

Oracle® Communications

Cloud Native Unified Data Repository User's Guide



Release 1.7.1

F33535-02

August 2020

The Oracle logo, consisting of a solid red square with the word "ORACLE" in white, uppercase, sans-serif font centered within it.

ORACLE®

F33535-02

Copyright © 2019, 2020, Oracle and/or its affiliates.

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

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

If this is software or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

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

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

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

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

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

Contents

1	Overview	
	Architecture	1-2
	Unified Data Repository Features	1-2
2	Configuring User Parameters	
3	Using Unified Data Repository (UDR) Console	
	Logging into CNC Console	3-1
	Global Configurations	3-2
	Service Configurations	3-3
	Data Repository Service	3-3
	Notify Service	3-5
	NRF Client Service	3-7
	Provisioning - Profile Data	3-11
	Provisioning - PCF Data	3-13
	Provisioning - SLF Data	3-15
	Provisioning - UDM Data	3-17
	Provisioning - Schema Management Data	3-18
	Logout of CNC Console	3-20
4	KPIs and Metrics	
A	Rollback Instructions for PCF Data	
B	ASM Specific Configuration	

List of Figures

3-1	CNC Console Login Screen	3-2
3-2	CNC Console Home Page	3-2
3-3	Global Configurations	3-3
3-4	Data Repository Service Configurations	3-4
3-5	Edit Datarepository Service Configurations	3-5
3-6	Notify Service Configurations	3-5
3-7	Edit Notify Service Configurations	3-7
3-8	NRF Client Configurations	3-8
3-9	Edit NRF Client Configurations	3-10
3-10	Provisioning - Profile Data	3-11
3-11	Sample Screen: Get - Profile Data - Response	3-12
3-12	Provisioning Operations for PCF Data	3-13
3-13	Provisioning PCF Get Response	3-14
3-14	Provisioning - SLF Data	3-15
3-15	Provisioning - SLF - Get Response	3-16
3-16	Provisioning Operations for UDM Data	3-17
3-17	Provisioning for Schema Management	3-19
3-18	Schema - Get - PCF AM Data	3-19
3-19	CNC Console Logout	3-20

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.

Patch Release 1.7.1

In this patch release, UDR can be deployed with service mesh like Aspen. This helps in controlling and monitoring the data flow within UDR microservices and outside as well.

Release 1.7

The following new features are introduced in this release:

- Extension of SLF provisioning with SLFGroupName on both UDR and ProvGw
- Supports enablement of various features of UDR
- Supports Helm test for NF deployment
- Supports customized labels and annotations in the Helm charts
- Supports CSAR packaging
- Integrated with Egress API gateway
- Integrated with CNC-Console for configuration of UDR
- Supports 4G policy data storage
- Support Diameter Sh interface for subscriber profile

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.
- 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.

 **Note:**

The Sh interface is available for subscriber profile operations only.

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 OCNRF. These values are useful to add custom annotation(s) to all non-Load Balancer Type Services that OCNRF 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 Credential Secret Name	ocudr-secrets	Not Applicable	
releaseDbName	Release Db Name	udr_release	Not Applicable	
configServerFullNameOverride	Config Server Full Name Override	nudr-config-server	Not Applicable	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
udrServices	Services supported on the UDR deployment, This config will decide the schema execution on the udrdb which is done by the nudr-preinstall hook pod.	nudr-group-id-map	All/nudr-dr/nudr-group-id-map	This release is specifically for SLF, so default value is nudr-group-id-map
udsfEnable	Flag to enable UDSF services on the deployment	false	true/false	
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	1.7.1	Not Applicable	
test.config.logLevel	Log level for helm test pod	WARN	Possible Values - WARN INFO DEBUG	
test.config.timeout	Timeout value for the helm test operation. If exceeded helm test will be considered as failure	120	Range: 1-300 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	
preinstall.image.tag	Image version for nudr-prehook pod image	1.7.1	Not Applicable	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
preinstall.config.logLevel	Log level for preinstall hook pod	WARN	Possible Values - WARN INFO 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 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 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 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/nudr_datarepository_service	Not applicable	
image.tag	Tag of Image	1.7.1	Not applicable	
image.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent Never	
subscriber.autocreate	Flag to enable auto creation of subscriber	true	true/false	This flag will enable 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 will control 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 - WARN INFO 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	Possible Values- ClusterIP NodePort LoadBalancer	The kubernetes service type for exposing UDR deployment Note: Suggested to be set as ClusterIP (default value) always

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
service.port.http	HTTP port	5001	Not applicable	The http port to be used in nudr-drservice service
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	
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.

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	false	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	1.7.1	Not applicable	
image.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent 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. Whether retry should be done will be based on notification.retryerrorcodes configuration.
notification.retryinterval		5	Range: 1 - 60 Unit: Seconds	The retry interval for notifications in case of failure. Unit is in seconds. 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 separated	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	25	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 - WARN INFO 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	Possible Values- ClusterIP NodePort LoadBalancer	The kubernetes service type for exposing UDR deployment 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) to nudr-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.

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	
host.baseurl	NRF url for registration	http://ocnrf-ingressgateway.mynrf.svc.cluster.local/nrf-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

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
logging.level.root	Log Level	WARN	Possible Values - WARN INFO DEBUG	Log level of the UDR nrf client pod
image.name	Docker Image name	ocudr/ nudr_nrf_client_s ervice	Not applicable	
image.tag	Tag of Image	1.7.1	Not applicable	
image.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent 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
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. 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	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
livenessProbeMaxRetry	Max retries of liveness probe 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 registration 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	
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.customExtensions.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.

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
deployment.customExtension.annotations	Custom Annotations that needs to be added to nudr-nrf-client specific deployment.	null	Not Applicable 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
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.
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.replicaCount	Replicas of nudr-config pod	1	Not applicable	Number of nudr-config pods to be maintained by replica set created with deployment
image.name	Docker Image name	ocudr/nudr_config	Not applicable	
image.pullPolicy	This setting indicates whether image needs to be pulled or not	Always	Possible Values - Always IfNotPresent Never	
image.tag	Tag of Image	1.7.1	Not applicable	
logging.level.root	Log Level	WARN	Possible Values - WARN INFO DEBUG	Log level of the nudr-config pod
maxReplicas	Maximum Replicas	1	Not applicable	Maximum number of pods
minReplicas	Minimum Replicas	1	Not applicable	Minimum number of pods
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	
resources.requests.cpu	Cpu Allotment for nudr-dr-service pod	2	Not applicable	The cpu to be allocated for nudr-config pod during deployment
resources.requests.memory	Memory allotment for nudr-dr-service pod	2Gi	Not applicable	The memory to be allocated for nudr-config pod during deployment
resources.target.averageCpuUtil	CPU utilization limit for autoscaling	80	Not Applicable	CPU utilization limit for creating HPA
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.
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.http2enabled	Enabled HTTP2 support flag for rest server	true	true/false	Enable/Disable HTTP2 support for rest server
service.port.http	HTTP port	5001	Not applicable	The http port to be used in nudr-config service
service.port.https	HTTPS port	5002	Not applicable	The https port to be used for nudr-config service
service.port.management	Management port	9000	Not applicable	The actuator management port to be used for nudr-config service
service.type	UDR service type	ClusterIP	Possible Values- ClusterIP NodePort LoadBalancer	The kubernetes service type for exposing UDR deployment Note: Suggested to be set as ClusterIP (default value) always

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

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
envLoggingLevelApp	Log Level	WARN	Possible Values - WARN INFO 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
resources.requests.cpu	Cpu Allotment for nudr-drservice pod	2	Not applicable	The cpu to be allocated for nudr-config-server pod during deployment
service.type	UDR service type	ClusterIP	Possible Values- ClusterIP NodePort LoadBalancer	The kubernetes service type for exposing UDR deployment Note: Suggested to be set as ClusterIP (default value) always
resources.requests.memory	Memory allotment for nudr-drservice pod	512Mi	Not applicable	The memory to be allocated for nudr-config-server pod during deployment
enabled	Flag to enable/disable nudr-config-server service	true	true/false	
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:latest	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	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
resources.limits.cpu	Cpu allotment limitation	2	Not applicable	
resources.limits.memory	Memory allotment limitation	2Gi	Not applicable	

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	ClusterIP	Possible Values- ClusterIP NodePort LoadBalancer	
global.metallbAllocationEnabled	Enable or disable Address Pool for Metallb	true	true/false	
global.metallbAllocationAnnotation	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.publicHttpSignalingPort	Port used on which ingressgateway listens for incoming http requests.	80	Valid Port	
global.publicHttpSignallingPort	Port used on which ingressgateway listens for incoming https requests.	443	Valid Port	
image.name	Docker image name	ocudr/ocingress_gateway	Not applicable	
image.tag	Image version tag	1.7.7	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 - Always IfNotPresent Never	
initContainersImage.name	Docker Image name	ocudr/ configurationinit	Not applicable	
initContainersImage.tag	Image version tag	1.2.0	Not applicable	
initContainersImage.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent Never	
updateContainerImage.name	Docker Image name	ocudr/ configurationupdate	Not applicable	
updateContainerImage.tag	Image version tag	1.2.0	Not applicable	
updateContainerImage.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent 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 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 will guarantee for the ingress-gateway init container, and K8s 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 ingress-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 ingress-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 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	
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	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
openTracing.jaeger.probabilisticSampler	Probabilistic 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
	Supported cipher suites for ssl	<ul style="list-style-type: none"> - TLS_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384 - TLS_ECDHE_RSA_WITH_AES_256_GCM_SHA384 - TLS_ECDHE_RSA_WITH_CHACHA20_POLY1305_SHA256 - TLS_DHE_RSA_WITH_AES_256_GCM_SHA384 - TLS_DHE_RSA_WITH_AES_256_CCM - TLS_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256 - TLS_ECDHE_RSA_WITH_AES_128_GCM_SHA256 	Not applicable	
oauthValidatorEnabled	OAUTH Configuration	false	Not Applicable	
enableIncomingHttp	Enabling for accepting http requests	true	Not Applicable	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
enableIncomingH ttps	Enabling for accepting https requests	false	true or false	
enableOutgoingH ttps	Enabling for sending https requests	false	true or false	
maxRequestsQu euedPerDestinati on	Queue Size at the ocudr- endpoint pod	5000	Not Applicable	
maxConnections PerIp	Connections from endpoint to other microServices	10	Not Applicable	
serviceMeshChe ck	Load balancing will be handled by Ingress gateway, if true it would be handled by serviceMesh	true	true/false	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
routesConfig	Routes configured to connect to different micro services of UDR	<pre> - id: traffic_mapp ing_http uri: http://{{ .R elease.Name }}-nudr- drservice:50 01 path: / nudr-dr/** order: 1 - id: traffic_mapp ing_http_pro v uri: http://{{ .R elease.Name }}-nudr- drservice:50 01 path: / nudr-dr- prov/** order: 2 - id: traffic_mapp ing_http_mgm t uri: http://{{ .R elease.Name }}-nudr- drservice:50 01 path: / nudr-dr- mgm/** order: 3 - id: traffic_mapp ing_http_uds f uri: http://{{ .R elease.Name }}-nudr- </pre>	Not Applicable	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
		<pre> drservice:50 01 path: / nudsf-dr/** order: 4 - id: traffic_mapp ing_http_gro up uri: http://{ .R elease.Name }}-nudr- drservice:50 01 path: / nudr-group- id-map/** order: 5 - id: traffic_mapp ing_http_gro up_prov uri: http://{ .R elease.Name }}-nudr- drservice:50 01 path: / nudr-group- id-map- prov/** order: 6 - id: traffic_mapp ing_http_slf _group_prov uri: http://{ .R elease.Name }}-nudr- drservice:50 01 path: /slf- group- prov/** order: 7 </pre>		

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.

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	
image.name	Docker image name	ocudr/ocuegress_gateway	Not applicable	
image.tag	Image version tag	1.7.7	Not applicable	
image.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent Never	
initContainersImage.name	Docker Image name	ocudr/configurationinit	Not applicable	
initContainersImage.tag	Image version tag	1.2.0	Not applicable	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
initContainersImage.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent Never	
updateContainerImage.name	Docker Image name	ocudr/ configurationupdate	Not applicable	
updateContainerImage.tag	Image version tag	1.2.0	Not applicable	
updateContainerImage.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent 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	
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	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
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	
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

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
service.initialAlgorithm	Algorithm to be used 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	
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	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
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	
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	ocudr-tracer-jaeger-agent.ocudr-infra	Valid FQDN	

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
openTracing.jaeger.udpSender.port	Jaeger agent service UDP port	6831	Valid Port	
openTracing.jaeger.probabilisticSampler	Probabilistic 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	
oauthClientEnabled	Enable if oauth is required	false	true or false	Enable based on Oauth configuration
nrfAuthority	Nrf Authority configuration	10.75.224.7:8085	Not Applicable	
nrfInstanceid	Nrf Instance Id	fe7d992b-0541-4c7d-ab84-c6d70b1b01b1	Not Applicable	
consumerPlmnMNC	plmnmc	345	Not Applicable	
consumerPlmnMCC	plmncc	567	Not Applicable	
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.customExtensions	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.

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
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.

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

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
enabled	To enable service.	false	Not applicable	Used to enable or disable service.
image.name	Docker Image name	ocudr/nudr_diameterproxy	Not applicable	
image.tag	Tag of Image	1.7.1	Not applicable	
image.pullPolicy	This setting will tell if image need to be pulled or not	Always	Possible Values - Always IfNotPresent Never	
logging.level.root	Log Level	WARN	Possible Values - WARN INFO 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
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

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
service.type	UDR service type	ClusterIP	Possible Values- ClusterIP NodePort LoadBalancer	The Kubernetes service type for exposing UDR deployment Note: Suggested to be set as ClusterIP (default value) always
service.diameter.type	Diameter service type	LoadBalancer	Possible Values- ClusterIP NodePort 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
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

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
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 if threadcount is 0 then application choose the threadCount based on pod profile size
diameter.IO.queueSize	Queue size for IO	0	2048 to 8192	the count should be the power of 2 if queueSize is 0 then application choose the queueSize based on pod profile size

Parameter	Description	Default value	Range or Possible Values (If applicable)	Notes
diameter.messageBuffer.threadCount	Number of threads for process the message	0	0 to 2* CPU	Number of threads to handle messages in diameterproxy pod if threadcount is 0 then application choose the threadCount based on pod profile size
diameter.peer.setting	Diameter peer setting	reconnectDelay: 3 responseTimeout: 4 connectionTimeout: 3 watchdogInterval: 6 transport: 'TCP' reconnectLimit: 50	Not Applicable	<ol style="list-style-type: none"> 1. reconnect delay for diameter reconnect (in seconds). 2. total turnaround time for process the diameter messages. (in sec) 3. TCP connection timeout time. (in sec) 4. DWR and DWA messages every number of time (in sec) 5. Transport layer 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' responseOnly: false namespace: 'seagull1' host: '10.75.185.158' domain: 'svc.cluster.local' port: 4096 realm: 'seagull1.com' identity: 'seagull1a.seagull1.com'	Not applicable	the diameter server peer node information *it should be yaml list *default values are template , how to add peer nodes.
diameter.peer.clientNodes	diameter client peers	- identity: 'seagull1a.seagull1.com' realm: 'seagull1.com' - identity: 'seagull1.com' realm: 'seagull1.com'	Not applicable	the diameter client node information *it should be yaml list *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.

3

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 and NRF client service. The CNC Console also allows you to perform provisioning operations for profile-data, PCF, SLF, UDM 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. In order to login successfully to 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:

```
10.75.224.18 cncc-iam-ingress-gateway.cncc.svc.cluster.local
10.75.224.18 cncc-core-ingress-gateway.cncc.svc.cluster.local
```

Note:

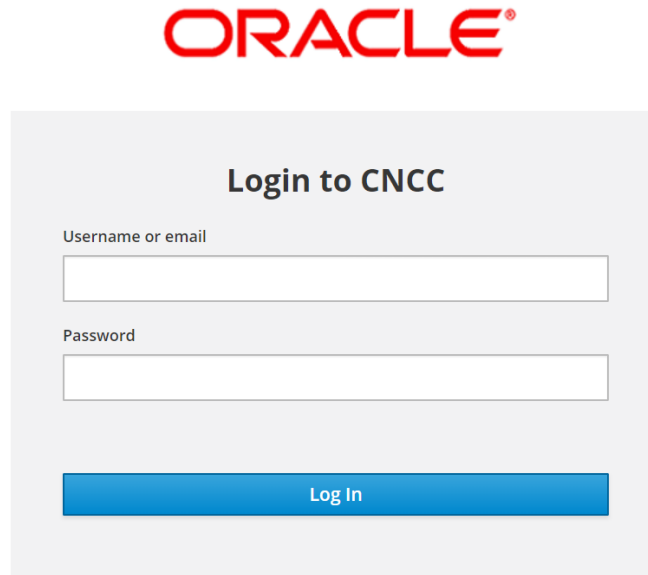
The IP Address in the above lines may change when deployment cluster changes.

Save and close the notepad.

To login to CNC Console:

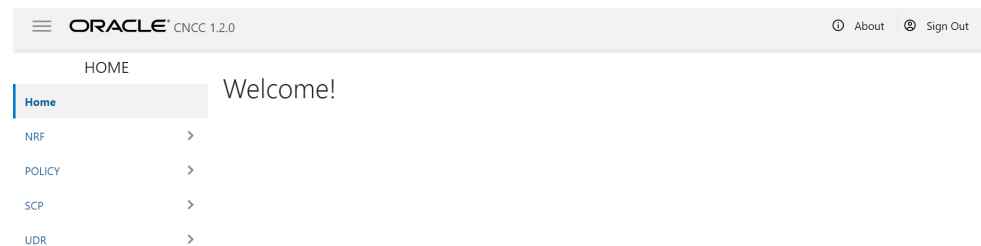
1. Type **http://cncc-core-ingress-gateway.cncc.svc.cluster.local:<PortNumber>/** in the web browser and press Enter. Following screen appears:

Figure 3-1 CNC Console Login Screen



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

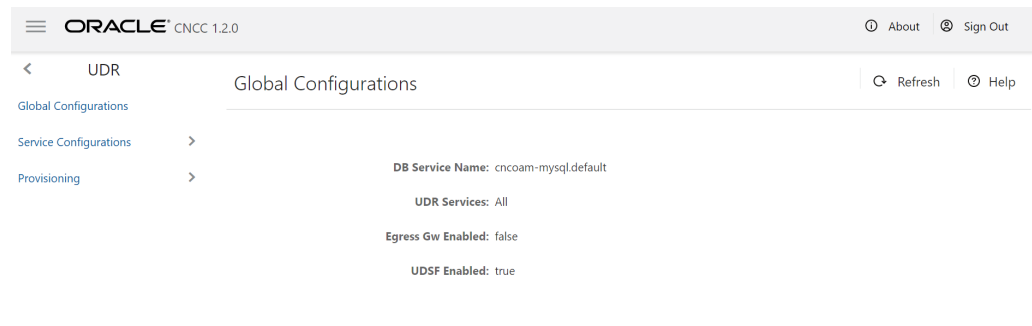
Figure 3-2 CNC Console Home Page



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 3-3 Global Configurations

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.

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

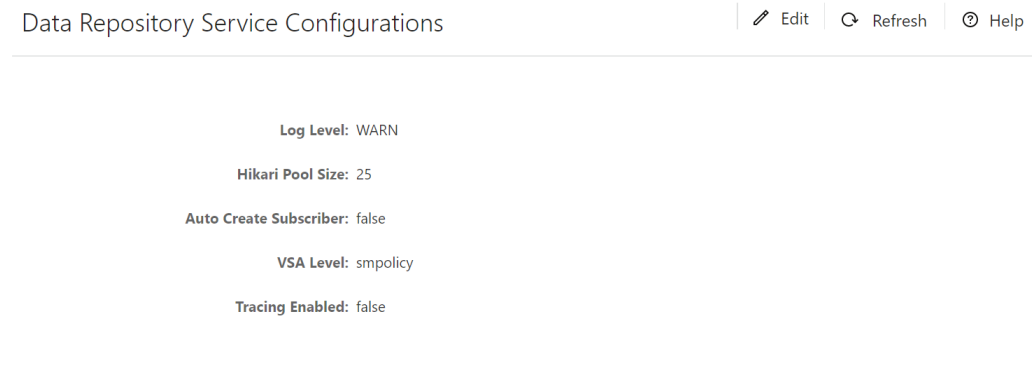
To configure UDR services using CNC Console, you can use Service Configurations. In the left navigation pane of the CNC Console application, click **UDR** → **Service Configurations**.

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

- [Data Repository Service](#)
- [Notify Service](#)
- [NRF Client Service](#)

Data Repository Service

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

Figure 3-4 Data Repository Service Configurations

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

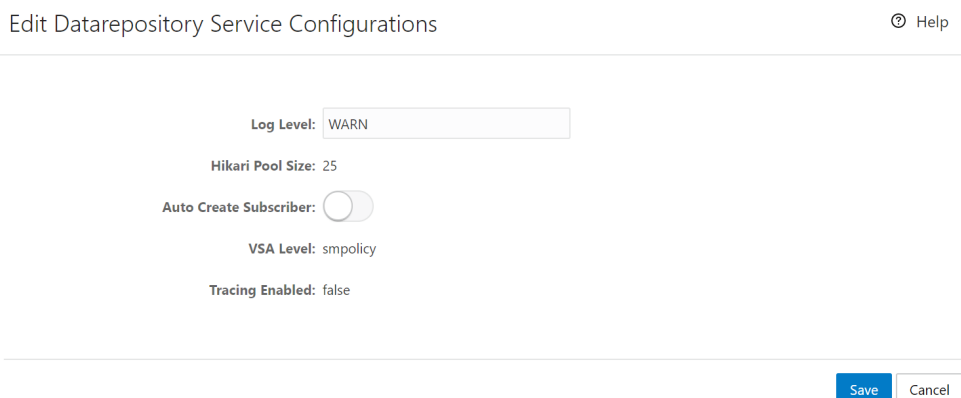
Field Name	Default Value	Attribute	Description
Log Level	WARN	Editable	This field shows the log level of the data repository service. Its possible values are: <ul style="list-style-type: none"> • WARN • INFO • DEBUG
Hikari Pool Size	25	Read-only	The hikari pool connection size to be created at start 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 3-5 Edit Datarepository Service Configurations

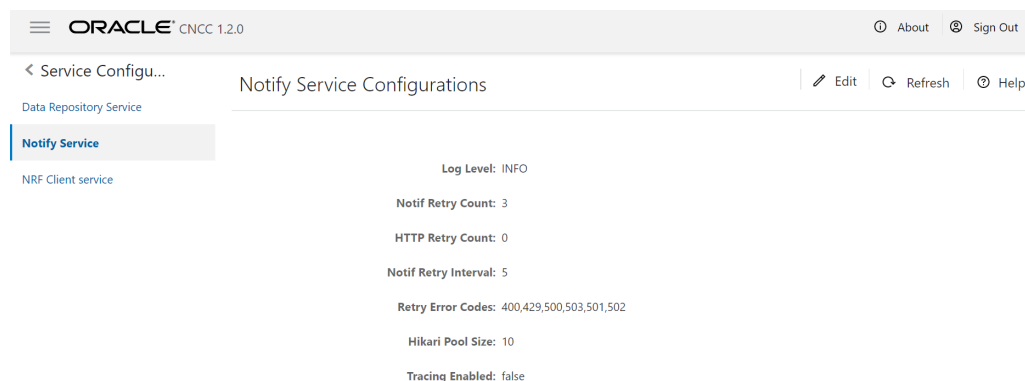


2. Update the configuration details as required and click **Save**. A confirmation message, "Save successfully." appears.
3. If you do not want to modify any configuration, click **Cancel**. You navigate back to the Data Repository Service Configurations screen.
4. Click **Refresh** to reload the Data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

Notify Service

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

Figure 3-6 Notify Service Configurations



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: <ul style="list-style-type: none"> • WARN • INFO • DEBUG
Notif Retry Count	3	Editable	Number of notification attempts to be done in case of notification failures. 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. 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 Datarepository Service Configurations** screen appears.

Figure 3-7 Edit Notify Service Configurations

Edit Notify Service Configurations 🔗 Help

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		

Save Cancel

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 3-8 NRF Client Configurations

ORACLE[®] CNCC 1.2.0

[About](#)
[Sign Out](#)

< Service Configu...
NRF Client Configurations

[Edit](#)
[Refresh](#)
[Help](#)

Data Repository Service

Notify Service

NRF Client service

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 Endpoint: 10.0.0.0

Priority: 10

Pod Capacity Multiplier: 500

▶ **PLMN List**

MCC	MNC	
310	14	Add Edit Delete

▶ **SUPI Ranges**

Start	End	
10000000000	20000000000	Add Edit Delete

▶ **GPSI**

Start	End	
10000000000	20000000000	Add 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: <ul style="list-style-type: none"> • WARN • INFO • 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 allotted to ocdur-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 registration 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 3-9 Edit NRF Client Configurations

ORACLE CNCC 1.2.0 About Sign Out

< Service Configu... Edit NRF Client Configurations Help

Data Repository Service

Notify Service

NRF Client service

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 Endpoint: 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

Save Cancel

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.

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 3-10 Provisioning - Profile Data

The screenshot shows the 'Provisioning Operations for Profile Data' screen. On the left, there is a navigation pane with the following items: Profile Data (selected), PCF Data, SLF Data, UDM Data, and Schema Management Data. The main area contains the following fields:

- Resource Name: Profile Data (dropdown menu)
- UE ID: (text input field)
- Version: v1 (dropdown menu)
- Query Parameters: (text input field)

At the bottom right, there is a toolbar with the following icons: Get, Add, Edit, Delete, and Clear.

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-111111114
3. Click **Get**. A confirmation message, '**Fetches Successfully**' appears for a second and the result appears in the **Response** text area as follows:

Figure 3-11 Sample Screen: Get - Profile Data - Response

Response	Request
<pre>{ "profile-data": { "NAI": ["3333333332", "3333333331", "3333333333"], "IMSI": ["2222222221", "2222222222", "2222222223"], "MSISDN": ["1111111112", "1111111114", "1111111114"] } }</pre>	

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 3-12 Provisioning Operations for PCF Data

Provisioning Operations for PCF Data

Resource Name:	Access And Mobility Policy Data ▾
UE ID:	<input type="text"/>
Version:	v2 ▾
Query Parameters:	<input type="text"/>

⊕ Get |
 ✎ Edit |
 🗑 Delete |
 ⊗ Clear

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

- Select **Resource Name** as either Access And Mobility Policy Data, Session Management Policy Data or UE Policy Set.
- Extract PCF data from UDR database on the basis of UE ID entered
- Modify an existing PCF data on the basis of UE ID

- Delete an 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 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. Enter **Query Parameters**.
4. Click **Get**. A confirmation message, '**Fetchd Successfully.**' appears and the result appears in the **Response** text area as follows:

Figure 3-13 Provisioning PCF Get Response

Response	Request
<pre>{ "pralnfos": { "p1": { "prald": "p1", "ecgiList": [{ "plmnlid": { "mcc": "976", "mnc": "32" }, "eutraCellId": "92FFdBE" }, { "plmnlid": { "mcc": "977", "mnc": "32" }, "eutraCellId": "92FFdBE" }] } } }</pre>	

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

Adding or Editing a PCF Data

To add or edit 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. Enter the **Query Parameters**.
4. Click **Edit**. The **Request** text area displays an existing PCF data.
5. Type-in the updated profile data.
6. 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. Enter the **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 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 3-14 Provisioning - SLF Data

The screenshot shows the 'Provisioning Operations for SLF Data' interface. On the left, a navigation menu lists 'Profile Data', 'PCF Data', 'SLF Data' (highlighted), 'UDM Data', and 'Schema Management Data'. The main content area has the title 'Provisioning Operations for SLF Data' and contains the following fields:

- Resource Name: SLF GroupName (dropdown menu)
- UE ID: (text input field)
- Version: v1 (dropdown menu)
- Query Parameters: (text input field)

At the bottom right of the form, there are four action icons: Get, Edit, Delete, and Clear.

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

- Extract SLF data from UDR database on the basis of UE ID entered.
- Modify an existing SLF data on the basis of UE ID.
- Delete an existing SLF data that is no more needed in the UDR database.

Extracting SLF Data

To extract the SLF data:

1. Select the **Resource Name** as either SLF GroupName or NFGroupID Map.
2. Type an appropriate **UE ID**. For example: msisdn-111111112
3. Enter the **Query Parameters**.
4. Click **Get**. A confirmation message, '**Fetchd Successfully.**' appears and the result appears in the **Response** text area as follows:

Figure 3-15 Provisioning - SLF - Get Response

Response	Request
<pre>{ "nfGroupIDs": { "NRF": "nrf-group-name", "UDM": "udm-group-name" } }</pre>	

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

Adding or Editing a SLF Data

To add or edit a SLF data:

1. Select the **Resource Name** as either SLF GroupName or NFGroupID Map.
2. Type an appropriate **UE ID**. For example: msisdn-111111112
3. Click **Edit**. The **Request** text area displays an existing SLF data.
4. Type-in the updated SLF data.
5. 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 GroupName or NFGroupID Map.
2. Type an appropriate **UE ID**. For example: msisdn-111111112

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 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 3-16 Provisioning Operations for UDM Data

Provisioning Operations for UDM Data

Resource Name:	Access And Mobility Subscriptio... ▼
UE ID:	<input type="text"/>
PDU Session ID:	<input type="text"/>
Serving PLMN ID:	<input type="text"/>
Version:	v1 ▼
Query Parameters:	<input type="text"/>

 Get |  Edit |  Delete |  Clear

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, '**Fetchd 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 or Editing a UDM Data

To add or 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 3-17 Provisioning for Schema Management

Schema Management

Resource Name: PCF AM Data

API Version: v1

Query Parameters:

Get Edit Delete Rollback Clear

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, '**Fetches Successfully**,' appears and the result appears in the **Response** text area as follows:

Figure 3-18 Schema - Get - PCF AM Data

Resource Name: PCF AM Data

Get Edit Delete Clear

Response Request

```
{
  "$schema": "http://json-schema.org/draft-07/schema#",
  "type": "object",
  "properties": {
    "subscats": {
      "minItems": "1",
      "type": "array",
      "items": {
        "type": "string"
      }
    },
    "pralnfos": {
      "patternProperties": {
        "^(*)$": {
          "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 exist, an Error message summary appears.

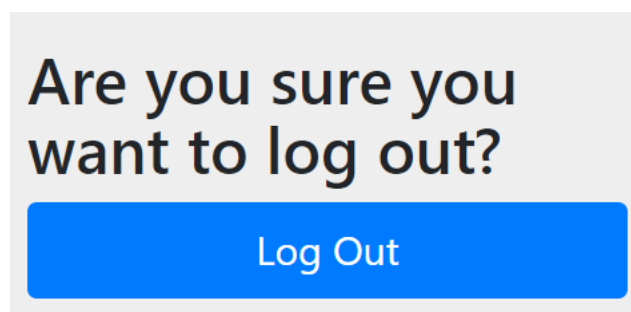
Clearing a Profile Data

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 3-19 CNC Console Logout



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

4

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_https_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-dr-service (All UDR operations)
		UDRTotalResponses	udr_rest_allresponse_total	Total number of responses sent by nudr-dr-service (All UDR operations)
		UDRTotalSuccessResponses	udr_rest_successResponse_total	Total number of success responses sent by nudr-dr-service (All UDR operations)
		UDRTotalFailureResponses	udr_rest_failureResponse_total	Total number of failure responses sent by nudr-dr-service (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_mgmt_operations_success_total{Method="PUT"}	Total number of successful schema updates processed by nudr-drservice
		UDRPutSchemaFailure	udr_schema_mgmt_operations_failure_total{Method="PUT"}	Total number of failure schema updates processed by nudr-drservice
	GET	UDRGetSchemaSuccess	udr_schema_mgmt_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_successsResponse_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-dr-service
		UDRDeleteSubscriptionResponseFailure	nudr_dr_subscription_failed_response_total{Method="DELETE"}	Total number of failed DELETE Subscription response sent by nudr-dr-service
Notification	Notifications to notification service	UDRNotificationRequests	nudr_dr_notif_signal_sent_total	Total notification sent successfully by nudr-dr-service 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 receives 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 receive 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 specific to each nf type,method type and provisioning 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 provisioning 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 provisioning requests for each nf type

Category	Sub-category	Description	Metric Name	Notes
		UDR NF Type Specific Metrics - Success response method wise	udr_nf_type_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_nf_type_specific_requests_status{NFType="PCF/UDM/SLF",StatusCode=~"4.*",Method="GET/POST/PUT/PATCH/DELETE"}+udr_nf_type_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.*"}+ 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 subscriptions 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-dr-service
		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_failed_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 - Notifications 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 - Notifications 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	UDRNRFDeRegistrationRequests Sent	udr_nrf_deregistration_requests_total	Total number of registration requests sent by nrf-client-service to NRF
		UDRNRFDeRegistrationRequests Successful	udr_nrf_deregistration_success_total	Total number for successfully processed deregistration requests
Heartbeat towards NRF	Heartbeat	UDRNRFHeartBeatUpdateRequestsSent	udr_nrf_heartbeat_update_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
		UDRNRFHeartBeatUpdateRequestsSuccessful	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
		UDRNRFRegistrationRequestsSuccessful	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
		UDRNRFDeRegistrationRequestsSuccessful	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
		UDRNRFHeartBeatUpdateRequestsSuccessful	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 successfully	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_responses_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_User Data Requests	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_U DA_Success_Me ssages	nudr_diameterpr oxy_responses_t otal{MessageTyp e="UDA", ResultCode="200 1"}	The total number of success User Data Answer messages received on diameterproxy.
		Diameterproxy_U DA_Failure_Mess ages	nudr_diameterpr oxy_responses_t otal{MessageTyp e="UDA", ResultCode ! ="2001"}	The total number of failure User Data Answer messages received on diameterproxy.
		Diameterproxy_U DR_Rejected_By _Unknown_User	nudr_diameterpr oxy_request_reje cted_unknown_u ser_total{Messag eType="UDR"}	The total number User data requests messages failed because of Unknown user.
		Diameterproxy_U DR_Rejected_By _Invalid_Service _Indication	nudr_diameterpr oxy_rejected_inv alid_service_indi cation_requests_ total{MessageTyp e="UDR"}	The total number User Data request messages failed because of invalid service passed.
	PROFILE UPDATE	Diameterproxy_P UR_Messages	nudr_diameterpr oxy_requests_tot al{MessageType= "PUR"}	The total number of profile update request messages received on diameterproxy.
		Diameterproxy_P UA_Success_Me ssages	nudr_diameterpr oxy_responses_t otal{MessageTyp e="PUA", ResultCode="200 1"}	The total number of success profile data update requests
		Diameterproxy_P UA_Failure_Mess ages	nudr_diameterpr oxy_responses_t otal{MessageTyp e="PUA", ResultCode ! ="2001"}	The total number of failure profile data update requests
		Diameterproxy_P UR_Rejected_By _Unknown_User	nudr_diameterpr oxy_request_reje cted_unknown_u ser_total{Messag eType="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_received_From_Notify-service	nudr_diameterproxy_total_notification_received_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_res ponses_total{Method= 'GET',StatusCode="40 4"}
Response received	PUT	Pegging the response of PUT Request with response code : 200	nudr_config_total_res ponses_total{Method= 'PUT',StatusCode="20 0"}
Response received	PUT	Pegging the response of PUT Request with response code : 400	nudr_config_total_res ponses_total{Method= 'PUT',StatusCode="40 0"}
Response received	PUT	Pegging the response of PUT Request with response code : 500	nudr_config_total_res ponses_total{Method= 'PUT',StatusCode="50 0"}
Response received	POST	Pegging the response of POST Request with response code : 201	nudr_config_total_res ponses_total{Method= 'POST',StatusCode="2 01"}
Response received	POST	Pegging the response of POST Request with response code : 400	nudr_config_total_res ponses_total{Method= 'POST',StatusCode="4 00"}
Response received	POST	Pegging the response of POST Request with response code : 500	nudr_config_total_res ponses_total{Method= 'POST',StatusCode="5 00"}

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_ingr essgateway_http _requests_total[5 m]))
No of Responses/sec	All	UDR Ingress Response Rate	Not Applicable	sum(irate(oc_ingr essgateway_http _responses_tota l[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_ingr essgateway_http _responses_tota l{Method="POST", Status=~"2.*"} [5m]))/ sum(irate(oc_ingr essgateway_http _responses_tota l[5m]))

KPI Details	Service Operation	KPI	Response Code	Notes
	PutRequests	rate of Put Requests with success response	201	$\frac{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"PUT"}, \text{Status}=\sim\text{"2.*"}\}[5\text{m}]))}{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}[5\text{m}]))}$
	GetRequests	rate of Get Requests with success response	200	$\frac{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"GET"}, \text{Status}=\sim\text{"2.*"}\}[5\text{m}]))}{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}[5\text{m}]))}$
	DeleteRequests	rate of Delete Requests with success response	204	$\frac{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"DELETE"}, \text{Status}=\sim\text{"2.*"}\}[5\text{m}]))}{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}[5\text{m}]))}$
	PatchRequests	rate of Patch Requests with success response	204	$\frac{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"PATCH"}, \text{Status}=\sim\text{"2.*"}\}[5\text{m}]))}{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}[5\text{m}]))}$
No of 4xx responses of each type/No of 4xx responses per second	PostRequest	rate of PostRequests failed with 4xx	4xx	$\frac{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"POST"}, \text{Status}=\sim\text{"4.*"}\}[5\text{m}]))}{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Status}=\sim\text{"4.*"}\}[5\text{m}]))}$

KPI Details	Service Operation	KPI	Response Code	Notes
	PutRequests	rate of PutRequests failed with 4xx	4xx	$\frac{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"PUT"}, \text{Status}=\sim\text{"4.*"}\}[5\text{m}]))}{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Status}=\sim\text{"4.*"}\}[5\text{m}]))}$
	GetRequests	rate of GetRequests failed with 4xx	4xx	$\frac{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"GET"}, \text{Status}=\sim\text{"4.*"}\}[5\text{m}]))}{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Status}=\sim\text{"4.*"}\}[5\text{m}]))}$
	DeleteRequests	rate of DeleteRequests failed with 4xx	4xx	$\frac{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"DELETE"}, \text{Status}=\sim\text{"4.*"}\}[5\text{m}]))}{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Status}=\sim\text{"4.*"}\}[5\text{m}]))}$
	PatchRequests	rate of PatchRequests failed with 4xx	4xx	$\frac{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"PATCH"}, \text{Status}=\sim\text{"4.*"}\}[5\text{m}]))}{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Status}=\sim\text{"4.*"}\}[5\text{m}]))}$

KPI Details	Service Operation	KPI	Response Code	Notes
No of 5xx responses of each type/No of 5xx responses per second	PostRequest	rate of PostRequests failed with 5xx	5xx	$\frac{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_requests_total}\{\text{Method}=\text{"POST"}\}[5\text{m}])) - \text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"POST"}\}[5\text{m}]))}{(\text{sum}(\text{irate}(\text{oc_ingressgateway_http_requests_total}[5\text{m}])) - \text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}[5\text{m}])))}$
	PutRequests	rate of PutRequests failed with 5xx	5xx	$\frac{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_requests_total}\{\text{Method}=\text{"PUT"}\}[5\text{m}])) - \text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"PUT"}\}[5\text{m}]))}{(\text{sum}(\text{irate}(\text{oc_ingressgateway_http_requests_total}[5\text{m}])) - \text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}[5\text{m}])))}$
	GetRequests	rate of GetRequests failed with 5xx	5xx	$\frac{\text{sum}(\text{irate}(\text{oc_ingressgateway_http_requests_total}\{\text{Method}=\text{"GET"}\}[5\text{m}])) - \text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"GET"}\}[5\text{m}]))}{(\text{sum}(\text{irate}(\text{oc_ingressgateway_http_requests_total}[5\text{m}])) - \text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}[5\text{m}])))}$

KPI Details	Service Operation	KPI	Response Code	Notes
	DeleteRequests	rate of DeleteRequests failed with 5xx	5xx	$\frac{(\text{sum}(\text{irate}(\text{oc_ingressgateway_http_requests_total}\{\text{Method}=\text{"DELETE"}\}\{5\text{m}\})) - \text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"DELETE"}\}\{5\text{m}\})))}{(\text{sum}(\text{irate}(\text{oc_ingressgateway_http_requests_total}\{5\text{m}\})) - \text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{5\text{m}\})))}$
	PatchRequests	rate of PatchRequests failed with 5xx	5xx	$\frac{(\text{sum}(\text{irate}(\text{oc_ingressgateway_http_requests_total}\{\text{Method}=\text{"PATCH"}\}\{5\text{m}\})) - \text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"PATCH"}\}\{5\text{m}\})))}{(\text{sum}(\text{irate}(\text{oc_ingressgateway_http_requests_total}\{5\text{m}\})) - \text{sum}(\text{irate}(\text{oc_ingressgateway_http_responses_total}\{5\text{m}\})))}$
Percentage of success requests of each type of operations	PostRequest	Percentage of Post Requests with success response	201	$\frac{\text{sum}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"POST"}, \text{Status}=\sim\text{"2.*"}\})}{\text{sum}(\text{oc_ingressgateway_http_responses_total})}$
	PutRequests	Percentage of Put Requests with success response	2xx	$\frac{\text{sum}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"PUT"}, \text{Status}=\sim\text{"2.*"}\})}{\text{sum}(\text{oc_ingressgateway_http_responses_total})}$

KPI Details	Service Operation	KPI	Response Code	Notes
	GetRequests	Percentage of Get Requests with success response	200	$\frac{\text{sum(oc_ingressgateway_http_responses_total\{Method="GET",Status=\sim"2.*"})}}{\text{sum(oc_ingressgateway_http_responses_total)}}$
	DeleteRequests	Percentage of Delete Requests with success response	204	$\frac{\text{sum(oc_ingressgateway_http_responses_total\{Method="DELETE",Status=\sim"2.*"})}}{\text{sum(oc_ingressgateway_http_responses_total)}}$
	PatchRequests	Percentage of Patch Requests with success response	204	$\frac{\text{sum(oc_ingressgateway_http_responses_total\{Method="PATCH",Status=\sim"2.*"})}}{\text{sum(oc_ingressgateway_http_responses_total)}}$
Percentage of 4xx requests of each type of operations	PostRequest	Percentage of PostRequests failed with 4xx	4xx	$\frac{\text{sum(oc_ingressgateway_http_responses_total\{Method="POST",Status=\sim"4.*"})}}{\text{sum(oc_ingressgateway_http_responses_total\{Status=\sim"4.*"})}}$
	PutRequests	Percentage of PutRequests failed with 4xx	4xx	$\frac{\text{sum(oc_ingressgateway_http_responses_total\{Method="PUT",Status=\sim"4.*"})}}{\text{sum(oc_ingressgateway_http_responses_total\{Status=\sim"4.*"})}}$
	GetRequests	Percentage of GetRequests failed with 4xx	4xx	$\frac{\text{sum(oc_ingressgateway_http_responses_total\{Method="GET",Status=\sim"4.*"})}}{\text{sum(oc_ingressgateway_http_responses_total\{Status=\sim"4.*"})}}$

KPI Details	Service Operation	KPI	Response Code	Notes
	DeleteRequests	Percentage of DeleteRequests failed with 4xx	4xx	$\frac{\text{sum(oc_ingressgateway_http_responses_total\{Method="DELETE", Status=~"4.*"\})}}{\text{sum(oc_ingressgateway_http_responses_total\{Status=~"4.*"\})}}$
	PatchRequests	Percentage of PatchRequests failed with 4xx	4xx	$\frac{\text{sum(oc_ingressgateway_http_responses_total\{Method="PATCH", Status=~"4.*"\})}}{\text{sum(oc_ingressgateway_http_responses_total\{Status=~"4.*"\})}}$
Percentage of 5xx requests of each type of operations	PostRequest	Percentage of PostRequests failed with 5xx	5xx	$\frac{\text{sum(oc_ingressgateway_http_requests_total\{Method="POST"\}) - \text{sum(oc_ingressgateway_http_responses_total\{Method="POST"\})}}{\text{sum(oc_ingressgateway_http_requests_total) - \text{sum(oc_ingressgateway_http_responses_total)}}$
	PutRequests	Percentage of PutRequests failed with 5xx	5xx	$\frac{\text{sum(oc_ingressgateway_http_requests_total\{Method="PUT"\}) - \text{sum(oc_ingressgateway_http_responses_total\{Method="PUT"\})}}{\text{sum(oc_ingressgateway_http_requests_total) - \text{sum(oc_ingressgateway_http_responses_total)}}$

KPI Details	Service Operation	KPI	Response Code	Notes
	GetRequests	Percentage of GetRequests failed with 5xx	5xx	$(\text{sum}(\text{oc_ingressgateway_http_requests_total}\{\text{Method}=\text{"GET"}\}) - \text{sum}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"GET"}\})) / ((\text{sum}(\text{oc_ingressgateway_http_requests_total}) - \text{sum}(\text{oc_ingressgateway_http_responses_total}))$
	DeleteRequests	Percentage of DeleteRequests failed with 5xx	5xx	$(\text{sum}(\text{oc_ingressgateway_http_requests_total}\{\text{Method}=\text{"DELETE"}\}) - \text{sum}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"DELETE"}\})) / ((\text{sum}(\text{oc_ingressgateway_http_requests_total}) - \text{sum}(\text{oc_ingressgateway_http_responses_total}))$
	PatchRequests	Percentage of PatchRequests failed with 5xx	5xx	$(\text{sum}(\text{oc_ingressgateway_http_requests_total}\{\text{Method}=\text{"PATCH"}\}) - \text{sum}(\text{oc_ingressgateway_http_responses_total}\{\text{Method}=\text{"PATCH"}\})) / ((\text{sum}(\text{oc_ingressgateway_http_requests_total}) - \text{sum}(\text{oc_ingressgateway_http_responses_total}))$

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	$\text{sum}(\text{irate}(\text{oc_egressgateway_http_requests_total}[5\text{m}]))$

KPI Details	Service Operation	KPI	Response Code	Notes
No of Responses/sec	All	UDR Egress Response Rate	Not Applicable	sum(irate(oc_egr essgateway_http _requests_total[5 m]))
No of Successful Responses of each type/No of Successful Responses per second	PostRequest	rate of Post/Get/Put/Delete Requests with success response. 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_egr essgateway_http _responses_total{Method="POST",Status=~"2.*"}[5m]))/sum(irate(oc_egr essgateway_http _responses_total[5m]))
No of 4xx responses of each type/No of 4xx responses per second	PostRequest	rate of PostRequests failed with 4xx. 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_egr essgateway_http _responses_total{Method="POST",Status=~"4.*"}[5m]))/sum(irate(oc_egr essgateway_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. 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_egr essgateway_http _requests_total{Method="POST"}[5m]))-sum(irate(oc_egr essgateway_http _responses_total{Method="POST"}[5m])))/sum(irate(oc_egr essgateway_http _requests_total[5m]))-sum(irate(oc_egr essgateway_http _responses_total[5m]))

KPI Details	Service Operation	KPI	Response Code	Notes
Percentage of success requests of each type of operations	PostRequest	Percentage of Post Requests with success response. NOTE : In metric name just replace with POST with any other method which you want to check(ex:GET,PUT.etc)	201	$\frac{\text{sum(oc_egressgateway_http_responses_total\{Method="POST",Status=~"2.*"\})}}{\text{sum(oc_egressgateway_http_responses_total)}}$

A

Rollback Instructions for PCF Data

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
```

B

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:
  # 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

```