values are sent to nrf during regisration from UDR. |
| SUPI Ranges | [{"start": "10000000000", "end": "20000000000"}]" | Editable | SUPI range supported with UDR |
| GPSI | "[{"start": "10000000000", "end": "20000000000"}]" | Editable | Gpsi Range supported with UDR |

### Editing NRF Client Configurations

To update or modify any configuration information related to NRF client:

1. Click **Edit**. The **Edit NRF Client Configurations** screen appears.

**Figure 5-9 Edit NRF Client Configurations**



Edit NRF Client Configurations

Log Level: WARN

Heartbeat Timer: 300

Allowed Nf Types: PCF,UDM

UDR GroupId: udr-1

Supported DataSets: POLICY,SUBSCRIPTION

SSL: false

Liveness Probe Max Retry: 5

NRF Base Url: http://ocnrf-ingressgateway.mynrf.svc.clu

Tracing Enabled:

Scheme: http

UDR IP Endpiont: 10.0.0.0

Priority: 10

Pod Capacity Multiplier: 500

PLMN List

| MCC | MNC | Add |
|---|---|---|
| 310 | 14 | Edit Delete |

GPSI

| Start | End | Add |
|---|---|---|
| 10000000000 | 20000000000 | Edit Delete |

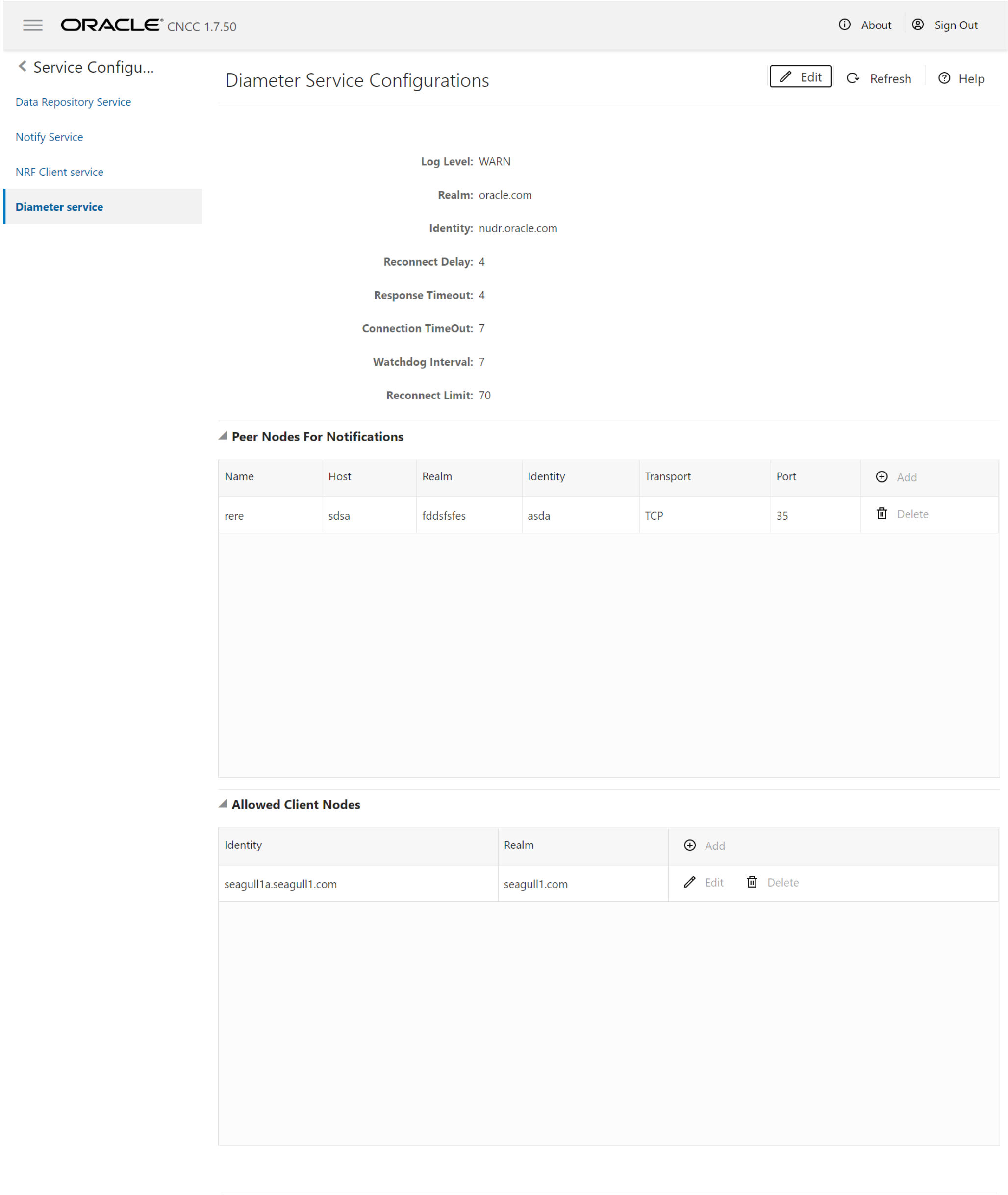
2. Update the configuration details as required and click **Save**. A confirmation message, "Save successfully." appears.

3. If you do not want to modify any configuration, click **Cancel**. You navigate back to the NRF Client Configurations screen.
4. Click **Refresh** to reload the NRF Client Configurations screen.

# Diameter Service

Click **UDR** → **Service Configurations** → **Diameter Service**. The **Diameter Service Configurations** screen appears as shown below:

### Figure 5-10 Diameter Service



Diameter Service Configurations

Log Level: WARN

Realm: oracle.com

Identity: nudr.oracle.com

Reconnect Delay: 4

Response Timeout: 4

Connection TimeOut: 7

Watchdog Interval: 7

Reconnect Limit: 70

Peer Nodes For Notifications

| Name | Host | Realm | Identity | Transport | Port |
|---|---|---|---|---|---|
| rere | sdsa | fddsfsfes | asda | TCP | 35 |

Allowed Client Nodes

| Identity | Realm |
|---|---|
| seagull1a.seagull1.com | seagull1.com |

The following table explains the Diameter Service Configurations screen in detail:

| Field Name | Default Value | Attribute | Description |
|---|---|---|---|
| Log Level | WARN | Editable | This field shows the log level of the data repository service. Its possible values are:<br>• WARN<br>• INFO<br>• DEBUG |
| Realm | oracle.com | Read-only | UDR Diameter server realm. |
| Identity | nudr.oracle.com | Read-only | UDR Diameter server identity/fqdn. |
| Reconnect Delay | 4 | Editable | Waiting time for reconnecting server peer in seconds. |
| Response Timeout | 4 | Editable | Time period to get the response from server peer in seconds. |
| Connection Timeout | 7 | Editable | Time period in which, both client and server should establish the connection. |
| Watchdog Interval | 7 | Editable | Interval of time to send Diameter watch dog message periodically. |
| Reconnect Limit | 70 | Read-only | Number of attempts allowed to reconnect with diameter server peer if server is unreachable/down. |
| Peer Nodes For Notifications | [ {"name": "seagull","responseOnly": true,"host": "10.75.185.158","realm": "seagull1.com","identity": "seagull1a.seagull1.com","port": 3868,"transport": "TCP"} ] | Editable | Server peer nodes used for sending notification. |
| peerNodesForNotifications.name | seagull | Editable | Name of the server peer |
| peerNodesForNotifications.responseOnly | true | Editable | Specify initiator or responder |
| peerNodesForNotifications.host | 10.75.185.158 | Editable | Host IP Address of the diameter server. |
| peerNodesForNotifications.realm | seagull1.com | Editable | Host server realm |
| peerNodesForNotifications.identity | seagull1a.seagull1.com | Editable | Host server identity or fqdn. |
| peerNodesForNotifications.port | 3868 | Editable | Host server port |

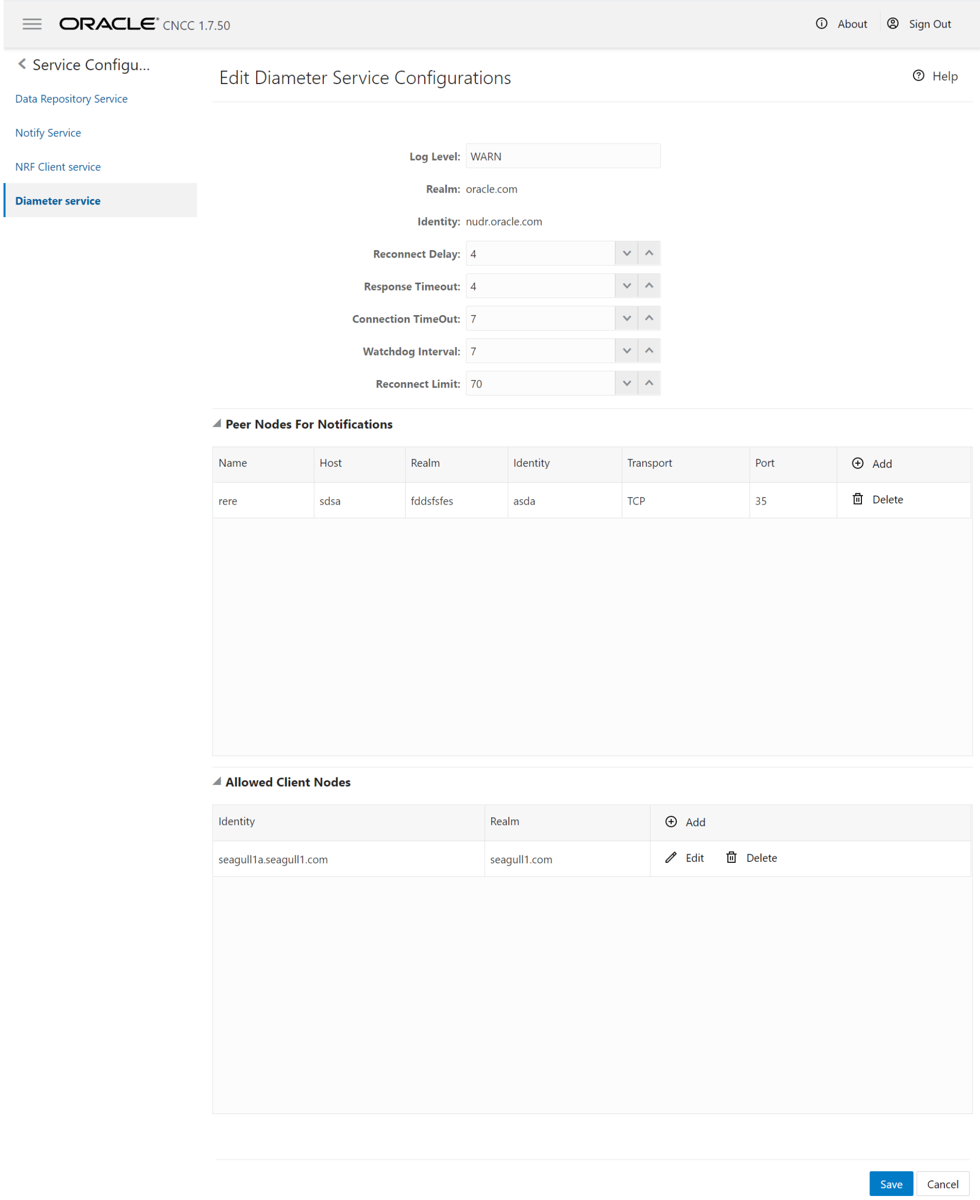
| Field Name | Default Value | Attribute | Description |
|---|---|---|---|
| peerNodesForNotifications.transport | TCP | Editable | TCP connection type |
| Allowed Client Nodes | [{"identity": "seagull1a.seagull1.com","realm": "seagull1.com"}] | Editable | Client peer who wants to connect 5G UDR through sh interface. |
| allowedClientNodes.identity | seagull1a.seagull1.com | Editable | Client peer identity/ fqdn who wants to connect 5G UDR through sh interface. |
| allowedClientNodes.realm | seagull1.com | Editable | Client peer realm who wants to connect 5G UDR through sh interface. |

### Editing Diameter Service Configurations

To update or modify any configuration information related to diameter service:

1. Click **Edit**. The **Edit Diameter Service Configurations** screen appears:

**Figure 5-11    Editing Diameter Service Configurations**



Edit Diameter Service Configurations

Log Level: WARN
Realm: oracle.com
Identity: nudr.oracle.com
Reconnect Delay: 4
Response Timeout: 4
Connection TimeOut: 7
Watchdog Interval: 7
Reconnect Limit: 70

Peer Nodes For Notifications

| Name | Host | Realm | Identity | Transport | Port | Add |
|---|---|---|---|---|---|---|
| rere | sdsa | fddsfsfes | asda | TCP | 35 | Delete |

Allowed Client Nodes

| Identity | Realm | Add |
|---|---|---|
| seagull1a.seagull1.com | seagull1.com | Edit Delete |

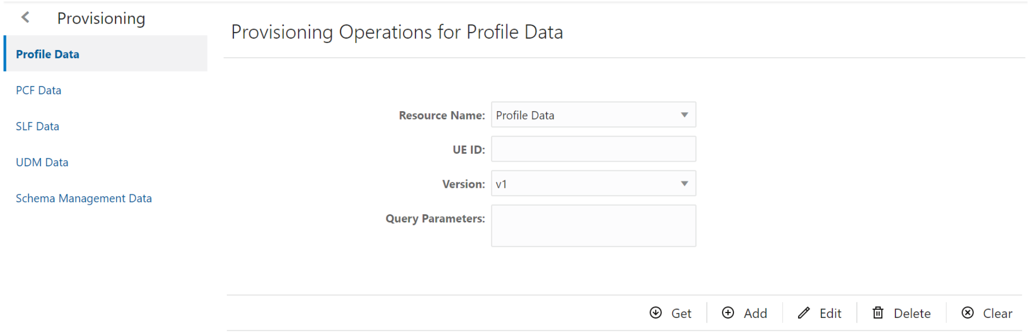
2. Update the configuration details as required and click **Save**. A confirmation message, "Save successfully." appears.
3. If you do not want to modify any configuration, click **Cancel**. You navigate back to the Diameter Service Configurations screen.
4. Click **Refresh** to reload the Diameter Service Configurations screen.

# Provisioning - Profile Data

In this section, you will learn to operate provisioning operations for Profile Data.

In the CNC Console application, click **UDR** → **Provisioning** → **Profile Data** in the left navigation pane. Following screen appears:

**Figure 5-12 Provisioning - Profile Data**



Resource Name: Profile Data
UE ID:
Version: v1
Query Parameters:

In the Provisioning Operations for Profile Data screen, user can:

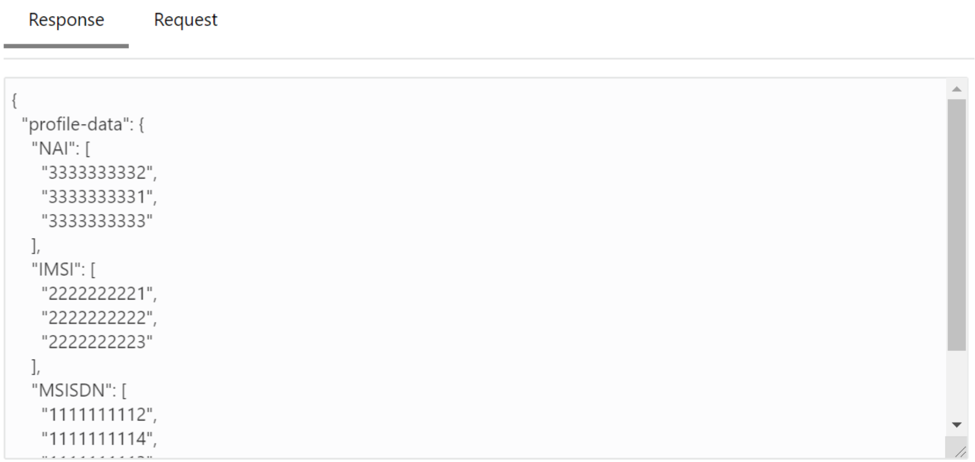
- Extract profile data from UDR database on the basis of UE ID entered
- Add new profile data to the UDR database
- Modify an existing profile data on the basis of UE ID
- Delete an existing profile data that is no more needed in the UDR database

### Extracting Profile Data

To extract a profile data:

1. Ensure **Resource Name** is set to 'Profile Data'.
2. Type an appropriate **UE ID**. For example: msisdn-1111111114
3. Click **Get**. A confirmation message, '**Fetched Successfully**' appears for a second and the result appears in the **Response** text area as follows:

   **Figure 5-13 Sample Screen: Get - Profile Data - Response**

```
{
 "profile-data": {
  "NAI": [
   "3333333332",
   "3333333331",
   "3333333333"
  ],
  "IMSI": [
   "2222222221",
   "2222222222",
   "2222222223"
  ],
  "MSISDN": [
   "1111111112",
   "1111111114",
```

4. An **Error message summary** appears in case of incorrect UE ID entered.

### Adding a Profile Data

To add a profile data:

1. Ensure **Resource Name** is set to 'Profile Data'.
2. Type an appropriate **UE ID**. For example: msisdn-1111111114
3. Click **Add**. The **Request** text area appears.
4. Type-in the profile data that you want to add to the UDR database.
5. Click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

### Editing a Profile Data

To edit a profile data:

1. Ensure **Resource Name** is set to 'Profile Data'.
2. Type an appropriate **UE ID**. For example: msisdn-1111111114
3. Click **Edit**. The **Request** text area displays an existing profile data.
4. Type-in the updated profile data.
5. Click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

### Deleting a Profile Data

To delete a profile data:

1. Ensure **Resource Name** is set to 'Profile Data'.
2. Type an appropriate **UE ID**. For example: msisdn-1111111114
3. Click **Delete**.
4. The '**Deleted Successfully**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

### Clearing a Profile Data

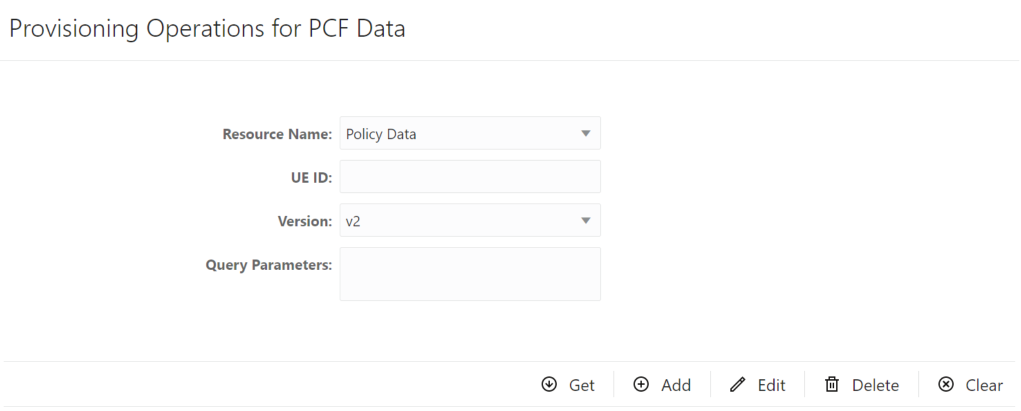
Click the **Clear** option to clear all the text fields and text areas of the Provisioning Operations for Profile Data.

# Provisioning - PCF Data

In this section, you will learn to operate provisioning operations for PCF (Policy Control Function) Data.

In the CNC Console application, click **UDR** → **Provisioning** → **PCF Data** in the left navigation pane. The following screen appears:

**Figure 5-14 Provisioning Operations for PCF Data**



Provisioning Operations for PCF Data

Resource Name: Policy Data

UE ID:

Version: v2

Query Parameters:

In the Provisioning Operations for PCF Data screen, user can:

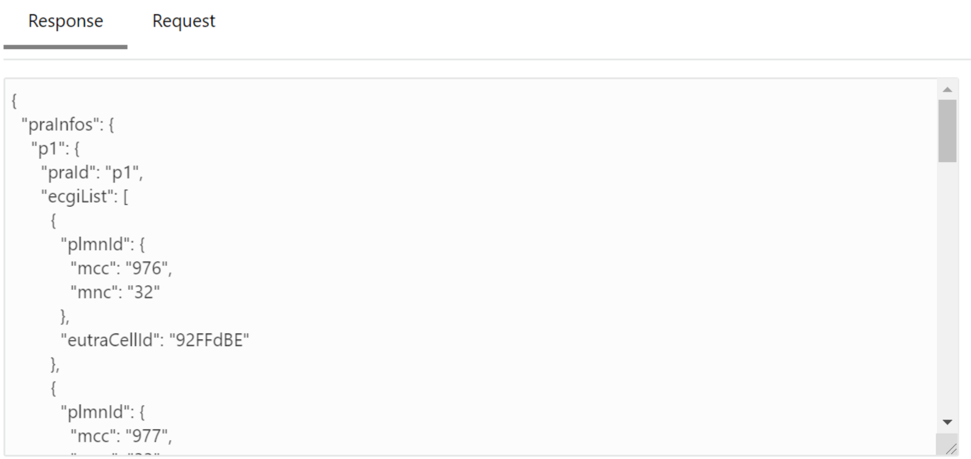
- Select **Resource Name** as Policy Data.
- Extract PCF data from UDR database on the basis of UE ID entered.
- Add PCF Data on the basis of payload.
- Modify the existing PCF data on the basis of UE ID.
- Delete the existing PCF data that is no more needed in the UDR database.

### Extracting PCF Data

To extract the PCF data:

1. Select **Resource Name** as Policy Data.
2. Type an appropriate **UE ID**. For example: msisdn-9111111112.
3. Click **Get**. A confirmation message, '**Fetched Successfully.**' appears and the result appears in the **Response** text area as follows:

**Figure 5-15 Provisioning PCF Get Response**



```
{
 "praInfos": {
  "p1": {
   "praId": "p1",
   "ecgiList": [
    {
     "plmnId": {
      "mcc": "976",
      "mnc": "32"
     },
     "eutraCellId": "92FFdBE"
    },
    {
     "plmnId": {
      "mcc": "977",
```

4. An **Error message summary** appears in case of incorrect UE ID entered.

### Adding a PCF Data

To add a PCF data:

1. Select the **Resource Name** as Policy Data.
2. Type an appropriate **UE ID**. For example: msisdn-9111111112.
3. Click **Add**. The **Request** text area displays an existing PCF data.
4. Type-in the updated profile data.
5. Click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

### Editing a PCF Data

To edit a PCF data:

1. Select the **Resource Name** as Policy Data.
2. Type an appropriate **UE ID**. For example: msisdn-9111111112.
3. Click **Edit**. The **Request** text area displays an existing PCF data.
4. Type-in the updated profile data.
5. Click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

### Deleting a PCF Data

To delete a PCF Data:

1. Select the **Resource Name** as either Access And Mobility Policy Data, Session Management Policy Data or UE Policy Set.
2. Type an appropriate **UE ID**. For example: msisdn-9111111112.
3. Click **Delete**.
4. The '**Deleted Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

### Clearing a Profile Data

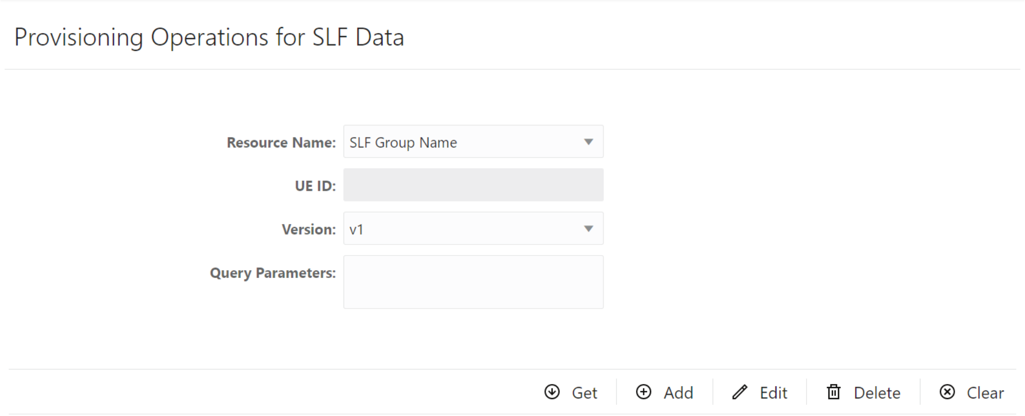
Click the **Clear** option to clear the text fields and text areas of the Provisioning Operations for PCF screen.

# Provisioning - SLF Data

In this section, you will learn to operate provisioning operations for SLF (Subscriber Location Function) Data.

In the CNC Console application, click **UDR** → **Provisioning** → **SLF Data** in the left navigation pane. The following screen appears:

**Figure 5-16 Provisioning SLF Data**



Provisioning Operations for SLF Data

Resource Name: SLF Group Name

UE ID:

Version: v1

Query Parameters:

> **Note:**
>
> SLF Group Name must be provisioned before SLF Subscriber is provisioned.

In the Provisioning Operations for SLF Data screen, user can:

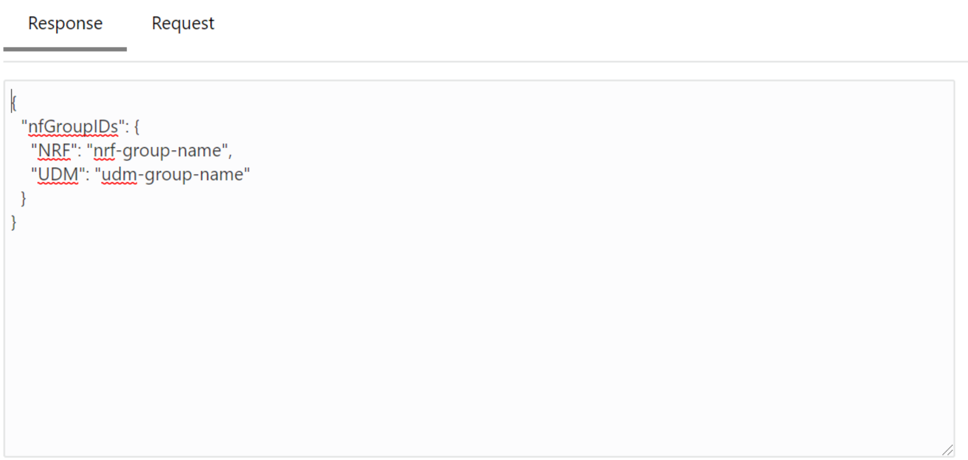
- Extract SLF data from UDR database on the basis SLF Group Name (Query Parameter) or SLF Subscriber (UEID).
- Add or edit an existing SLF data on basis of SLF Group Name (Query Parameter) or SLF Subscriber (UEID).
- Delete an existing SLF data that is no more needed in the UDR database.

### Extracting SLF Data on the basis of SLF Group Name

To extract the SLF data on the basis of SLF Group Name:

1. Select the **Resource Name** as SLF Group Name.
2. Enter the **Query Parameters**.
3. Click **Get**. A confirmation message, '**Fetched Successfully.**' appears and the result appears in the **Response** text area as follows:

**Figure 5-17 Provisioning SLF Get Response**



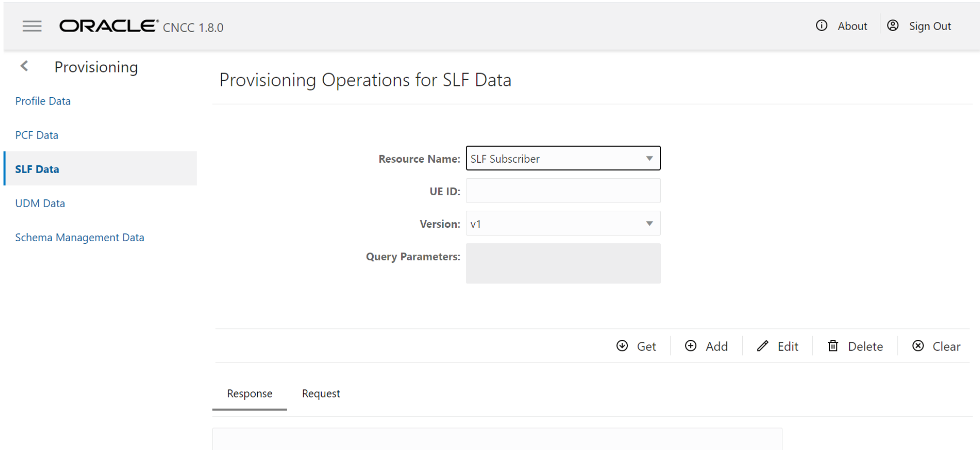
```
{
  "nfGroupIDs": {
    "NRF": "nrf-group-name",
    "UDM": "udm-group-name"
  }
}
```

### Extracting SLF Data on the basis of SLF Subscriber

To extract the SLF data on the basis of SLF Subscriber:

1. Select the **Resource Name** as SLF Subscriber. The following screen appears:

**Figure 5-18 SLF Data on the basis of SLF Subscriber**

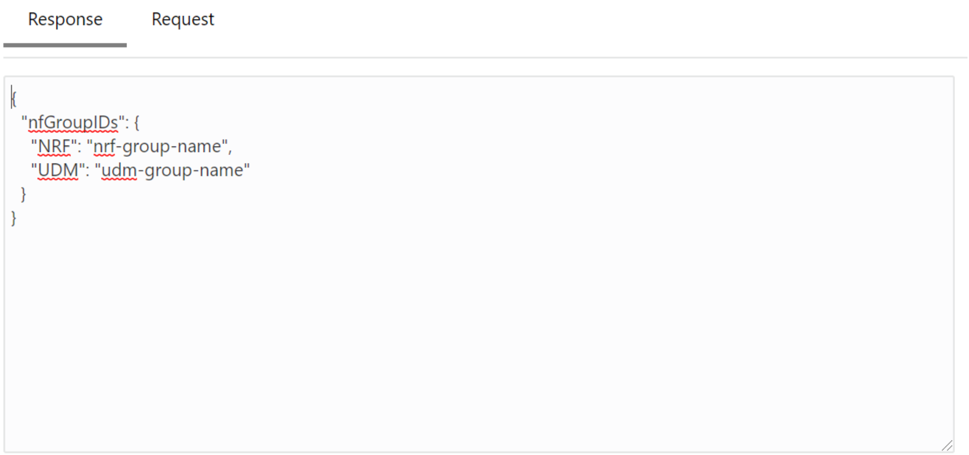


Provisioning Operations for SLF Data

Resource Name: SLF Subscriber
UE ID:
Version: v1
Query Parameters:

2. Type an appropriate **UE ID**. For example: msisdn-1111111112
3. Click **Get**. A confirmation message, '**Fetched Successfully.**' appears and the result appears in the **Response** text area as follows:

**Figure 5-19 Provisioning SLF Get Response**



```
{
  "nfGroupIDs": {
    "NRF": "nrf-group-name",
    "UDM": "udm-group-name"
  }
}
```

4. An **Error message summary** appears in case you enter an incorrect UE ID.

### Adding SLF Data on the basis of SLF Group Name or SLF Subscriber

To add the SLF data:

1. Select the **Resource Name** as either SLF Group Name or SLF Subscriber.
2. If you have selected the Resource Name as SLF Group Name then,
   a. Enter the **Query Parameters**.
   b. Click **Add**. The **Request** text area appears.
   c. Type the SLF data.
   d. Click **Submit**. The '**Saved Successfully.**' message appears.

   **[OR]**
3. If you have selected the Resource Name as SLF Subscriber then,
   a. Type an appropriate **UE ID**. For example: msisdn-1111111112.
   b. Click **Add**. The **Request** text area appears.
   c. Type-in the SLF data.
   d. Click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

### Editing SLF Data

To edit a SLF data:

1. Select the **Resource Name** as either SLF Group Name or SLF Subscriber.
2. If you have selected the Resource Name as SLF Group Name then,
   a. Type in the **Query Parameters** and click **Get**. The result appears in the **Response** text area.
   b. Click **Edit**. The **Request** text area displays an existing SLF data. Modify the relevant details.
   c. Click **Submit**. The '**Saved Successfully.**' message appears.

   **[OR]**
3. If you have selected the Resource Name as SLF Subscriber then,
   a. Type in the **UE ID** and click **Get**. The result appears in the **Response** text area.
   b. Click **Edit** and modify the required details. The **Request** text area displays an existing SLF data. Modify the relevant details.
   c. Click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

**Deleting a SLF Data**

To delete a SLF Data:

1. Select the **Resource Name** as either SLF Group Name or SLF Subscriber.
2. If you have selected the Resource Name as SLF Group Name then enter the 'Query Parameters' and if you have selected the Resource Name as SLF Subscriber then enter the 'UE ID'. **For example:** msisdn-1111111112
3. Click **Delete**.
4. The '**Deleted Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

**Clearing a Profile Data**

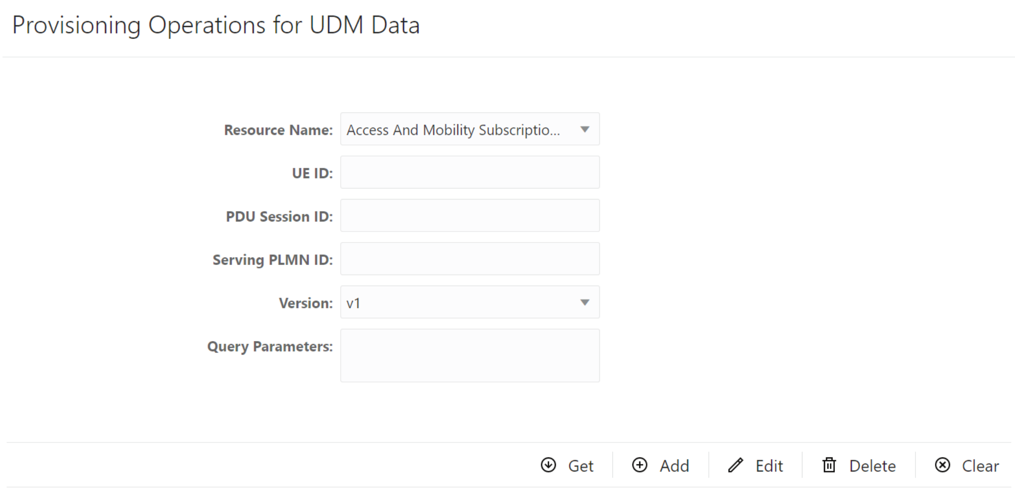
Click the **Clear** option to clear the text fields and text areas of the Provisioning Operations for SLF Data screen.

## Provisioning - UDM Data

In this section, you will learn to operate provisioning operations for UDM (Unified Data Management) Data.

In the CNC Console application, click **UDR** → **Provisioning** → **UDM Data** in the left navigation pane. The following screen appears:

**Figure 5-20 Provisioning Operations for UDM Data**



Provisioning Operations for UDM Data

Resource Name: Access And Mobility Subscriptio...

UE ID:

PDU Session ID:

Serving PLMN ID:

Version: v1

Query Parameters:

In the Provisioning Operations for UDM Data screen, user can extract, modify and delete UDM data on the basis of Resource Name, UE ID, PDU Session ID, Serving PLMN ID, Version and Query Parameters entered.

### Extracting UDM Data

To extract the UDM data:

1. Select the **Resource Name** from the drop-down list.
2. Type an appropriate **UE ID**, **PDU Session ID** and **Serving PLMN ID** in their respective fields.
3. Select an appropriate **Version** and enter **Query Parameters**.
4. Click **Get**. A confirmation message, '**Fetched Successfully.**' appears and the result appears in the **Response** text area.
5. An **Error message summary** appears in case of incorrect UE ID entered.

### Adding a UDM Data

To add a UDM data:

1. Select the **Resource Name**.
2. Type an appropriate **UE ID**, **PDU Session ID** and **Serving PLMN ID** in their respective fields.
3. Select an appropriate **Version** and enter **Query Parameters**.
4. Click **Edit**. The **Request** text area appears.
5. Enter details and click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

### Editing a UDM Data

To edit a UDM data:

1. Select the **Resource Name**.
2. Type an appropriate **UE ID**, **PDU Session ID** and **Serving PLMN ID** in their respective fields.
3. Select an appropriate **Version** and enter **Query Parameters**.
4. Click **Edit**. The **Request** text area displays an existing UDM data.
5. Edit as required and click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

### Deleting a UDM Data

To delete a UDM Data:

1. Enter the **Resource Name**.
2. Type an appropriate **UE ID**, **PDU Session ID** and **Serving PLMN ID** in their respective fields.
3. Select an appropriate **Version** and enter **Query Parameters**.
4. Click **Delete**.
5. The '**Deleted Successfully.**' message appears.

> **Note:**
>
> If a user enters any incorrect input, an Error message summary appears.

### Clearing a Profile Data

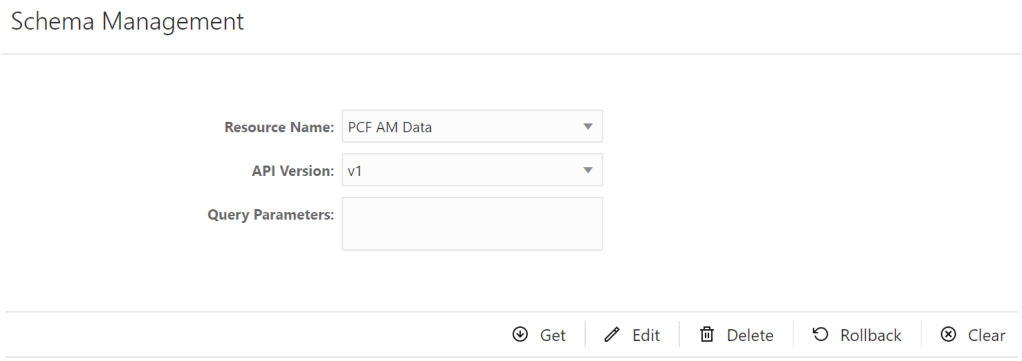
Click the **Clear** option to clear the text fields and text areas of the Provisioning Operations for UDM screen.

## Provisioning - Schema Management Data

In this section, you will learn to operate provisioning operations for schemas.

In the CNC Console application, click **UDR** → **Provisioning** → **Schema Management Data** in the left navigation pane. The following screen appears:

**Figure 5-21 Provisioning for Schema Management**



Schema Management

Resource Name: PCF AM Data

API Version: v1

Query Parameters:

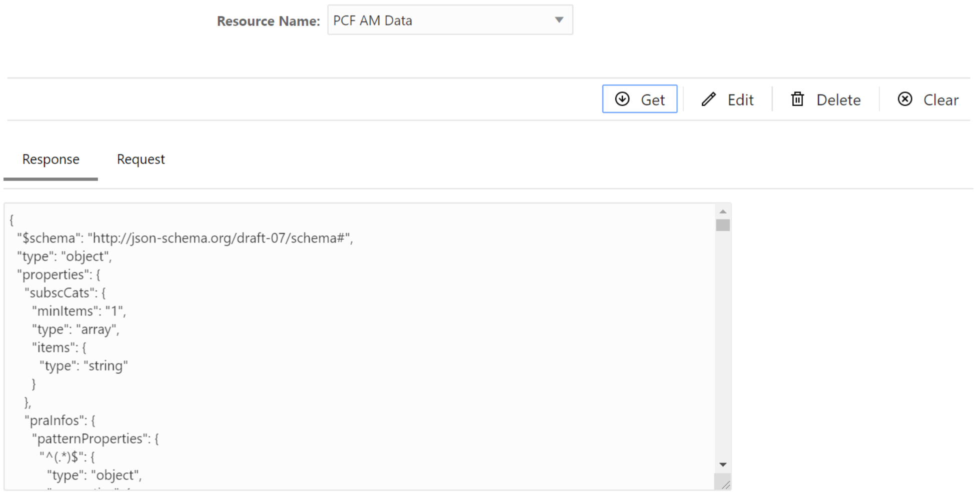
In the **Schema Management** screen, user can extract, modify and delete a schema on the basis of Resource Name and Query Parameters. This resource name can be either PCF AM Data, PCF SM Data or PCF UE Policy Set.

### Extracting a Schema

To extract a schema:

1. Select the **Resource Name**. It can be either PCF AM Data, PCF SM Data or PCF UE Policy Set.
2. Enter the **Query Parameters**.
3. Click **Get**. A confirmation message, '**Fetched Successfully.**' appears and the result appears in the **Response** text area as follows:

**Figure 5-22 Schema - Get - PCF AM Data**



Resource Name: PCF AM Data

```
{
 "$schema": "http://json-schema.org/draft-07/schema#",
 "type": "object",
 "properties": {
  "subscCats": {
   "minItems": "1",
   "type": "array",
   "items": {
    "type": "string"
   }
  },
  "praInfos": {
   "patternProperties": {
    "^(.*)$": {
     "type": "object",
```

4. An **Error message summary** appears in case of incorrect UE ID entered.

### Adding or Editing a Schema

To add or edit a schema:

1. Select the **Resource Name**.
2. Enter the **Query Parameters**.
3. Click **Edit**. The **Request** text area displays the selected schema.
4. Edit as required and click **Submit**. The '**Saved Successfully.**' message appears.

> **Note:**
>
> If a user edits the schema inappropriately, an Error message summary appears.

**Deleting a Schema**

To delete a schema:

1. Select the **Resource Name**.
2. Enter the **Query Parameters**.
3. Click **Delete**.
4. The '**Deleted Successfully.**' message appears.

> **Note:**
>
> If a user selects a schema that does not exists, an Error message summary appears.

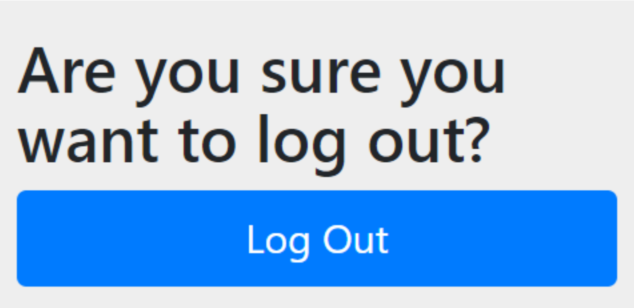
**Clearing a Profile Data**

Click the **Clear** option to clear the text fields and text areas of the Schema Management screen.

# Logout of CNC Console

To logout of the CNC Console application, click the **Sign Out** link available at the top right corner. The following confirmation box appears:

**Figure 5-23 CNC Console Logout**



Are you sure you want to log out?

Click **Log Out**. The user exits the application and the CNC Console login screen appears.

# 6 KPIs and Metrics

This section provides information about the KPIs and Metrics for Oracle Communications Unified Data Repository (OCUDR).

> **Note:**
>
> A sample Grafana json is available in the UDR Customer Documentation on Oracle Help Center at . You can download and import the file to your repository. http://docs.oracle.com/en/industries/communications/cloud-native-core/release21.html

### Metrics

Following metrics are applicable to **Ingress API Gateway**.

| Metric | Metric Details | Service Operation | Response Code | Notes |
|---|---|---|---|---|
| Total Ingress Requests | oc_ingressgateway_http_requests_total | All | Not Applicable | oc_ingressgateway_http_requests_total |
| Total Ingress Responses | oc_ingressgateway_http_responses_total | All | Not Applicable | oc_ingressgateway_http_responses_total |
| Post Requests with success response | Total no of POST requests received by UDR | PostRequests | 201 | oc_ingressgateway_http_requests_total{Method="POST"} |
| Put Requests with success response | Total no of PUT requests received by UDR | PutRequests | 201 | oc_ingressgateway_http_requests_total{Method="PUT"} |
| Get Requests with success response | Total no of GET requests received by UDR | GetRequests | 200 | oc_ingressgateway_http_requests_total{Method="GET"} |
| Delete Requests with success response | Total no of DELETE requests received by UDR | DeleteRequests | 204 | oc_ingressgateway_http_requests_total{Method="DELETE"} |
| Patch Requests with success response | Total no of PATCH requests received by UDR | PatchRequests | 204 | oc_ingressgateway_http_requests_total{Method="PATCH"} |

| Metric | Metric Details | Service Operation | Response Code | Notes |
|---|---|---|---|---|
| PostRequests failed with 2xx | Total no of success POST response with status as 2xx | PostRequests | 2xx | oc_ingressgateway_http_responses_total{Method="POST",Status=~"2.*"} |
| PutRequests failed with 2xx | Total no of success PUT response with status as 2xx | PutRequests | 2xx | oc_ingressgateway_http_responses_total{Method="PUT",Status=~"2.*"} |
| GetRequests failed with 2xx | Total no of success GET response with status as 2xx | GetRequests | 2xx | oc_ingressgateway_http_responses_total{Method="GET",Status=~"2.*"} |
| DeleteRequests failed with 2xx | Total no of success DELETE response with status as 2xx | DeleteRequests | 2xx | oc_ingressgateway_http_responses_total{Method="DELETE",Status=~"2.*"} |
| PatchRequests failed with 2xx | Total no of success PATCH response with status as 2xx | PatchRequests | 2xx | oc_ingressgateway_http_responses_total{Method="PATCH",Status=~"2.*"} |
| PostRequests failed with 4xx | Total no of failure POST response with status as 4xx | PostRequests | 4xx | oc_ingressgateway_http_responses_total{Method="POST",Status=~"4.*"} |
| PutRequests failed with 4xx | Total no of failure PUT response with status as 4xx | PutRequests | 4xx | oc_ingressgateway_http_responses_total{Method="PUT",Status=~"4.*"} |
| GetRequests failed with 4xx | Total no of failure GET response with status as 4xx | GetRequests | 4xx | oc_ingressgateway_http_responses_total{Method="GET",Status=~"4.*"} |
| DeleteRequests failed with 4xx | Total no of failure DELETE response with status as 4xx | DeleteRequests | 4xx | oc_ingressgateway_http_responses_total{Method="DELETE",Status=~"4.*"} |
| PatchRequests failed with 4xx | Total no of failure PATCH response with status as 4xx | PatchRequests | 4xx | oc_ingressgateway_http_responses_total{Method="PATCH",Status=~"4.*"} |

| Metric | Metric Details | Service Operation | Response Code | Notes |
|---|---|---|---|---|
| No of HTTP request received by UDR | Total no of HTTP requests received by UDR .Specific to method if we want we can add method as one more tag by comma separated | All | Not Applicable | oc_ingressgateway_http_requests_total{Scheme="HTTP"} |
| No of HTTPS request received by UDR | Total no of HTTPS requests received by UDR.Specific to method if we want we can add method as one more tag by comma separated | All | Not Applicable | oc_ingressgateway_http_requests_total{Scheme="HTTPS"} |

Following metrics are applicable to **Egress API Gateway**.

| Metric | Metric Details | UDR Microservice | Service Operation | Response Code | Notes |
|---|---|---|---|---|---|
| Total Egress Requests | oc_egressgateway_http_requests_total | ocudr-egress-gateway | All | Not Applicable | oc_egressgateway_http_requests_total |
| Total Egress Responses | oc_egressgateway_http_responses_total | ocudr-egress-gateway | All | Not Applicable | oc_egressgateway_http_responses_total |
| Number of Requests sent by UDR with POST towards other NF's | Total no of POST requests sent by UDR | ocudr-egress-gateway | PostRequests | 201 | oc_egressgateway_http_requests_total{Method="POST"} |
| Number of Requests sent by UDR with PUT towards other NF's | Total no of PUT requests sent by UDR | ocudr-egress-gateway | PutRequests | 201 | oc_egressgateway_http_requests_total{Method="PUT"} |
| Number of Requests sent by UDR with GET towards other NF's | Total no of GET requests sent by UDR | ocudr-egress-gateway | GetRequests | 200 | oc_egressgateway_http_requests_total{Method="GET"} |

| Metric | Metric Details | UDR Microservice | Service Operation | Response Code | Notes |
|---|---|---|---|---|---|
| Number of Requests sent by UDR with DELETE towards other NF's | Total no of DELETE requests sent by UDR | ocudr-egress-gateway | DeleteRequests | 204 | oc_egressgateway_http_requests_total{Method="DELETE"} |
| PostRequests with 2xx | Total no of success POST response with status as 2xx | ocudr-egress-gateway | PostRequests | 2xx | oc_egressgateway_http_responses_total{Method="POST",Status=~"2.*"} |
| PutRequests with 2xx | Total no of success PUT response with status as 2xx | ocudr-egress-gateway | PutRequests | 2xx | oc_egressgateway_http_responses_total{Method="PUT",Status=~"2.*"} |
| GetRequests with 2xx | Total no of success GET response with status as 2xx | ocudr-egress-gateway | GetRequests | 2xx | oc_egressgateway_http_responses_total{Method="GET",Status=~"2.*"} |
| DeleteRequests with 2xx | Total no of success DELETE response with status as 2xx | ocudr-egress-gateway | DeleteRequests | 2xx | oc_egressgateway_http_responses_total{Method="DELETE",Status=~"2.*"} |
| PostRequests with 4xx | Total no of failure POST response with status as 4xx | ocudr-egress-gateway | PostRequests | 4xx | oc_egressgateway_http_responses_total{Method="POST",Status=~"4.*"} |
| PutRequests with 4xx | Total no of failure PUT response with status as 4xx | ocudr-egress-gateway | PutRequests | 4xx | oc_egressgateway_http_responses_total{Method="PUT",Status=~"4.*"} |
| GetRequests with 4xx | Total no of failure GET response with status as 4xx | ocudr-egress-gateway | GetRequests | 4xx | oc_egressgateway_http_responses_total{Method="GET",Status=~"4.*"} |

| Metric | Metric Details | UDR Microservice | Service Operation | Response Code | Notes |
|---|---|---|---|---|---|
| DeleteRequests with 4xx | Total no of failure DELETE response with status as 4xx | ocudr-egress-gateway | DeleteRequests | 4xx | oc_egressgateway_http_responses_total{Method="DELETE",Status=~"4.*"} |
| No of HTTP request sent by UDR | Total no of HTTP requests sent by UDR.Specific to method if we want we can add method as one more tag by comma separated | ocudr-egress-gateway | All | Not Applicable | oc_egressgateway_http_requests_total{Scheme="HTTP"} |
| No of HTTPS request sent by UDR | Total no of HTTPS requests sent by UDR.Specific to method if we want we can add method as one more tag by comma separated | ocudr-egress-gateway | All | Not Applicable | oc_egressgateway_http_requests_total{Scheme="HTTPS"} |

Following metrics are applicable to the **nudr-dr service**.

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| Rest Controller UDR | Aggregate | UDRTotalRequests | udr_rest_allrequests_total | Total number of requests received by nudr-drservice (All UDR operations) |
| | | UDRTotalResponses | udr_rest_allresponse_total | Total number of responses sent by nudr-drservice (All UDR operations) |
| | | UDRTotalSuccessResponses | udr_rest_successResponse_total | Total number of success responses sent by nudr-drservice (All UDR operations) |
| | | UDRTotalFailureResponses | udr_rest_failureResponse_total | Total number of failure responses sent by nudr-drservice (All UDR operations) |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | PUT | UDRPutRequests | udr_rest_request_total{Method="PUT"} | Total number of PUT requests received by nudr-drservice (UDR related) |
| | | UDRPutSuccessResponses | udr_rest_response_success_total{Method="PUT"} | Total number of PUT success responses sent by nudr-drservice (UDR related) |
| | | UDRPutFailureResponses | udr_rest_response_failure_total{Method="PUT"} | Total number of PUT failure responses sent by nudr-drservice (UDR related) |
| | PATCH | UDRPatchRequests | udr_rest_request_total{Method="PATCH"} | Total number of PATCH requests received by nudr-drservice (UDR related) |
| | | UDRPatchSuccessResponses | udr_rest_response_success_total{Method="PATCH"} | Total number of PATCH success responses sent by nudr-drservice (UDR related) |
| | | UDRPatchFailureResponses | udr_rest_response_failure_total{Method="PATCH"} | Total number of PATCH failure responses sent by nudr-drservice (UDR related) |
| | GET | UDRGetRequests | udr_rest_request_total{Method="GET"} | Total number of GET requests received by nudr-drservice (UDR related) |
| | | UDRGetSuccessResponses | udr_rest_response_success_total{Method="GET"} | Total number of GET success responses sent by nudr-drservice (UDR related) |
| | | UDRGetFailureResponses | udr_rest_response_failure_total{Method="GET"} | Total number of GET failure responses sent by nudr-drservice (UDR related) |
| | DELETE | UDRDeleteRequests | udr_rest_request_total{Method="DELETE"} | Total number of DELETE requests received by nudr-drservice (UDR related) |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | UDRDeleteSuccessResponses | udr_rest_response_success_total{Method="DELETE"} | Total number of DELETE success responses sent by nudr-drservice (UDR related) |
| | | UDRDeleteFailureResponses | udr_rest_response_failure_total{Method="DELETE"} | Total number of DELETE failure responses sent by nudr-drservice (UDR related) |
| | POST | UDRPostRequests | udr_rest_request_total{Method="POST"} | Total number of GET requests received by nudr-drservice (UDR related) |
| | | UDRPostSuccessResponses | udr_rest_response_success_total{Method="POST"} | Total number of GET success responses sent by nudr-drservice (UDR related) |
| | | UDRPostFailureResponses | udr_rest_response_failure_total{Method="POST"} | Total number of GET failure responses sent by nudr-drservice (UDR related) |
| Schema Validation | - | UDRSchemaValidationSuccess | udr_schema_validation_success_total | Total success count of schema validations. Done for all operations with body. |
| | | UDRSchemaValidationFailure | udr_schema_validation_failure_total | Total failure count of schema validations. Done for all operations with body. |
| Schema Versioning | PUT | UDRPutSchemaSuccess | udr_schema_mgm_operations_success_total{Method="PUT"} | Total number of successful schema updates processed by nudr-drservice |
| | | UDRPutSchemaFailure | udr_schema_mgm_operations_failure_total{Method="PUT"} | Total number of failure schema updates processed by nudr-drservice |
| | GET | UDRGetSchemaSuccess | udr_schema_mgm_operations_success_total{Method="GET"} | Total number of successful schema GET operations processed by nudr-drservice |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | UDRGetSchemaFailure | udr_schema_mgm_operations_failure_total{Method="GET"} | Total number of failure schema GET operations processed by nudr-drservice |
| | DELETE | UDRDeleteSchemaSuccess | udr_schema_mgm_operations_success_total{Method="DELETE"} | Total number of successful schema deletion processed by nudr-drservice |
| | | UDRDeleteSchemaFailure | udr_schema_mgm_operations_failure_total{Method="DELETE"} | Total number of successful schema deletion processed by nudr-drservice |
| DB | Reads | DbReadSuccess | udr_db_operations_success_total{Method="READ"} | Total number of successful DB reads performed by nudr-drservice on the back end NDB Cluster |
| | | DbReadFailure | udr_db_operations_failure_total{Method="READ"} | Total number of failed DB reads performed by nudr-drservice on the back end NDB Cluster |
| | Writes/Updates | DbUpdateSuccess | udr_db_operations_success_total{Method="UPDATE"} | Total number of successful DB writes performed by nudr-drservice on the back end NDB Cluster |
| | | DbUpdateFailure | udr_db_operations_failure_total{Method="UPDATE"} | Total number of failed DB writes performed by nudr-drservice on the back end NDB Cluster |
| | Transaction | DbTotalTxnAttempted | udr_total_transactions_total | Total number of successful DB transactions (includes all DB operations) performed by nudr-drservice on the back end NDB Cluster |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | DbTxnSuccess | udr_successful_transactions_total | Total number of failed DB transactions (includes all DB operations) performed by nudr-drservice on the back end NDB Cluster |
| Rest Controller UDSF | Aggregate | UDSFTotalRequests | udsf_rest_allrequests_total | Total number of requests received by nudr-drservice (All UDSF operations) |
| | | UDSFTotalResponse | udsf_rest_allresponse_total | Total number of responses sent by nudr-drservice (All UDSF operations) |
| | | UDSFTotalSuccessResponses | udsf_rest_successResponse_total | Total number of success responses sent by nudr-drservice (All UDSF operations) |
| | | UDSFTotalFailureResponses | udsf_rest_failureResponse_total | Total number of failure responses sent by nudr-drservice (All UDSF operations) |
| | PUT | UDSFPutRequests | udsf_rest_request_total{Method="PUT"} | Total number of PUT requests received by nudr-drservice (UDSF related) |
| | | UDSFPutSuccessResponses | udsf_rest_response_success_total{Method="PUT"} | Total number of PUT success responses sent by nudr-drservice (UDSF related) |
| | | UDSFPutFailureResponses | udsf_rest_response_failure_total{Method="PUT"} | Total number of PUT failure responses sent by nudr-drservice (UDSF related) |
| | GET | UDSFGetRequests | udsf_rest_request_total{Method="GET"} | Total number of GET requests received by nudr-drservice (UDSF related) |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | UDSFGETSuccessResponses | udsf_rest_response_success_total{Method="GET"} | Total number of GET success responses sent by nudr-drservice (UDSF related) |
| | | UDSFGETFailureResponses | udsf_rest_response_failure_total{Method="GET"} | Total number of GET failure responses sent by nudr-drservice (UDSF related) |
| | DELETE | UDSFDeleteRequests | udsf_rest_request_total{Method="DELETE"} | Total number of DELETE requests received by nudr-drservice (UDSF related) |
| | | UDSFDeleteSuccessResponses | udsf_rest_response_success_total{Method="DELETE"} | Total number of DELETE success responses sent by nudr-drservice (UDSF related) |
| | | UDSFDeleteFailureResponses | udsf_rest_response_failure_total{Method="DELETE"} | Total number of DELETE failure responses sent by nudr-drservice (UDSF related) |
| Subscription | Aggregate | UDRSubscriptionRequests | nudr_dr_all_subscription_requests_total | Total number of subscription requests received by nudr-drservice (All operations) |
| | | UDRSubscriptionRequestSuccessful | nudr_dr_all_subscription_responses_total | Total number of successful subscription request processed successfully by nudr-drservice (All operations) |
| | | UDRSubscriptionResponseSuccessful | nudr_dr_all_subscription_successful_responses_total | Total number of successful subscription response sent by nudr-drservice (All operations) |
| | | UDRSubscriptionResponseFailure | nudr_dr_all_subscription_failed_responses_total | Total number of successful subscription response sent by nudr-drservice (All operations) |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | POST | UDRPostSubscriptionRequests | nudr_dr_subscription_request_total{Method="POST"} | Total number of POST Subscription Requests received by nudr-drservice |
| | | UDRPostSubscriptionResponseSuccess | nudr_dr_subscription_successful_response_total{Method="POST"} | Total number of successful POST Subscription response sent by nudr-drservice |
| | | UDRPostSubscriptionResponseFailure | nudr_dr_subscription_failed_response_total{Method="POST"} | Total number of failed POST Subscription response sent by nudr-drservice |
| | PUT | UDRPutSubscriptionRequests | nudr_dr_subscription_request_total{Method="PUT"} | Total number of PUT Subscription Requests received by nudr-drservice |
| | | UDRPutSubscriptionResponseSuccess | nudr_dr_subscription_successful_response_total{Method="PUT"} | Total number of successful PUT Subscription response sent by nudr-drservice |
| | | UDRPutSubscriptionResponseFailure | nudr_dr_subscription_failed_response_total{Method="PUT"} | Total number of failed PUT Subscription response sent by nudr-drservice |
| | GET | UDRGetSubscriptionRequests | nudr_dr_subscription_request_total{Method="GET"} | Total number of GET Subscription Requests received by nudr-drservice |
| | | UDRGetSubscriptionResponseSuccess | nudr_dr_subscription_successful_response_total{Method="GET"} | Total number of successful GET Subscription response sent by nudr-drservice |
| | | UDRGetSubscriptionResponseFailure | nudr_dr_subscription_failed_response_total{Method="GET"} | Total number of failed GET Subscription response sent by nudr-drservice |
| | DELETE | UDRDeleteSubscriptionRequests | nudr_dr_subscription_request_total{Method=”DELETE"} | Total number of DELETE Subscription Requests received by nudr-drservice |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | UDRDeleteSubscriptionResponseSuccess | nudr_dr_subscription_successful_response_total{Method="DELETE"} | Total number of successful DELETE Subscription response sent by nudr-drservice |
| | | UDRDeleteSubscriptionResponseFailure | nudr_dr_subscription_failed_response_total{Method="DELETE"} | Total number of failed DELETE Subscription response sent by nudr-drservice |
| Notification | Notifications to notification service | UDRNotificationRequests | nudr_dr_notif_signal_sent_total | Total notification sent successfully by nudr-drservice to nudr-notify-service |
| | | UDRNotificationFailure | nudr_dr_notif_signal_failed_notifications_total | Total notification failed to send to nudr-notify-service |
| | | UDRUpdateNotificationSuccess | nudr_dr_notif_signal_notification_update_success_total | Total update notifications sent successfully to nudr-notify-service |
| | | UDRUpdateNotificationFailure | nudr_dr_notif_signal_notification_update_failed_total | Total update notifications failed to send to nudr-notify-service |
| | | UDRDeleteNotificationSuccess | nudr_dr_notif_signal_notification_delete_success_total | Total delete notifications sent successfully to nudr-notify-service |
| | | UDRDeleteNotificationFailure | nudr_dr_notif_signal_notification_delete_failed_total | Total delete notifications failed to send to nudr-notify-service |
| NF TYPE METRICS | | UDR NF Type Specific Metrics | udr_nftype_specific_requests{NFType="PCF/UDM/SLF"} | Total number of requests that udr recevies from different nf types like ex: PCF,UDM,SLF |
| | | UDR NF Type Specific Metrics - Method wise | udr_nftype_specific_requests{NFType="PCF/UDM/SLF",Method="GET/POST/PUT/DELETE/PATCH"} | Total number of requests that udr receives from different nf types along with method types |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | UDR NF Type Specific Metrics - Provisioning Requests | udr_nftype_specific_requests{NFType="PCF/UDM/SLF",ProvRequest="Y/N"} | Total number of provisioning requests that will recevie udr specific to nf type |
| | | UDR NF Type Specific Metrics - Method wise and Provisioning Requests | udr_nftype_specific_requests{NFType="PCF/UDM/SLF",Method="GET/POST/PUT/DELETE/PATCH",ProvRequest="Y/N"} | Total number of requests specifc to each nf type,method type and provisiong requests |
| | | UDR NF Type Specific Metrics - Based on Subtype | udr_nftype_specific_requests{NFType="SLF" ,subtype="SLFGroupName/NFGroupIdMap"} | Total number of requests specific to SLF based on subtype |
| | | UDR NF Type Specific Metrics - Method Wise and Subtype | udr_nftype_specific_requests{NFType="SLF",Method="PUT",subtype="SLFGroupName/NFGroupIdMap"} | Total number of requests specific to SLF , method wise and subtype |
| | | UDR NF Type Specific Metrics - Subtype wise and Provisioning requests | udr_nftype_specific_requests{NFType="SLF",ProvRequest="Y" , subtype="SLFGroupName/NFGroupIdMap"} | Total number of requests specific to SLF, subtype wise and provisioning request |
| | | UDR NF Type Specific Metrics - Success Response - Subtype wise | udr_nftype_specific_requests_status{NFType="SLF",StatusCode=~"2.*", subtype="SLFGroupName/NFGroupIdMap"} | Total number of success responses for SLF based on subtype |
| | | UDR NF Type Specific Metrics - Success Response - Subtype wise and provisioning request | udr_nftype_specific_requests_status{NFType="SLF",StatusCode=~"2.*",ProvRequest="Y" , subtype="SLFGroupName/NFGroupIdMap"} | Total number of success responses for SLF based on subtype and provisioning request |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | UDR NF Type Specific Metrics - Success Response - Subtype wise,Method wise and provisioning request | udr_nftype_specific_requests_status{NFType="SLF",StatusCode=~"2.*",Method="GET/PUT/DELETE" , subtype="SLFGroupName/NFGroupIdMap"} | Total number of success responses for SLF based on subtype, method wise |
| | | UDR NF Type Specific Metrics - Success Response | udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"2.*"} | Total number of success responses for each nf type that udr sent |
| | | UDR NF Type Specific Metrics - Failure Response | udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"4.*"}+udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"5.*"} | Total number of failure responses for each nf type that udr sent |
| | | UDR NF Type Specific Metrics - Success Response for provisioning requests | udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"2.*",ProvRequest="Y"} | Total number of success responses of provisioing requests for each nf type |
| | | UDR NF Type Specific Metrics - Failure Response for provisioning requests | udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"4.*",ProvRequest="Y"}+udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"5.*",ProvRequest="Y"} | Total number of failure responses of provisioing requests for each nf type |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | UDR NF Type Specific Metrics - Success response method wise | udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"2.*",Method="GET/POST/PUT/PATCH/DELETE"} | Total number of success responses of each method type of nf type |
| | | UDR NF Type Specific Metrics - Failure response method wise | udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"4.*",Method="GET/POST/PUT/PATCH/DELETE"}+udr_nftype_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"5.*",Method="GET/POST/PUT/PATCH/DELETE"} | Total number of failure responses of each method type of nf type |
| NF TYPE - SUBSCRIPTION METRICS | | UDR NF Type Subscription Metrics | nudr_dr_nf_type_subscription_request{NFType="PCF/UDM"} | Total number of subscription requests for each nf type i.e PCF,UDM |
| | | UDR NF Type Subscription Metrics - Method Wise | nudr_dr_nf_type_subscription_request{NFType="PCF/UDM",Method="POST/PUT/DELETE/GET"} | Total number of subscription requests for each nf type as well as each method type |
| | | UDR NF Type Subscription Metrics - Success Response | nudr_dr_nf_type_subscription_request_status{NFType="PCF/UDM",StatusCode="2xx"} | Total number of success responses of subscriptions |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | UDR NF Type Subscription Metrics - Failure Response | nudr_dr_nf_type_subscription_request_status{NFType="PCF/UDM",StatusCode=~"4.*"}+<br>nudr_dr_nf_type_subscription_request_status{NFType="PCF/UDM",StatusCode=~"4.*"} | Total number of failure responses of subscriptions |
| | | UDR NF Type Subscription Metrics - Success Response method wise | nudr_dr_nf_type_subscription_request_status{NFType="PCF/UDM",StatusCode="2xx",Method="POST/PUT/DELETE/GET/PATCH"} | Total number of subscription success based on nf type,method type. |
| | | UDR NF Type Subscription Metrics - Failure Response Method wise | nudr_dr_nf_type_subscription_request_status{NFType="PCF/UDM",StatusCode=~"4.*",Method="POST/PUT/DELETE/GET/PATCH"}+nudr_dr_nf_type_subscription_request_status{NFType="PCF/UDM",StatusCode=~"4.*",Method="POST/PUT/DELETE/GET/PATCH"} | Total number of subscriptios failed based on nf type,method type. |
| Cnc console | | If the configuration change is successfully received | nudr_dr_config_service_update{StatusCode:200} | |
| Cnc console | | If the configuration change throws a error | nudr_dr_config_service_update{StatusCode:400} | |
| | | total number of vsa requests | udr_vsa_specific_requests_total {PUT,GET} | |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | total number of status request | udr_vsa_specific_requests_status_total {200, 201, 204, 400, 404, 409} | |

Following metrics are applicable to **nudr-notify-service**.

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| Notification | Notifications received on notify service | Notifyservice_UDRNotificationRequests | nudr_notif_notifications_signals_received_total | Total number of notifications received on nudr-notify-service from nudr-drservice |
| | | Notifyservice_UDRFailedNotificationsProcessing | nudr_notif_notifications_processing_failed_total | Total number of notification failure processing on nudr-notify-service |
| | Notifications to NF | Notifyservice_UDRNotificationsSent | nudr_notif_notifications_sent_total | Total number of notifications sent to network |
| | | Notifyservice_UDRSendNotificationFailures | nudr_notif_notifications_send_fail_total | Total number of notifications failed to send to network |
| | | Notifyservice_UDRNotificationsMarkedRetry | nudr_notif_notifications_marked_for_retry_total | Total number of notifications marked for retry after notification failure response |
| | | NotifyService_UDRNotificationResponses | nudr_notif_notifications_ack_total | Total number of responses for notifications sent to network |
| | | NotifyService_UDRNotificationSuccessResponses | nudr_notif_notifications_ack_2xx_total | Total number of success responses for notifications sent to network |
| | | NotifyService_UDRNotification4xxResponses | nudr_notif_notifications_ack_4xx_total | Total number of 4xx error responses for notifications sent to network |
| | | NotifyService_UDRNotification5xxResponses | nudr_notif_notifications_ack_5xx_total | Total number of 5xx error responses for notifications sent to network |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| NF TYPE - NOTIFICATION METRICS | | UDR NF Type Notification Metrics - Notifcations received to notify service from different nf type | nudr_notif_notifications_nftype_signals_received{NFType="UDM/PCF"} | Total number of notifications received to sent out to udm or pcf |
| | | UDR NF Type Notification Metrics - Notifcations received to notify service from nf types | nudr_notif_notifications_nftype_signals_received{NFType="UDM/PCF",Method="UPDATE/DELETE"} | Total number of notifications received to sent out to udm or pcf with update/delete requests. |
| CNC Console | | If the configuration change is received successfully | nudr_notify_config_service_update{StatusCode:200} | |
| CNC Console | | If the configuration change throws error while processing | nudr_notify_config_service_update{StatusCode:400} | |
| | | vsa notification received | nudr_notif_notifications_vsa_signals_received {UPDATE, DELETE} | |

Following metrics are applicable to **nudr-nrf-client** service.

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | Deregistration | UDRNRFDeRegistrationRequestsSent | udr_nrf_deregistration_requests_total | Total number of registration requests sent by nrf-client-service to NRF |
| | | UDRNRFDeRegistrationRequestsSuccessfull | udr_nrf_deregistration_success_total | Total number for successfully processed deregistration requests |
| Heartbeat towards NRF | Heartbeat | UDRNRFHeartBeatUpdateRequestsSent | udr_nrf_heartBeatUpdate_requests_total | Total number of heartbeat requests sent by nrf-client-service toward NRF to keep the status of UDR active |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | UDRNRFHeartBeatUpdateRequestsSuccessfull | udr_nrf_heartBeatUpdate_success_total | Total number of successfully processed heartbeat messages |
| | LivenessProbe | UDRNRFLivenessProbeFailure | udr_nrf_livenessProbe_failure_total | Total number of failure attempts of liveness probe check on the udr micro services before registration |
| UDR Registration with NRF | Registration | UDRNRFRegistrationRequestsSent | udr_nrf_registration_requests_total | Total number of registration requests sent by nrf-client-service to NRF |
| | | UDRNRFRegistrationRequestsSuccessfull | udr_nrf_registration_success_total | Total number for successfully processed registration requests |
| | Deregistration | UDRNRFDeRegistrationRequestsSent | udr_nrf_deregistration_requests_total | Total number of registration requests sent by nrf-client-service to NRF |
| | | UDRNRFDeRegistrationRequestsSuccessfull | udr_nrf_deregistration_success_total | Total number for successfully processed deregistration requests |
| Heartbeat towards NRF | Heartbeat | UDRNRFHeartBeatUpdateRequestsSent | udr_nrf_heartBeatUpdate_requests_total | Total number of heartbeat requests sent by nrf-client-service toward NRF to keep the status of UDR active |
| | | UDRNRFHeartBeatUpdateRequestsSuccessfull | udr_nrf_heartBeatUpdate_success_total | Total number of successfully processed heartbeat messages |
| | LivenessProbe | UDRNRFLivenessProbeFailure | udr_nrf_livenessProbe_failure_total | Total number of failure attempts of liveness probe check on the udr micro services before registration |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| CNC Console | | If the configuration update is received succesfully | udr_nrf_client_service_configUpdate{StatusCode:200} | |
| CNC Console | | If the configuration update throws error | udr_nrf_client_service_configUpdate{StatusCode:400} | |

Following metrics are applicable to **nudr-diameterproxy-service**.

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| SH Messages | | DIameterproxy_Request_Messages | nudr_diameterproxy_total_requests_total | The total number of requests received on the diameterproxy microservice. |
| | | Diameterproxy_Response_Messages | nudr_diameterproxy_total_response_total | Total number of Sh message responses from diameter peer |
| | | Diameterproxy_Success_Messages | nudr_diameterproxy_responses_total{ResultCode="2001"} | Total number of successful Sh message responses from diameter peer |
| | | Diameterproxy_Failure_Messages | nudr_diameterproxy_responses_total{ResultCode=!"2001"} | Total number of failure Sh message responses from diameter peer |
| | | Diameterproxy_Decode_Failed_Messages | nudr_diameterproxy_message_decoding_failed_total | The total number of messages failed while decoding and invalid AVPs in the message. |
| | | Diameterproxy_Discarded Messages | nudr_diameterproxy_request_discarded_total | The total number of discarded messages due to invalid command codes. |
| | USER DATA | Diameterproxy_UDR_Messages | nudr_diameterproxy_requests_total{MessageType="UDR"} | The total number of User Data Requests messages received on diameterproxy. |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | Diameterproxy_UDA_Success_Messages | nudr_diameterproxy_responses_total{MessageType="UDA", ResultCode="2001"} | The total number of success User Data Answer messages received on diameterproxy. |
| | | Diameterproxy_UDA_Failure_Messages | nudr_diameterproxy_responses_total{MessageType="UDA", ResultCode !="2001"} | The total number of failure User Data Answer messages received on diameterproxy. |
| | | Diameterproxy_UDR_Rejected_By_Unknown_User | nudr_diameterproxy_request_rejected_unknown_user_total{MessageType="UDR"} | The total number User data requests messages failed because of Unknown user. |
| | | Diameterproxy_UDR_Rejected_By_Invalid_Service_Indication | nudr_diameterproxy_rejected_invalid_service_indication_requests_total{MessageType="UDR"} | The total number User Data request messages failed because of invalid service passed. |
| | PROFILE UPDATE | Diameterproxy_PUR_Messages | nudr_diameterproxy_requests_total{MessageType="PUR"} | The total number of profile update request messages received on diameterproxy. |
| | | Diameterproxy_PUA_Success_Messages | nudr_diameterproxy_responses_total{MessageType="PUA", ResultCode="2001"} | The total number of success profile data update requests |
| | | Diameterproxy_PUA_Failure_Messages | nudr_diameterproxy_responses_total{MessageType="PUA", ResultCode !="2001"} | The total number of failure profile data update requests |
| | | Diameterproxy_PUR_Rejected_By_Unknown_User | nudr_diameterproxy_request_rejected_unknown_user_total{MessageType="PUR"} | The total number of PUR messages rejected by unknown user( Subscriber doesn't exists). |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | Diameterproxy_PUR_Rejected_By_Invalid_Service_Indication | nudr_diameterproxy_rejected_invalid_service_indication_requests_total{MessageType="PUR"} | The total number of failed profile update requests with passing invalid service indication. |
| | SUBSCRIPTION NOTIFICATION | Diameterproxy_SNR_Messages | nudr_diameterproxy_requests_total{MessageType="SNR"} | The total number of subscriber notification request messages received on diameterproxy. |
| | | Diameterproxy_SNA_Success_Messages | nudr_diameterproxy_responses_total{MessageType="SNA", ResultCode="2001"} | The total number of Success subscriber notification messages. |
| | | Diameterproxy_SNA_Failure_Messages | nudr_diameterproxy_responses_total{MessageType="SNA", ResultCode !="2001"} | The total number of failure subscriber notification messages |
| | | Diameterproxy_SNR_Rejected_By_Unknown_User | nudr_diameterproxy_request_rejected_unknown_user_total{MessageType="SNR"} | The total number of Subscriber notification messages rejected by an unknown user |
| | | Diameterproxy_SNR_Rejected_By_Invalid_Service_Indication | nudr_diameterproxy_rejected_invalid_service_indication_requests_total{MessageType="SNR"} | The total number of subscriber notification requests rejected by passing invalid service indications |
| | PUSH NOTIFICATION | Diameterproxy_PNR_Messages | udr_diameterproxy_requests_total{MessageType="PNR"} | The total number of Push notification messages sent from diameterproxy. |
| | | Diameterproxy_PNA_Success_Messages | nudr_diameterproxy_responses_total{MessageType="PNA", ResultCode="2001"} | The total number of success Push notification sent from diameterproxy. |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| | | Diameterproxy_PNA_Failure_Messages | nudr_diameterproxy_responses_total{MessageType="PNA", ResultCode !="2001"} | The total number of failure push notification sent from diameterproxy. |
| Rest Requests | | Diameterproxy_Total_Rest_Sent_To_Dr-service | nudr_diameterproxy_rest_total_req_msgs_total | The total number of rest requests sent from diameterproxy to dr-service |
| | | Diameterproxy_Total_Rest_success_responses | nudr_diameterproxy_rest_success_res_msgs_total | The total number of success responses from dr-service. |
| | | Diameterproxy_Total_Rest_Failure_responses | nudr_diameterproxy_rest_failure_res_msgs_total | The total number of failure responses from dr-service |
| | | Diameter_Total_Rest_request_recieved_From_Notify-service | nudr_diameterproxy_total_notification_recieved_total | The total number of rest requests received from notify-service. |
| | | Diameter_Total_Rest_Failure_requests_From_notify-service | nudr_diameterproxy_notification_failed_total | The total number of rest requests are failed to send Push notification from the diameterproxy service. |

Following metrics are applicable to **nudr-config service**.

| Category | Sub Category | Description | Metric Name |
|---|---|---|---|
| Schema validation of payloads | Response Code: 200 | Pegging schema validation with 200 as response code | nudr_config_schema_validation_total{StatusCode="200"} |
| Schema Validation of Payloads | Response Code:400 | Pegging schema validation with 400 as response code | nudr_config_schema_validation_total{StatusCode="400"} |
| Total Request | GET | Pegging all the GET Request received by config service | nudr_config_total_requests_total{Method='GET'} |
| Total Request | POST | Pegging all the POST Request received by config service | nudr_config_total_requests_total{Method='POST'} |
| Total Request | PUT | Pegging all the PUT Request received by config service | nudr_config_total_requests_total{Method='PUT'} |
| Response received | GET | Pegging the response of GET Request with response code : 200 | nudr_config_total_responses_total{Method='GET',StatusCode="200"} |

| Category | Sub Category | Description | Metric Name |
|---|---|---|---|
| Response received | GET | Pegging the response of GET Request with response code : 404 | nudr_config_total_responses_total{Method='GET',StatusCode="404"} |
| Response received | PUT | Pegging the response of PUT Request with response code : 200 | nudr_config_total_responses_total{Method='PUT',StatusCode="200"} |
| Response received | PUT | Pegging the response of PUT Request with response code : 400 | nudr_config_total_responses_total{Method='PUT',StatusCode="400"} |
| Response received | PUT | Pegging the response of PUT Request with response code : 500 | nudr_config_total_responses_total{Method='PUT',StatusCode="500"} |
| Response received | POST | Pegging the response of POST Request with response code : 201 | nudr_config_total_responses_total{Method='POST',StatusCode="201"} |
| Response received | POST | Pegging the response of POST Request with response code : 400 | nudr_config_total_responses_total{Method='POST',StatusCode="400"} |
| Response received | POST | Pegging the response of POST Request with response code : 500 | nudr_config_total_responses_total{Method='POST',StatusCode="500"} |

Following table provides metrics for **nudr-migration** micro service for OCUDR.

| Category | Description | Metric Name | Notes |
|---|---|---|---|
| Total Records Start/End Range | UDR Total number of records received from 4G | nudr_migration_total_requests_total | Total number of requests with start and end range |
| 4G UDR | UDR Total number of records received from 4G | nudr_migration_total_requests_read_total | Total number of requests read from 4G UDR |
| 5G UDR | UDR Total number of records processed to 5G | nudr_migration_total_requests_processed_total | Total number of requests processed to 5G UDR |
| 4G UDR | Total number of requests read from 4G UDR with success/failure | nudr_migration_total_requests_read_status_total{Status="success/failure"} | Total number of requests read from 4G UDR with success/failure |
| 5G UDR | Total number of requests processed to 5G UDR with success/failure | nudr_migration_total_requests_processed_status_total{Status="success/failure"} | Total number of requests processed to 5G UDR with success/failure |
| 4G UDR | Total number of subscribers not found while reading from 4G UDR | nudr_migration_total_no_of_source_subscriber_not_exist_total | Total number of subscribers not found while reading from 4G UDR |

| Category | Description | Metric Name | Notes |
|---|---|---|---|
| 5G UDR | Total number of target keys exists in 5G UDR | nudr_migration_total_no_of_target_udr_keys_exist | Total number of target keys exists in 5G UDR |

Following table provides metrics for **nudr-bulk-import** micro service for OCUDR.

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| Response | Success | Total number of success response from UDR for PCF | Nudr_bulk_import_PCF_total{StatusCode="204/201" ,Status="Success"} | Total number of response received from UDR for success for PCF |
| Response | Failure | Total number of failure response from UDR for PCF | Nudr_bulk_import_PCF_total{StatusCode="500/400/404/503" ,Status="Failure"} | Total number of failure response from UDR for PCF |
| Response | Success | Total number of success response from UDR for SLF | Nudr_bulk_import_SLF_total{StatusCode="204/201" ,Status="Success"} | Total number of success response from UDR for SLF |
| Response | Failure | Total number of failure response from UDR for SLF | Nudr_bulk_import_SLF_total{StatusCode="500/400/404/503" ,Status="Failure"} | Total number of failure response from UDR for SLF |
| Request | Success | Total number of records read from csv file. | nudr_bulk_import_csvfile_records_read_total{Method="DELETE/PUT/POST" , Status="Success"} | Total number of successful record reads from csv file for DELETE, PUT,POST operation . |
| Request | Failure | Total number of records read from csv file. | nudr_bulk_import_csvfile_records_read_total{Method="DELETE/PUT/POST" , Status="Failure"} | Total number of records in case of failure reads from the csv file for DELETE,PUT,POST operation. |
| Response | Success | Total number of requests processed by UDR | nudr_bulk_import_records_processed_total{Method = "POST/PUT/DELETE" , StatusCode="201/204", Status="Success"} | Total number of request successfully processed by UDR for POST,PUT,DELETE operation |

| Category | Sub-category | Description | Metric Name | Notes |
|---|---|---|---|---|
| Response | Success | Total number of requests failed at UDR | nudr_bulk_import_records_processed_total{Method = "POST/PUT/DELETE" , StatusCode="500/404/400/503", Status="Failure"} | Total number of request failed at UDR for POST,PUT,DELETE operation. |

Following table provides KPI details about the **ocudr-ingress-gateway** micro service for OCUDR.

| KPI Details | Service Operation | KPI | Response Code | Notes |
|---|---|---|---|---|
| No of Requests/sec | All | UDR Ingress Request Rate | Not Applicable | sum(irate(oc_ingressgateway_http_requests_total[5m])) |
| No of Responses/sec | All | UDR Ingress Response Rate | Not Applicable | sum(irate(oc_ingressgateway_http_responses_total[5m])) |
| No of Successful Responses of each type/No of Successful Responses per second | PostRequest | rate of Post Requests with success response | 201 | sum(irate(oc_ingressgateway_http_responses_total{Method="POST",Status=~"2.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total[5m])) |
| | PutRequests | rate of Put Requests with success response | 201 | sum(irate(oc_ingressgateway_http_responses_total{Method="PUT", Status=~"2.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total[5m])) |
| | GetRequests | rate of Get Requests with success response | 200 | sum(irate(oc_ingressgateway_http_responses_total{Method="GET", Status=~"2.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total[5m])) |

| KPI Details | Service Operation | KPI | Response Code | Notes |
|---|---|---|---|---|
| | DeleteRequests | rate of Delete Requests with success response | 204 | sum(irate(oc_ingressgateway_http_responses_total{Method="DELETE",Status=~"2.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total[5m])) |
| | PatchRequests | rate of Patch Requests with success response | 204 | sum(irate(oc_ingressgateway_http_responses_total{Method="PATCH",Status=~"2.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total[5m])) |
| No of 4xx responses of each type/No of 4xx responses per second | PostRequest | rate of PostRequests failed with 4xx | 4xx | sum(irate(oc_ingressgateway_http_responses_total{Method="POST",Status=~"4.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total{Status=~"4.*"}[5m])) |
| | PutRequests | rate of PutRequests failed with 4xx | 4xx | sum(irate(oc_ingressgateway_http_responses_total{Method="PUT", Status=~"4.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total{Status=~"4.*"}[5m])) |
| | GetRequests | rate of GetRequests failed with 4xx | 4xx | sum(irate(oc_ingressgateway_http_responses_total{Method="GET", Status=~"4.*"}[5m]))/ sum(irate(oc_ingressgateway_http_responses_total{Status=~"4.*"}[5m])) |

| KPI Details | Service Operation | KPI | Response Code | Notes |
|---|---|---|---|---|
|  | DeleteRequests | rate of DeleteRequests failed with 4xx | 4xx | sum(irate(oc_ingressgateway_http_responses_total{Method="DELETE",Status=~"4.*"}[5m]))/sum(irate(oc_ingressgateway_http_responses_total{Status=~"4.*"}[5m])) |
|  | PatchRequests | rate of PatchRequests failed with 4xx | 4xx | sum(irate(oc_ingressgateway_http_responses_total{Method="PATCH",Status=~"4.*"}[5m]))/sum(irate(oc_ingressgateway_http_responses_total{Status=~"4.*"}[5m])) |
| No of 5xx responses of each type/No of 5xx responses per second | PostRequest | rate of PostRequests failed with 5xx | 5xx | (sum(irate(oc_ingressgateway_http_requests_total{Method="POST"}[5m]))-sum(irate(oc_ingressgateway_http_responses_total{Method="POST"}[5m])))/(sum(irate(oc_ingressgateway_http_requests_total[5m]))-sum(irate(oc_ingressgateway_http_responses_total[5m]))) |

| KPI Details | Service Operation | KPI | Response Code | Notes |
|---|---|---|---|---|
| | PutRequests | rate of PutRequests failed with 5xx | 5xx | (sum(irate(oc_ingressgateway_http_requests_total{Method="PUT"}[5m]))-sum(irate(oc_ingressgateway_http_responses_total{Method="PUT"}[5m])))/(sum(irate(oc_ingressgateway_http_requests_total[5m]))-sum(irate(oc_ingressgateway_http_responses_total[5m]))) |
| | GetRequests | rate of GetRequests failed with 5xx | 5xx | (sum(irate(oc_ingressgateway_http_requests_total{Method="GET"}[5m]))-sum(irate(oc_ingressgateway_http_responses_total{Method="GET"}[5m])))/(sum(irate(oc_ingressgateway_http_requests_total[5m]))-sum(irate(oc_ingressgateway_http_responses_total[5m]))) |
| | DeleteRequests | rate of DeleteRequests failed with 5xx | 5xx | (sum(irate(oc_ingressgateway_http_requests_total{Method="DELETE"}[5m]))-sum(irate(oc_ingressgateway_http_responses_total{Method="DELETE"}[5m])))/(sum(irate(oc_ingressgateway_http_requests_total[5m]))-sum(irate(oc_ingressgateway_http_responses_total[5m]))) |

| KPI Details | Service Operation | KPI | Response Code | Notes |
|---|---|---|---|---|
| | PatchRequests | rate of PatchRequests failed with 5xx | 5xx | (sum(irate(oc_ingressgateway_http_requests_total{Method="PATCH"}[5m]))-sum(irate(oc_ingressgateway_http_responses_total{Method="PATCH"}[5m])))/(sum(irate(oc_ingressgateway_http_requests_total[5m]))-sum(irate(oc_ingressgateway_http_responses_total[5m]))) |
| Percentage of success requests of each type of operations | PostRequest | Percentage of Post Requests with success response | 201 | sum(oc_ingressgateway_http_responses_total{Method="POST",Status=~"2.*"})/sum(oc_ingressgateway_http_responses_total) |
| | PutRequests | Percentage of Put Requests with success response | 2xx | sum(oc_ingressgateway_http_responses_total{Method="PUT",Status=~"2.*"})/sum(oc_ingressgateway_http_responses_total) |
| | GetRequests | Percentage of Get Requests with success response | 200 | sum(oc_ingressgateway_http_responses_total{Method="GET",Status=~"2.*"})/sum(oc_ingressgateway_http_responses_total) |
| | DeleteRequests | Percentage of Delete Requests with success response | 204 | sum(oc_ingressgateway_http_responses_total{Method="DELETE",Status=~"2.*"})/sum(oc_ingressgateway_http_responses_total) |

| KPI Details | Service Operation | KPI | Response Code | Notes |
|---|---|---|---|---|
| | PatchRequests | Percentage of Patch Requests with success response | 204 | sum(oc_ingressgateway_http_responses_total{Method="PATCH",Status=~"2.*"})/sum(oc_ingressgateway_http_responses_total) |
| Percentage of 4xx requests of each type of operations | PostRequest | Percentage of PostRequests failed with 4xx | 4xx | sum(oc_ingressgateway_http_responses_total{Method="POST",Status=~"4.*"})/sum(oc_ingressgateway_http_responses_total{Status=~"4.*"}) |
| | PutRequests | Percentage of PutRequests failed with 4xx | 4xx | sum(oc_ingressgateway_http_responses_total{Method="PUT",Status=~"4.*"})/sum(oc_ingressgateway_http_responses_total{Status=~"4.*"}) |
| | GetRequests | Percentage of GetRequests failed with 4xx | 4xx | sum(oc_ingressgateway_http_responses_total{Method="GET",Status=~"4.*"})/sum(oc_ingressgateway_http_responses_total{Status=~"4.*"}) |
| | DeleteRequests | Percentage of DeleteRequests failed with 4xx | 4xx | sum(oc_ingressgateway_http_responses_total{Method="DELETE",Status=~"4.*"})/sum(oc_ingressgateway_http_responses_total{Status=~"4.*"}) |
| | PatchRequests | Percentage of PatchRequests failed with 4xx | 4xx | sum(oc_ingressgateway_http_responses_total{Method="PATCH",Status=~"4.*"})/sum(oc_ingressgateway_http_responses_total{Status=~"4.*"}) |

| KPI Details | Service Operation | KPI | Response Code | Notes |
|---|---|---|---|---|
| Percentage of 5xx requests of each type of operations | PostRequest | Percentage of PostRequests failed with 5xx | 5xx | (sum(oc_ingressgateway_http_requests_total{Method="POST"})-sum(oc_ingressgateway_http_responses_total{Method="POST"}))/(sum(oc_ingressgateway_http_requests_total)-sum(oc_ingressgateway_http_responses_total)) |
| | PutRequests | Percentage of PutRequests failed with 5xx | 5xx | (sum(oc_ingressgateway_http_requests_total{Method="PUT"})-sum(oc_ingressgateway_http_responses_total{Method="PUT"}))/(sum(oc_ingressgateway_http_requests_total)-sum(oc_ingressgateway_http_responses_total)) |
| | GetRequests | Percentage of GetRequests failed with 5xx | 5xx | (sum(oc_ingressgateway_http_requests_total{Method="GET"})-sum(oc_ingressgateway_http_responses_total{Method="GET"}))/(sum(oc_ingressgateway_http_requests_total)-sum(oc_ingressgateway_http_responses_total)) |
| | DeleteRequests | Percentage of DeleteRequests failed with 5xx | 5xx | (sum(oc_ingressgateway_http_requests_total{Method="DELETE"})-sum(oc_ingressgateway_http_responses_total{Method="DELETE"}))/(sum(oc_ingressgateway_http_requests_total)-sum(oc_ingressgateway_http_responses_total)) |

| KPI Details | Service Operation | KPI | Response Code | Notes |
|---|---|---|---|---|
| | PatchRequests | Percentage of PatchRequests failed with 5xx | 5xx | (sum(oc_ingressgateway_http_requests_total{Method="PATCH"})-sum(oc_ingressgateway_http_responses_total{Method="PATCH"}))/(sum(oc_ingressgateway_http_requests_total)-sum(oc_ingressgateway_http_responses_total)) |

Following table provides KPI details about the **ocudr-ingress-gateway** micro service for OCUDR.

| KPI Details | Service Operation | KPI | Response Code | Notes |
|---|---|---|---|---|
| No of Requests/sec | All | UDR Egress Request Rate | Not Applicable | sum(irate(oc_egressgateway_http_requests_total[5m])) |
| No of Responses/sec | All | UDR Egress Response Rate | Not Applicable | sum(irate(oc_egressgateway_http_requests_total[5m])) |
| No of Successful Responses of each type/No of Successful Responses per second | PostRequest | rate of Post/Get/Put/Delete Requests with success response.<br>NOTE : In metric name just replace with POST with any other method which you want to check(ex:GET,PUT..etc) | 201 | sum(irate(oc_egressgateway_http_responses_total{Method="POST",Status=~"2.*"}[5m]))/sum(irate(oc_egressgateway_http_responses_total[5m])) |

| KPI Details | Service Operation | KPI | Response Code | Notes |
|---|---|---|---|---|
| No of 4xx responses of each type/No of 4xx responses per second | PostRequest | rate of PostRequests failed with 4xx.<br>NOTE : In metric name just replace with POST with any other method which you want to check(ex:GET,PUT..etc) | 4xx | sum(irate(oc_egressgateway_http_responses_total{Method="POST",Status=~"4.*"}[5m]))/sum(irate(oc_egressgateway_http_responses_total{Status=~"4.*"}[5m])) |
| No of 5xx responses of each type/No of 5xx responses per second | PostRequest | rate of PostRequests failed with 5xx.<br>NOTE : In metric name just replace with POST with any other method which you want to check(ex:GET,PUT..etc) | 5xx | (sum(irate(oc_egressgateway_http_requests_total{Method="POST"}[5m]))-sum(irate(oc_egressgateway_http_responses_total{Method="POST"}[5m])))/(sum(irate(oc_egressgateway_http_requests_total[5m]))-sum(irate(oc_egressgateway_http_responses_total[5m]))) |
| Percentage of success requests of each type of operations | PostRequest | Percentage of Post Requests with success response.<br>NOTE : In metric name just replace with POST with any other method which you want to check(ex:GET,PUT..etc) | 201 | sum(oc_egressgateway_http_responses_total{Method="POST",Status=~"2.*"})/sum(oc_egressgateway_http_responses_total) |

# 7 Alert Details

The Alert details are as follows:

> **Note:**
>
> Max Ingress requests/sec in consideration is 1000/second.

| Alert | Severity | Alert Details |
|---|---|---|
| Alert if Ingress traffic reaches 95% of max TPS | Critical | Traffic Rate is above critical threshold |
| Alert if Ingress traffic reaches 90% of max TPS | Major | Traffic Rate is above major threshold |
| Alert if Ingress traffic reaches 80% of max TPS | Minor | Traffic Rate is above minor threshold |
| Alert if all error rate exceeds 0.1% of the total transactions | Warning | Transaction Error rate is above 0.1 Percent of Total Transactions |
| Alert if all error rate exceeds 1% of the total transactions | Warning | Transaction Error rate is above 1 Percent of Total Transactions |
| Alert if all error rate exceeds 10% of the total transactions | Minor | Transaction Error rate is above 10 Percent of Total Transactions |
| Alert if all error rate exceeds 25% of the total transactions | Major | Transaction Error rate is above 25 Percent of Total Transactions |
| Alert if all error rate exceeds 50% of the total transactions | Critical | Transaction Error rate is above 50 Percent of Total Transactions |
| Alert if authentication status error rate is 1% of all ingress authentication traffic | Warning | Authentication error rate is about 1% of authentication traffic in the given sliding window |
| Alert if authentication status error rate is 10% of all ingress authentication traffic | Minor | Authentication error rate is about 10% of authentication traffic in the given sliding window |
| Alert if authentication status error rate is 25% of all ingress authentication traffic | Major | Authentication error rate is about 25% of authentication traffic in the given sliding window |
| Alert if authentication status error rate is 50% of all ingress authentication traffic | Critical | Authentication error rate is about 50% of authentication traffic in the given sliding window |

| Alert | Severity | Alert Details |
|---|---|---|
| Alert if number of subscribers not found is 1% of all ingress traffic | Warning | Total number of response if subscriber not found is about 1% of ingress traffic in the reporting time frame |
| Alert if number of subscribers not found is 10% of all ingress traffic | Minor | Total number of response if subscriber not found is about 10% of ingress traffic in the reporting time frame |
| Alert if number of subscribers not found is 25% of all ingress traffic | Major | Total number of response if subscriber not found is about 25% of ingress traffic in the reporting time frame |
| Alert if number of subscribers not found is 50% of all ingress traffic | Critical | Total number of response if subscriber not found is about 50% of ingress traffic in the reporting time frame |

8

# Configuring Alerts

To configure Alerts:

> **Note:**
>
> In the below procedure, _NAME_ is the Helm Chart Release Name and _Namespace_ is the Prometheus NameSpace.

1. Execute the following command to take backup of current config map of Prometheus.

   ```
   kubectl get configmaps occne-prometheus-server -o yaml -n occne-infra
   > /tmp/tempConfig.yaml
   ```

2. Execute the following commands to add UDR alerts file to the Prometheus configmap yaml file.

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

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

3. Execute the following command to update the Prometheus configmap yaml file with updated UDR alert file name.

   ```
   kubectl replace configmap occne-prometheus-server -f /tmp/
   tempConfig.yaml
   ```

4. Execute the following command to add the UDR alerts rules in configmap under UDR alert file name.

   ```
   kubectl patch configmap occne-prometheus-server -n occne-infra --type
   merge --patch "$(cat ~/UdrAlertrules.yaml)"
   ```

   > **Note:**
   >
   > Prometheus server takes updated configmap reloaded after sometime automatically (approximately 20 sec).

# 9 Updating Alert Config Details

This section shares the content of the UDRAlertrules.yaml file.

> **Note:**
>
> The default namespace for UDR is OCUDR. You can change it as per deployment.

**language: sql**

```
apiVersion: v1
data:
  alertsudr: |
    groups:
    - name: OcudrAlerts
      rules:
      - alert: OcudrTrafficRateAboveMinorThreshold
        annotations:
          description: 'Ingress traffic Rate is above minor threshold 
i.e. 800
          requests per second (current value is: {{ $value }})'
          summary: 'Traffic Rate is above 80 Percent of Max requests 
per second(1000)'
        expr: 
sum(rate(oc_ingressgateway_http_requests_total{app_kubernetes_io_name=
        "ingressgateway",kubernetes_namespace="ocudr"}[20m])) >= 800 < 
900
        labels:
          severity: Minor
      - alert: OcudrTrafficRateAboveMajorThreshold
        annotations:
          description: 'Ingress traffic Rate is above major threshold 
i.e. 900 requests
          per second (current value is: {{ $value }})'
          summary: 'Traffic Rate is above 90 Percent of Max requests 
per second(1000)'
        expr: 
sum(rate(oc_ingressgateway_http_requests_total{app_kubernetes_io_name=
        "ingressgateway",kubernetes_namespace="ocudr"}[20m])) >= 900 < 
950
        labels:
          severity: Major
      - alert: OcudrTrafficRateAboveCriticalThreshold
        annotations:
          description: 'Ingress traffic Rate is above critical 
threshold i.e. 950 requests
```

```
          per second (current value is: {{ $value }})'
          summary: 'Traffic Rate is above 95 Percent of Max requests
per second(1000)'
        expr:
sum(rate(oc_ingressgateway_http_requests_total{app_kubernetes_io_name=
        "ingressgateway",kubernetes_namespace="ocudr"}[20m])) >= 950
        labels:
          severity: Critical
      - alert: OcudrTransactionErrorRateAbove0.1Percent
        annotations:
          description: 'Transaction Error rate is above 0.1 Percent of
Total Transactions
         (current value is {{ $value }})'
          summary: 'Transaction Error Rate detected above 0.1 Percent
of Total
          Transactions'
        expr: (sum(rate(oc_ingressgateway_http_responses_total{Status!
~"2.*",

app_kubernetes_io_name="ingressgateway",kubernetes_namespace="ocudr"}
[20m]) or
        (up * 0 ) ) )/sum(rate(oc_ingressgateway_http_responses_total

{app_kubernetes_io_name="ingressgateway",kubernetes_namespace="ocudr"}
[20m]))
         * 100 >= 0.1 < 1
        labels:
          severity: Warning
      - alert: OcudrTransactionErrorRateAbove1Percent
        annotations:
          description: 'Transaction Error rate is above 1 Percent of
Total Transactions
         (current value is {{ $value }})'
          summary: 'Transaction Error Rate detected above 1 Percent of
Total Transactions'
        expr: (sum(rate(oc_ingressgateway_http_responses_total{Status!
~"2.*",

app_kubernetes_io_name="ingressgateway",kubernetes_namespace="ocudr"}
[20m]) or
        (up * 0 ) ) )/sum(rate(oc_ingressgateway_http_responses_total

{app_kubernetes_io_name="ingressgateway",kubernetes_namespace="ocudr"}
[20m]))
         * 100 >= 1 < 10
        labels:
          severity: Warning
      - alert: OcudrTransactionErrorRateAbove10Percent
        annotations:
          description: 'Transaction Error rate is above 10 Percent of
Total Transactions
        (current value is {{ $value }})'
          summary: 'Transaction Error Rate detected above 10 Percent of
Total Transactions'
        expr: (sum(rate(oc_ingressgateway_http_responses_total{Status!
~"2.*",

app_kubernetes_io_name="ingressgateway",kubernetes_namespace="ocudr"}
[20m])
        or (up * 0 ) ) )/sum(rate(oc_ingressgateway_http_responses_total

{app_kubernetes_io_name="ingressgateway",kubernetes_namespace="ocudr"}
[20m]))
        * 100 >= 10 < 25
        labels:
          severity: Minor
      - alert: OcudrTransactionErrorRateAbove25Percent
        annotations:
          description: 'Transaction Error Rate detected above 25
Percent of
           Total Transactions (current value is {{ $value }})'
          summary: 'Transaction Error Rate detected above 25 Percent of
Total Transactions'
        expr: (sum(rate(oc_ingressgateway_http_responses_total
        {Status!~"2.*",app_kubernetes_io_name="ingressgateway",
        kubernetes_namespace="ocudr"}[20m]) or (up * 0 ) ) )/sum
        (rate(oc_ingressgateway_http_responses_total

{app_kubernetes_io_name="ingressgateway",kubernetes_namespace="ocudr"}
[20m]))
         * 100 >= 25 < 50
        labels:
          severity: Major
      - alert: OcudrTransactionErrorRateAbove50Percent
        annotations:
          description: 'Transaction Error Rate detected above 50
Percent of
        Total Transactions (current value is {{ $value }})'
          summary: 'Transaction Error Rate detected above 50 Percent of
Total Transactions'
        expr: (sum(rate(oc_ingressgateway_http_responses_total{Status!
~"2.*",

app_kubernetes_io_name="ingressgateway",kubernetes_namespace="ocudr"}
[20m])
        or (up * 0 ) ) )/sum(rate(oc_ingressgateway_http_responses_total

{app_kubernetes_io_name="ingressgateway",kubernetes_namespace="ocudr"}
[20m]))
         * 100 >= 50
        labels:
          severity: Critical
      - alert: OcudrSubscriberNotFoundAbove1Percent
        annotations:
          description: 'Total number of response if subscriber not
found is about 1% of
        ingress traffic'
          summary: 'Total number of response if subscriber not found is
about 1% of
        ingress traffic'
        expr: (sum(rate(udr_subscriber_not_found_total[10m]))/

sum(rate(oc_ingressgateway_http_requests_total{kubernetes_namespace="ocu
dr"}
        [10m])))*100 >= 1 < 10
        labels:
          severity: Warning
      - alert: OcudrSubscriberNotFoundAbove10Percent
        annotations:
          description: 'Total number of response if subscriber not
found is about 10% of
        ingress traffic'
          summary: 'Total number of response if subscriber not found is
about 10% of
        ingress traffic'
        expr: (sum(rate(udr_subscriber_not_found_total[10m]))/sum(rate

(oc_ingressgateway_http_requests_total{kubernetes_namespace="ocudr"}
[10m])))*100 >=
         10 < 25
        labels:
          severity: Minor
      - alert: OcudrSubscriberNotFoundAbove25Percent
        annotations:
          description: 'Total number of response if subscriber not
found is about 25% of
        ingress traffic'
          summary: 'Total number of response if subscriber not found is
about 25% of
        ingress traffic'
        expr: (sum(rate(udr_subscriber_not_found_total[10m]))/sum(rate

(oc_ingressgateway_http_requests_total{kubernetes_namespace="ocudr"}
[10m])))*100 >=
        25 < 50
        labels:
          severity: Major
      - alert: OcudrSubscriberNotFoundAbove50Percent
        annotations:
          description: 'Total number of response if subscriber not
found is about 50% of
        ingress traffic'
          summary: 'Total number of response if subscriber not found is
about 50% of
        ingress traffic'
        expr: (sum(rate(udr_subscriber_not_found_total[10m]))/sum(rate

(oc_ingressgateway_http_requests_total{kubernetes_namespace="ocudr"}
[10m])))*100 >= 50
        labels:
          severity: Critical
```

# A
# ASM Specific Configuration

To configure ASM, you have to:

- Add the following annotation under **Global section** of UDR deployment.

```
 # ********  Sub-Section Start: Custom Extension Global Parameters
********

#*******************************************************************
*******
global:
  customExtension:
    allResources:
      labels: {}
      annotations:
        sidecar.istio.io/inject: "false"

    lbServices:
      labels: {}
      annotations: {}

    lbDeployments:
      labels: {}
      annotations:
        sidecar.istio.io/inject: "true"
        oracle.com/cnc: "true"
    nonlbServices:
      labels: {}
      annotations: {}

    nonlbDeployments:
      labels: {}
      annotations:
        sidecar.istio.io/inject: "true"
        oracle.com/cnc: "true"

  # ********  Sub-Section End: Custiom Extensions Global Parameters
********

#*******************************************************************
*******
```

- Enable Service Mesh Flag under **ingressgateway section**.

```
ingressgateway:

  global:
    # In case of ASPEN Service Mesh enabled, to support clear text
traffic
from outside of the cluster below flag needs to be true.

    istioIngressTlsSupport:

      ingressGateway: true

  # Mandatory: This flag needs to set it "true" is Service Mesh
would be present
where UDR will be deployed
  serviceMeshCheck: true
```

- Change Ingress Gateway Service Type to ClusterIP under **ingressgateway section**.

```
ingressgateway:
  global:
    # Service Type
    type: ClusterIP
```

- Exclude actuator ports from Aspen Mesh to avoid traffic through side car. These ports are used as actuator ports (used for readiness/liveness checks) for Ingress Gateway and UDR microservices. The default actuator port (service.port.management) used for UDR microservices is 9000 and Ingress/ Egress Gateway is 9090 (ingressgateway.ports.actuatorPort). If there is no change in default ports, you can use the annotation given below.

```
nudr-nrf-client-service:
  deployment:
    customExtension:
      labels: {}
      annotations:
        traffic.sidecar.istio.io/excludeOutboundPorts: "9000,9090"
```

- Create a destination rule and service entry to enable MYSQL connectivity service to establish a connection between UDR/SLF and NDB cluster. This is outside ASM. The sample templates are as follows:
  **Creating a Service for External MySQL instance**

```
apiVersion: v1
kind: Endpoints
metadata:
  name: mysql-connectivity-service-headless
  namespace: <ocudr-namespace>
subsets:
- addresses:
  - ip: <sql-node1-ip>
  - ip: <sql-node2-ip>
 ports:
  - port: 3306
    protocol: TCP
---
apiVersion: v1
```

```
kind: Service
metadata:
  name: mysql-connectivity-service-headless
  namespace: <ocudr-namespace>
spec:
  clusterIP: None
  ports:
  - port: 3306
    protocol: TCP
    targetPort: 3306
  sessionAffinity: None
  type: ClusterIP
---
apiVersion: v1
kind: Service
metadata:
  name: mysql-connectivity-service
  namespace: <ocudr-namespace>
spec:
  externalName: mysql-connectivity-service-headless.<ocudr-
namespace>.svc.cluster.local
  sessionAffinity: None
  type: ExternalName
```

#### Creation of Service Entry and DestinationRule for External DB instance

```
apiVersion: networking.istio.io/v1alpha3
kind: ServiceEntry
metadata:
  name: mysql-external-se
  namespace: <ocudr-namespace>
spec:
  hosts:
  - mysql-connectivity-service-headless.<ocudr-
namespace>.svc.cluster.local
  ports:
  - number: 3306
    name: mysql
    protocol: MySQL
  location: MESH_EXTERNAL
---
apiVersion: networking.istio.io/v1alpha3
kind: DestinationRule
metadata:
  name: mysql-external-dr
  namespace: <ocudr-namespace>
spec:
  host: mysql-connectivity-service-headless.<ocudr-
namespace>.svc.cluster.local
  trafficPolicy:
    tls:
      mode: DISABLE
```

B

# Rollback Instructions for PCF Data to Release v15

In this Appendix, you will learn to revert to 29.519 v15.3.0 for PCF Data.

> **Note:**
>
> In a deployed UDR, you must configure only 1 version of PCF data (either v16.2.0 or v15.3.0). By default, UDR 1.6.0 supports 29.519 v16.2.0 version of PCF data. If there are multiple versions, it will create inconsistency in the json data stored on UDR.

> **Note:**
>
> If the user wants to be compatible with 29.519 v15.3 of PCF data, follow the instructions while deploying UDR and do not change, if there are some subscribers provisioned using any older schema.

**Schema Rollback to 29.519 15.3.0**

A script (***rollbackPCFschema_15_3.py***) is provided for this procedure (as part of the customer documentation). The steps to execute the script are:

1. Open the script and edit the following details as per the udrdb configurations.

   ```
   mydb = mysql.connector.connect(
   host="localhost",
   user="root",
   passwd="xxxxxxxx",
   database="udrdb")
   ```

2. Execute the below command:

   ```
   python rollbackPCFschema_15_3.py
   ```