Oracle® Communications
Network Repository Function (NRF) Cloud Native User's Guide

Release 1.2

ORACLE®

Oracle Communications Network Repository Function (NRF) Cloud Native User's Guide, Release 1.2

F21619-01

Contents

List of Figures

List of Tables

ORACLE

1
Introduction

This document provides information about the role of Network Repository Function (NRF) in 5G Service Based Architecture.

The NRF is a key component of the 5G Service Based Architecture. The NRF maintains an updated repository of all the Network Functions (NFs) available in the operator's network along with the services provided by each of the NFs in the 5G core that are expected to be instantiated, scaled and terminated with minimal to no manual intervention. In addition to serving as a repository of the services, the NRF also supports discovery mechanisms that allows NFs to discover each other and get updated status of the desired NFs.

The NRF supports the following functions:

- Maintains the profiles of the available NF instances and their supported services in the 5G core network

- Allows consumer NF instances to discover other providers NF instances in the 5G core network

- Allows NF instances to track the status of other NF instances

The NRF interacts with every other NF in the 5G core network and it supports the above functions through the following services:

- **Management Services**

- **Discovery Services**

Terminology and Acronyms

The following table provides information about the acronyms and the terminology used in the document.

Table 1-1   Terminology/Acronyms

Field	Description
3GPP	3rd Generation Partnership Project
5G-AN	5G Access Network
5GC	5G Core Network
5G System	3GPP system consisting of 5G Access Network (AN), 5G Core Network and UE
AMF	Access and Mobility Management Function
API Gateway	Application that sits in front of an application programming interface (API) and acts as a single point of entry for a defined group of micro services.
CNE	Cloud Native Environment

Table 1-1   (Cont.) Terminology/Acronyms

Dimension	Dimension is a tag of Metric. For Example, "ocnrf_nfRegister_rx_requests_total {{ OriginatorNfType }} {{NrfLevel }} {{NfInstanceId }}" In the example above, OriginatorNfType, NrfLevel, and NfInstanceId are dimensions.
DNS	Domain Name System
FQDN	Fully Qualified Domain Name
KBs	Kubernetes
KPI	Key Performance Indicator
MMI	Machine Machine Interface
MPS	Messages Per Second
NDB	Network Database
NF	Network Function
Network Function	A functional building block within a network infrastructure, which has well defined external interfaces and well defined functional behavior. In practical terms, a network function is often a network node or physical appliance.
Network Slice	A logical network that provides specific network capabilities and network characteristics.
Network Slice instance	A set of Network Function instances and the required resources (e.g. compute, storage and networking resources) which form a deployed Network Slice.
NF Consumer	A generic way to refer to an NF which consumes services provided by another NF. Ex: An AMF is referred to as a Consumer when it consumes AMPolicy services provided by the PCF.
NF Instance	A specific instance of a network function type.
NF Producer or NF Provider	A generic way to refer to an NF which provides services that can be consumed by another NF. Ex: A PCF is a provider NF and provides AMPolicy Services
NRF	Network Repository Function or Network Function Repository Function
PCF	Policy Control Function
PLMN	Public Land Mobile Network
Resiliency	The ability of the NFV framework to limit disruption and return to normal or at a minimum acceptable service delivery level in the fame of a fault, failure, or an event that disrupts normal operation.
Scaling	Ability to dynamically extend/reduce resources granted to the Virtual Network Function (VNF) as needed. This includes scaling out/in or scaling up/down.
Scaling Out/In/ Horizontally	The ability to scale by add/remove resource instances (e.g. VMs). Also called scaling Horizontally.
Scaling Up/Down/ Vertically	The ability to scale by changing allocated resources, e.g. increase/decrease memory, CPU capacity or storage size.
SCP	Service Communication Proxy
SEPP	Security Edge Protection Proxy
SMF	Session Management Function
URI	Uniform Resource Identifier

NRF References

External References

- 3GPP TS 23.501 V15.3.0 - System Architecture for the 5G System

- 3GPP TS 23.502 V15.3.0 - Procedures for the 5G System

- 3GPP TS 29.510 V15.3.0 - 5G System; Network Function Repository Function; Stage 3

My Oracle Support

My Oracle Support (https://support.oracle.com) is your initial point of contact for all product support and training needs. A representative at Customer Access Support can assist you with My Oracle Support registration.

Call the Customer Access Support main number at 1-800-223-1711 (toll-free in the US), or call the Oracle Support hotline for your local country from the list at http://www.oracle.com/us/support/contact/index.html. When calling, make the selections in the sequence shown below on the Support telephone menu:

1. Select **2** for New Service Request.

2. Select **3** for Hardware, Networking and Solaris Operating System Support.

3. Select one of the following options:

    - For Technical issues such as creating a new Service Request (SR), select **1**.

    - For Non-technical issues such as registration or assistance with My Oracle Support, select **2**.

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

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

Locate Product Documentation on the Oracle Help Center Site

Oracle Communications customer documentation is available on the web at the Oracle Help Center (OHC) site, http://docs.oracle.com. You do not have to register to access these documents. Viewing these files requires Adobe Acrobat Reader, which can be downloaded at http://www.adobe.com.

1. Access the Oracle Help Center site at http://docs.oracle.com.

2. Click `Industries`.

3. Under the Oracle Communications subheading, click the `Oracle Communications documentation` link.

    The Communications Documentation page appears. Most products covered by these documentation sets will appear under the headings "Network Session Delivery and Control Infrastructure" or "Platforms."

4. Click on your Product and then the Release Number.

    A list of the entire documentation set for the selected product and release appears.

5. To download a file to your location, right-click the PDF link, select Save target as (or similar command based on your browser), and save to a local folder.

2
OCNRF Supported Services

This section includes information about the services supported by OCNRF.

OCNRF supports the following services:

NRF Management Services

The NRF Management service is identified by the service operation name `Nnrf_NFManagement`.

OCNRF supports the following management services:

> **Note:**
>
> The respective service operation name is mentioned next to each service.

- **Register NF instance** (`NFRegister`): Allows an NF instance to register its NF profile in the NRF along with the list of services provided by the NF instance.

- **Update NF instance** (`NFUpdate`): Enables an NF instance to partially update or replace the parameters of its NF profile in the NRF. It also allows to add or delete services provided by the NF instance.
  This operation supports the following:
  - Complete Replacement of NF profile
  - Add, Remove, or Update attributes of NF Profile
  - Heart beat & Load info of NF

- **De-register NF instance** (`NFDeregister`): Enables an NF instance to de-register its NF profile and the services provided by the NF instance from the 5G network.

- **Subscribe to Status** (`NFStatusSubscribe`): Enables an NF instance to subscribe the status changes of other NF instances registered in the NRF.

- **Unsubscribe to Status** (`NFStatusUnsubscribe`): Enables an NF instance to unsubscribe the status changes of other NF instances.

- **Receive Notifications of Status** (`NFStatusNotify`): Enables the NRF to notify changes in status of NF instances to any subscriber of NF status. Changes also include information regarding newly registered and de-registered NFs.

- **NFListRetrieval**: Allows the retrieval of a list of NF Instances that are currently registered in NRF. This service operation is not allowed to be invoked from an NRF in a different PLMN.

- **NFProfileRetrieval**: Allows the retrieval of the NF profile of a given NF instance currently registered in NRF. This service operation is not allowed to be invoked from an NRF in a different PLMN.

Discovery Service

The NRF Discovery service is identified by the service operation name `Nnrf_NFDiscovery Service`.

OCNRF supports the following Discovery service:

> **Note:**
>
> The respective service operation name is mentioned next to the supported Discovery service.
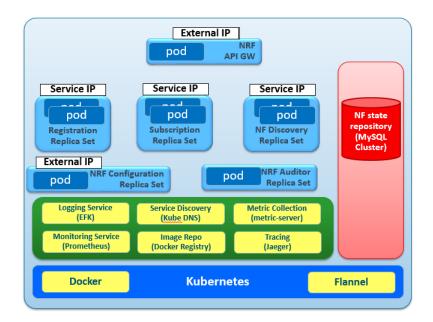
Discover NF instance (`NFDiscover`): NRF supports discovery of the IP address/FQDN of the NF instances, or NF Services that match certain input criteria.

3

OCNRF Architecture

OCNRF comprises of various microservices deployed in Kubernetes based Cloud Native Environment (CNE, example: OC-CNE). Some common services like logs or metrics data collection, analysis and graphs or charts visualization, etc. are provided by the environment. The microservices integrate with them and provide them necessary data. The following diagram describes the overall architecture of the OCNRF:

Figure 3-1   Oracle Communications Network Repository Function Architecture Diagram

The followings are the components of OCNRF product:

- **NRF API Gateway**
  All NRF service requests are sent to the NRF API Gateway. The API Gateway hides the deployment details of the NRF services and other components from other NFs in the CNE and external entities. It is deployed in the CNE as a Kuberbetes service by the NRF Helm Chart.

  > **Note:**
  >
  > **Ambassador** is used as NRF API Gateway

  The primary functions of the NRF API Gateway are as follows:

  – It shall be the NRF endpoint hosting the FQDN(s) of the NRF deployment

- Inspects the service URI and route the traffic to the appropriate upstream NRF micro-service

- Measures the request count routed to each micro-service

- Measures the response count based on the HTTP Status Code

- Generates HTTP Trace data for the requests routed through

- Generates the logs of the routing activity

- Integrates with CNE common services EFK, Prometheus and Jaeger for sending them logs, metrics data and HTTP Tracing data respectively

- **NF Registration MicroService**
  This microservice handles the following service operations:

  - receives and handles NFRegister service requests from the NFs

  - receives and handles NFUpdate service requests from the NFs

  - receives and handles NFDeregister service requests from the NFs

  - receives and handles NFListRetrieval service requests from the NFs

  - receives and handles NFProfileRetrieval service requests from the NFs

  - receives and handles the Heart-beat messages from the NFs

- **NF Subscription MicroService**
  This microservice handles the following service operations:

  - receives and handles NFStatusSubscribe service requests from the NFs

  - receives and handles NFStatusUnsubscribe service requests from the NFs

  - sends NFStatusNotify service requests towards the subscribed NFs

  - stores the subscription data in its own data store using the database service

- **NF Discover MicroService**
  This microservice handles the following service operations:

  - receives and handles NFDiscover service requests from the NFs

- **OCNRF Auditor MicroService**
  This microservice is internal to OCNRF. This microservice performs the following tasks:

  - finds and deletes the expired subscription records

  - finds and deletes the profile records which have been SUSPENDED for a very long time

  - processes the heartbeat records and update them if a heartbeat is missed

  - transitions the profiles from REGISTERED to SUSPENDED state if n consecutive heartbeats are missed

  - monitors the heart-beat expiry, mark the NF profiles as suspended and act appropriately on the suspended NF profiles

- **NRF Configuration Microservice**
  This microservice is used to do the configuration in OCNRF which can be changed dynamically by a operator/user using REST based interface. The dynamic configuration data or Provisioning data is application configuration data which is required for an OCNRF screening feature to function in a desired manner. This configuration data is managed by the OCNRF configuration service and is stored in a separate data store.

- **Data Service**

The data service is deployed by the CNE provider. It is a common CNE service with persistent data storage and highly available clustered MySql. The OCNRF microservices create their data store using the Data Service upon initialization.

> **Note:**
>
> For information on configuring NRF microservices, See *the OCNRF Cloud Native Installation and Upgrade Guide*.

4

NF Screening

NF Screening supports the functionality to screen the service requests received from 5G Network Functions (NFs) before allowing access to OCNRF services.

In this feature, OCNRF screens the incoming service operations from NFs on the basis of some attributes against set of rules configured at OCNRF. Once screening is passed successfully, OCNRF allows the Network Functions to perform the service operations.

This feature will provide extra security by restricting the NF that can use the service of OCNRF. Operator can decide which NF with required attributes can access the services provided by OCNRF. To implement this, operator can configure various screening lists in which attributes can be configured to tell which attribute is allowed or not.

> **Note:**
>
> By default, NF Screening feature is globally disabled. This feature can be enabled by setting the **nfScreeningRulesListStatus** attribute as "ENABLED" using REST based Interface.

For Configuring NF Screening feature, see Configuring NF Screening

Screening Lists

The screening can be in the form of Whitelists or Blacklists.

- When a screening list is configured to operate as a whitelist, the request is allowed to access the service only if the corresponding attribute value is present in the whitelist.

- When a screening list is configured to operate as a blacklist, the request is allowed to access the service only if the corresponding attribute value is not present in the blacklist.

Screening Lists can have rules for global and for each NF type.

- The global level screening lists allows operators to configure screening that is common to all NFs .

- Each NF Type level rules provides additional flexibility/granularity for screening that can be controlled on a per NF type basis.

It should be noted that an operator can configure rules at both Global level and Per NF Type level and requests are processed through both lists if configured. The only list that is available only at the global level and not at the per NF type is the "NF type list allowed to Register".

NF Screening feature supports following screening lists:

- Management Service Screening Lists: The screening lists identified in the Table 4-1 apply exclusively to NRF management service request.

Table 4-1   Management Service Screening Lists

S.No.	Screening List	Supported Screening Type	Screening Scope	Attribute in Request	Comment
1	NF FQDN list	Whitelist, Blacklist	Global, Per NF Type	• fqdn in NFProfile • fqdn in NFService	NRF screens the FQDN present in the request before allowing access to management service.
2	NF IP endpoint list	Whitelist, Blacklist	Global, Per NF Type	• ipv4Addresses, ipv6Addresses in NFProfile • ipEndPoints in NFService	NRF screens the IP endpoint/s present in the request before allowing access to management service.
3	Callback URI FQDN and IP list	Whitelist, Blacklist	Global, Per NF Type	• defaultNotificationSubscriptions in NFProfile and NFService (Implicit subscription) • nfStatusNotificationUri in SubscriptionData (explicit subscription)	NRF screens the callback URI present in the request before allowing access to management service. Host present in callback URI (FQDN+port or IP+port) should be used for screening. Check the callback URI format in section 4.4.3 (Callback URI structure) in TS 29.501-f20.
4	PLMN id list	Whitelist, Blacklist	Global, Per NF Type	plmnList in NFProfile	NRF screens the PLMN ID present in the request before allowing access to management service.
5	NF types allowed to Register	Whitelist	Global	nfType in NFProfile	NRF screens the NF type present in the request before allowing registration to 5G NFs.

5

Configuring Oracle Communications Network Repository Function

This section provides information for configuring Oracle Communications Network Repository Function.

OCNRF REST based configuration can be divided into two types.

- General Configuration i.e. OCNRF system options (which may be used by all the microservices of OCNRF)

- Configuring NF Screening

For information on other OCNRF configurations, refer to *OCNRF Cloud Native Installation and Upgrade Guide*.

General Configurations

Below table provides information for configuring general configurations in OCNRF.

General configuration - OCNRF system options

Table 5-1   Service API Interface

Resource Name	Resource URI	HTTP Method or Custom Operation	Description
nrf-configuration (Store)	{apiRoot}/nrf-configuration/v1/system-options	GET	Retrieves NRF system options configuration
nrf-configuration (Store)	{apiRoot}/nrf-configuration/v1/system-options	PUT	Updates NRF system options configuration

Resource Standard Methods
GET - Retrieve NRF System options configuration

Table 5-2   Data structures supported by the GET Response Body

Data Type	P	Cardinality	Response Codes	Description
ProblemDetails	M	1	500 Internal Server Error	The response body contains the error reason of the request message.
NrfSystemOptions	M	1	200 OK	Response body contains the NRF current system options

PUT - Update NRF System options configuration

Table 5-3 Data structures supported by the PUT Request Body

Data Type	P	Cardinality	Description
NrfSystemOptions	M	1	NrfSystemOptions details

Table 5-4 Data structures supported by the PUT Response Body

Data Type	P	Cardinality	Response Codes	Description
ProblemDetails	M	1	500 Internal Server Error	The response body contains the error reason of the request message.
ProblemDetails	M	1	400 Bad request	The response body contains the error reason of the request message.
NA			204 No content	Specifies that the update of NrfSystemOptions is successful.

Example 5-1 Data Model

> **Note:**
>
> For the attributes, other than **profilesCountInDiscoveryResponse** and **discoveryResultLoadThreshold**, if the attribute is not present, existing value in the database is used. It can be the default value or the last updated value, but at least one attribute should be present to not make PUT request empty.

Table 5-5 NrfSystemOptions - Parameters

Attribute Name	Data Type	Constraints	P	Default Values	Description
nfScreeningFeatureStatus	NfScreeningFeatureStatus		O	DISABLED	This attribute specifies if NF Screening is enabled or not.
nfScreeningFailureHttpCode	INTEGER		O	403 (Forbidden)	This attribute specifies what HTTP status code is returned if incoming request doesn't pass NF Screening rules barrier.

Table 5-5   (Cont.) NrfSystemOptions - Parameters

Attribute Name	Data Type	Constraints	P	Default Values	Description
nfHeartBeatTimer	INTEGER	Range - 30 seconds to 300 seconds	O	90	Value entered for this attribute is treated as seconds. If Heartbeat timer value is not received in NFProfile during NFRegister, this default value is used by OCNRF. If Heartbeat timer value is received in NFProfile during NFRegister, minimum value is used for validation and limit purpose. It means if value provided is less than minimum value, then minimum value is taken as heartbeat timer value. If Heartbeat timer value is received in NFProfile during NFRegister, maximum value of range is used for validation and limit purpose. It means if value provided is more than maximum value, then maximum value is taken as heartbeat timer value.
nrfPlmnList	array		O	MCC - 310 MNC - 14	310 MCC is for USA and 14 is for test purposes.

Table 5-5   (Cont.) NrfSystemOptions - Parameters

Attribute Name	Data Type	Constraints	P	Default Values	Description
nfNotifyLoadThreshold	INTEGER	0 - 99	O	5	Notification trigger is generated only when difference between the 'load' value reported by NF in most recent heartbeat and the 'load' reported in the last generated notification is more than configured value of the loadThreshold parameter.

Table 5-5   (Cont.) NrfSystemOptions - Parameters

Attribute Name	Data Type	Constraints	P	Default Values	Description
subscriptionValidityDuration	INTEGER	1 hour - 180 days (4320 hours)	O	2160	Value entered for this attribute is treated as hours. If Validity time attribute is not received in SubscriptionData during NFSubscribe, this default value is used for calculation of validity time (current time + default duration). If Validity time attribute is received in SubscriptionData during NFSubscribe, this minimum value is used for validation and limit purpose. It means if value provided is less than (current time + minimum value), then calculated value with minimum duration value is considered as validity time of subscription and similarly in case validity time is more than (current time + maximum duration), then calculated value with maximum duration is considered as validity time of subscription.

Table 5-5   (Cont.) NrfSystemOptions - Parameters

Attribute Name	Data Type	Constraints	P	Default Values	Description
nrfSupportForProfileChangesInNotification	ENUM (TRUE, FALSE)	TRUE or FALSE	O	FALSE	Flag to enable/disable to send profileChanges attribute instead of NFProfile in Notification.
nfProfileSuspendDuration	INTEGER		O	168	Value entered for this attribute is treated as hours. An NF is allowed to be in SUSPENDED state for this duration before it is deleted from OC-NRF database.
nfHearbeatMissAllowed	INTEGER		O	3	NF HeartBeat miss allowed after which NFProfile is marked as suspended.
discoveryValidityPeriod	INTEGER		O	3600	Value entered for this attribute is treated as seconds. Discovery request is valid till this value. Requester NF discovers again to get latest values.

Table 5-5   (Cont.) NrfSystemOptions - Parameters

Attribute Name	Data Type	Constraints	P	Default Values	Description
profilesCountInDiscoveryResponse	INTEGER	1 - 10	C	3	NFDiscover response contains NF profiles with count limited to this configured value. If this attribute is not present in configuration request then this functionality is disabled. In that case, all the profiles are returned, while fetching the record. In case, configured value is less than the configured range, then minimum value of range is considered. Similarly, if configured value is more than the configured range, then maximum value of range is considered. While running GET on NrfSystemOptions resource if this value is null, then it means this feature is disabled.

Table 5-5   (Cont.) NrfSystemOptions - Parameters

Attribute Name	Data Type	Constraints	P	Default Values	Description
discoveryResultLoadThreshold	INTEGER	1 - 100	C		NFDiscover response contains NF profiles with load attribute value less than this configured value. If this attribute is not present in configuration request then this functionality is disabled. While running GET on NrfSystemOptions resource if this value is null, then it means this feature is disabled.

Example 5-2   REST Message Sample

Request_Type: PUT

URL: *http://<k8s host>:<port>/nrf-configuration/v1/system-options*

```
{

    "nfScreeningFeatureStatus": "DISABLED",

    "nfScreeningFailureHttpCode": 403,

    "nfHeartBeatTimer": 90,

    "nrfPlmnList": [

        {

            "mcc": "310",

            "mnc": "14"

        }

    ],

    "nfNotifyLoadThreshold": 5,

    "subscriptionValidityDuration": 2160,

    "nrfSupportForProfileChangesInNotification": false,
```

```
    "nfProfileSuspendDuration": 168,

    "nfHearbeatMissAllowed": 3,

    "discoveryValidityPeriod": 3600,

    "profilesCountInDiscoveryResponse": 3

}
```

Request_Type: GET

URL: *http://<k8s host>:<port>/nrf-configuration/v1/system-options*

```
{

    "nfScreeningFeatureStatus": "DISABLED",

    "nfScreeningFailureHttpCode": 403,

    "nfHeartBeatTimer": 90,

    "nrfPlmnList": [

        {

            "mcc": "310",

            "mnc": "14"

        }

    ],

    "nfNotifyLoadThreshold": 5,

    "nrfSupportForProfileChangesInResponse": false,

    "subscriptionValidityDuration": 2160,

    "nrfSupportForProfileChangesInNotification": false,

    "nfProfileSuspendDuration": 168,

    "nfHearbeatMissAllowed": 3,

    "discoveryValidityPeriod": 3600,

    "profilesCountInDiscoveryResponse": 3,

    "discoveryResultLoadThreshold": null

}
```

Configuring NF Screening

This section provides information for configuring NF Screening.

RESTful Service Architecture to Configure Rules for NF Screening

Table 5-6   Resources and Methods Overview

Resource Name	Resource URI	HTTP Method or Custom Operation	Description
screening-rules (Store)	{apiRoot}/nrf-configuration/v1/screening-rules	GET	Returns all the screening rules
screening-rules (Document)	{apiRoot}/nrf-configuration/v1/screening-rules/{nfScreeningRulesListType}	GET	Returns screening rules corresponding to the specified NF Screening Rule List Type.
screening-rules (Document)	{apiRoot}/nrf-configuration/v1/screening-rules/{nfScreeningRulesListType}	PUT	Replace the complete specified NF Screening Rule List Type
screening-rules (Document)	{apiRoot}/nrf-configuration/v1/screening-rules/{nfScreeningRulesListType}	PATCH	Partially updates the specified NF Screening Rule List Type.

Resource Standard Methods
PUT - Updates a particular screening rule (except read only attributes)

Table 5-7   Data structures supported by the PUT Request Body

Data Type	P	Cardinality	Description
NfScreening Rules	M	1	NF Screening Rules which need to be updated.

Table 5-8   Data structures supported by the PUT Response Body

Data Type	P	Cardinality	Response Codes	Description
NA	M	1	204 No content	Successful response
ProblemDetails	C	1	404 NOT FOUND 500 INTERNAL ERROR 400 BAD REQUEST	The response body contains the error reason of the request message.

PATCH - Updates partially a particular screening rule (except read only attributes)

Table 5-9   Data structures supported by the PATCH Request Body

Data Type	P	Cardinality	Description
PatchDocument	M	1	It contains the list of changes to be made to the NF Screening Rule, according to the JSON PATCH format specified in IETF RFC 6902 [13].

Table 5-10   Data structures supported by the PATCH Response Body

Data Type	P	Cardinality	Response Codes	Description
NA	M	1	204 No content	Successful response
ProblemDetails	C	1	404 NOT FOUND 500 INTERNAL ERROR 400 BAD REQUEST	The response body contains the error reason of the request message.

GET - Collection of screening rules

Table 5-11   URI query parameters supported by the GET method

Name	Data Type	P	Cardinality	Description
nfScreeningRulesListType	NfScreeningRulesListType	O	0.1	The type of NF screening rules on this basis of rules list type.
nfScreeningRulesListStatus	NfScreeningRulesListStatus	O	0.1	Screening Rules List on the basis of status (Enabled or Disabled)

Table 5-12   Data structures supported by the GET Request Body

Data Type	P	Cardinality	Description
n/a			

Table 5-13   Data structures supported by the GET Response Body

Data Type	P	Cardinality	Response Codes	Description
ScreeningRulesResult	M	1	200 OK	The response body contains a list of screening lists, or an empty object if there are no screening rules to return in the query result.
ProblemDetails	C	1	500 INTERNAL ERROR 400 BAD REQUEST	The response body contains the error reason of the request message.

Table 5-14   ScreeningRulesResult - Parameters

Attribute Name	Data type	P	Cardinality	Description
nfScreeningRulesList	array (NfScreeningRules)	M	0.N	It shall contain an array of NF Screening List. An empty array means there is no NF Screening list configured.

GET - Particular screening list rule

Table 5-15   Data structures supported by the GET Request Body

Data Type	P	Cardinality	Description
n/a			

Table 5-16   Data structures supported by the GET Response Body

Data Type	P	Cardinality	Response Codes	Description
NfScreeningRules	M	1	200 OK	The response body contains requested screening list.
ProblemDetails	C	1	500 INTERNAL ERROR 400 BAD REQUEST	The response body contains the error reason of the request message.

Table 5-17   NfScreeningRules - Parameters

Attribute Name	Data type	P	Description
nfScreeningRulesListType	NfScreeningRulesListType	C	ReadOnly. It will be returned while retrieving the rule.
nfScreeningType	NfScreeningType	M	Screening type of complete screening list. Blacklist or whitelist. All the rules can be either blacklist or whitelist.
nfScreeningRulesListStatus	NfScreeningRulesListStatus	M	This attribute will enable or disable complete screening list.
globalScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if global screening rules need to be configured.
customNfScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for custom NF need to be configured.
nrfScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for NRF need to be configured.
udmScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for UDM need to be configured.
amfScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for AMF need to be configured.
smfScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for custom SMF need to be configured.
ausfScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for AUSF need to be configured.

Table 5-17   (Cont.) NfScreeningRules - Parameters

Attribute Name	Data type	P	Description
nefScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for NEF need to be configured.
pcfScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for PCF need to be configured.
nssfScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for NSSF need to be configured.
udrScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for UDR need to be configured.
lmfScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for IMF need to be configured.
gmlcScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for GMLC need to be configured.
fiveG_EirScreeningRules	NfScreeningRulesData	O	
seppScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for SEPP need to be configured.
upfScreeningRulesData	NfScreeningRulesData	O	
n3iwfScreeningRulesData	NfScreeningRulesData	O	
afScreeningRulesData	NfScreeningRulesData	O	
udsfScreeningRulesData	NfScreeningRulesData	O	
bsfScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for BSF need to be configured.
chfScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules for CHF need to be configured.
nwdafScreeningRulesData	NfScreeningRulesData	O	This attribute will be present if screening rules forNWDAF need to be configured.

Table 5-18   NfScreeningRulesData - Parameters

Attribute Name	Data type	P	Description
failureAction	FailureAction	M	
nfFqdn	NfFqdn	C	If this attribute is present in message it shouldn't be null. This attribute will be present if screeningListType is NF_FQDN.
nfCallBackUriList	array(NfCallBackUri)	C	If this attribute is present in message it shouldn't be null. This attribute will be present if screeningListType is CALLBACK_URI.
nfIpEndPointList	array(NfIpEndPoint)	C	If this attribute is present in message it shouldn't be null. This attribute may be present if screeningListType is NF_IP_ENDPOINT.
plmnList	array(PlmnId)	C	If this attribute is present in message it shouldn't be null. This attribute may be present if screeningListType is PLMN_ID.

Table 5-18   (Cont.) NfScreeningRulesData - Parameters

Attribute Name	Data type	P	Description
nfTypeList	array(NfTypeList)	C	If this attribute is present in message it shouldn't be null. This attribute may be present if screeningListType is NF_TYPE_REGISTER.

Table 5-19   NfScreeningRulesListType - Parameters

Enumeration Value	Description
"NF_FQDN"	Screening List type for NF FQDN
"NF_IP_ENDPOINT"	Screening list type for IP Endpoint
"CALLBACK_URI"	Screening list type for callback URIs in NF Service and nfStatusNotificationUri in SubscriptionData
"PLMN_ID"	Screening list type for PLMN ID
"NF_TYPE_REGISTER"	Screening list type for allowed NF Types to register

Table 5-20   NfScreeningType - Parameters

Enumeration Value	Description
"BLACKLIST"	When a screening list is configured to operate as a blacklist, the request is allowed to access the service only if the corresponding attribute value is not present in the blacklist.
"WHITELIST"	When a screening list is configured to operate as a whitelist, the request is allowed to access the service only if the corresponding attribute value is present in the whitelist.

Table 5-21   NfScreeningRulesListStatus - Parameters

Enumeration Value	Description
"ENABLED"	Screening List enabled to get apply
"DISABLED"	Screening List disabled to get apply

Table 5-22   FailureAction - Parameters

Enumeration Value	Description
"CONTINUE"	Continue Processing
"SEND_ERROR"	Send response with configured HTTP status code

Table 5-23   NfFqdn - Parameters

Attribute Name	Data type	P	Description
fqdn	array(FQDN)	C	Exact Fqdn to be matched. This is conditional, at least one attribute shall be present.

Table 5-23   (Cont.) NfFqdn - Parameters

Attribute Name	Data type	P	Description
pattern	array(string)	C	Regular Expression for FQDN. This is conditional, at least one attribute shall be present.

Table 5-24   NfIpEndPoint - Parameters

Attribute Name	Data type	P	Description
ipv4Address	Ipv4Addr	C	IPv4 address to be matched.
ipv4AddressRange	Ipv4AddressRange	C	Range of IPv4 addresses.
ipv6Address	Ipv6Addr	C	IPv6 address to be matched.
ipv6AddressRange	Ipv6AddressRange	C	Range of IPv6 addresses.
port	array(integer)	O	If this attribute is not configured then it will not be considered for validation.
portRange	array(PortRange)	O	If this attribute is not configured then it will not be considered for validation.

> **Note:**
>
> Depending on the conditions, only one of the ipv4Address, ipv4AddressRange, ipv6Address, and ipv6AddressRange attributes can be present.

Table 5-25   NfCallBackUri - Parameters

Attribute Name	Data type	P	Description
fqdn	FQDN	C	Exact Fqdn to be matched.
pattern	string	C	Regular Expression for FQDN, Ipv4Address, Ipv6Address.
ipv4Address	Ipv4Addr	C	IPv4 address to be matched.
ipv4AddressRange	Ipv4AddressRange	C	Range of IPv4 addresses.
ipv6Address	Ipv6Addr	C	IPv6 address to be matched.
ipv6AddressRange	Ipv6AddressRange	C	Range of IPv6 addresses.
port	array(integer)	O	If this attribute is not configured then it will not be considered for validation.
portRange	array(PortRange)	O	If this attribute is not configured then it will not be considered for validation.

> **Note:**
>
> Depending on the conditions, only one of the fqdn, pattern, ipv4Address,
> ipv4AddressRange, ipv6Address, and ipv6AddressRange attributes can be present.

Table 5-26   PortRange - Parameters

Attribute Name	Data type	P	Description
start	integer	M	First value identifying the start of port range.
end	integer	M	Last value identifying the end of port range.

Table 5-27   Ipv6AddressRange - Parameters

Attribute Name	Data type	P	Description
start	Ipv6Addr	M	First value identifying the start of an IPv6 Address range.
end	Ipv6Addr	M	Last value identifying the end of an IPv6 Address range.

Table 5-28   Common data types

Data Type	Reference
Ipv6Addr	3GPP TS 29.571
Ipv4Addr	3GPP TS 29.571
Ipv4AddressRange	3GPP TS 29.510
PlmnId	3GPP TS 29.571
Uri	3GPP TS 29.571
IpEndPoint	3GPP TS 29.510
NFType	3GPP TS 29.510
ProblemDetails	3GPP TS 29.571

REST message samples

Screening List Update

NF screening rules to update particular rule configuration (except read only attributes)

URL: *http://host:port/nrf-configuration/v1/ screening-rules /CALLBACK_URI*
Request_Type: PUT

Content-Type: *application/json*
Request Body

NF screening rules to get all of the configured rules

```
{

    "nfScreeningType": "BLACKLIST",

    "nfScreeningRulesListStatus": "ENABLED",
```

```
"globalScreeningRulesData": {

    "failureAction": "SEND_ERROR",

    "nfCallBackUriList": [

        {

            "ipv4AddressRange":{

                "start": "155.90.171.123",

                "end": "233.123.19.165"

            },

            "ports":[10,20]

        },

        {

            "ipv6AddressRange":{

                "start": "1001:cdba:0000:0000:0000:0000:3257:9652",

                "end": "3001:cdba:0000:0000:0000:0000:3257:9652"

            }

        }

    ]

},

"amfScreeningRulesData": {

    "failureAction": "CONTINUE",

    "nfCallBackUriList": [

        {

            "fqdn": "ocnrf-d5g.oracle.com"

        },

        {

            "ipv4AddressRange":{

                "start": "155.90.171.123",

                "end": "233.123.19.165"

            },

            "ports":[10,20]

        }
```

```
                    ]

            }

    }
```

URL: *http://host:port/nrf-configuration/v1/ screening-rules /*
Request_Type: GET

Response Body

```
{

    "nfScreeningRulesList": [

        {

            "nfScreeningRulesListType": "NF_FQDN",

            "nfScreeningType": "BLACKLIST",

            "nfScreeningRulesListStatus": "DISABLED"

        },

        {

            "nfScreeningRulesListType": "NF_IP_ENDPOINT",

            "nfScreeningType": "BLACKLIST",

            "nfScreeningRulesListStatus": "ENABLED",

            "amfScreeningRulesData": {

                "failureAction": "SEND_ERROR",

                "nfIpEndPointList": [

                    {

                        "ipv4Address": "198.21.87.192",

                        "ports": [

                            10,

                            20

                        ]

                    }

                ]

            }

        },

        {
```

```
"nfScreeningRulesListType": "CALLBACK_URI",

"nfScreeningType": "BLACKLIST",

"nfScreeningRulesListStatus": "ENABLED",

"globalScreeningRulesData": {

    "failureAction": "SEND_ERROR",

    "nfCallBackUriList": [

        {

            "fqdn": "ocnrf-d5g.oracle.com",

            "ports": [

                10,

                20

            ]

        }

    ]

}

},

{

"nfScreeningRulesListType": "PLMN_ID",

"nfScreeningType": "BLACKLIST",

"nfScreeningRulesListStatus": "DISABLED"

},

{

"nfScreeningRulesListType": "NF_TYPE_REGISTER",

"nfScreeningType": "WHITELIST",

"nfScreeningRulesListStatus": "ENABLED",

"globalScreeningRulesData": {

    "failureAction": "SEND_ERROR",

    "nfTypeList": [

        "AMF",

        "SMF",
```

```
                "PCF"

            ]

        }

    }

]

}
```

NF screening rules to get a particular configured rule

URL: *http://host:port/nrf-configuration/v1/ screening-rules /CALLBACK_URI*
Request_Type: GET

Response Body

```
{

    "nfScreeningRulesListType": "CALLBACK_URI",

    "nfScreeningType": "BLACKLIST",

    "nfScreeningRulesListStatus": "ENABLED",

    "globalScreeningRulesData": {

        "failureAction": "SEND_ERROR",

        "nfCallBackUriList": [

            {

                "ipv4AddressRange": {

                    "start": "155.90.171.123",

                    "end": "233.123.19.165"

                },

                "ports": [

                    10,

                    20

                ]

            },

            {

                "ipv6AddressRange": {

                    "start": "1001:cdba:0000:0000:0000:0000:3257:9652",

                    "end": "3001:cdba:0000:0000:0000:0000:3257:9652"

                }
```

```
                }

            ]

        },

        "amfScreeningRulesData": {

            "failureAction": "SEND_ERROR",

            "nfCallBackUriList": [

                {

                    "fqdn": "ocnrf-d5g.oracle.com"

                },

                {

                    "ipv4AddressRange": {

                        "start": "155.90.171.123",

                        "end": "233.123.19.165"

                    },

                    "ports": [

                        10,

                        20

                    ]

                }

            ]

        }

    }
```

NF screening rules for partial rule update

URL: *http://host:port/nrf-configuration/v1/ screening-rules /CALLBACK_URI*
Request_Type: PUT

Content-Type: *application/json-patch+json*
Request Body

```
[

    {"op":"remove","path":"/globalScreeningRulesData/nfCallBackUriList/2/ports/
0"},

    {"op":"replace","path":"/globalScreeningRulesData/failureAction","value":
"CONTINUE"}
```

]

URL: *http://host:port/nrf-configuration/v1/ screening-rules /CALLBACK_URI*
Request_Type: PUT

Content-Type: *application/json-patch+json*
Response Body

```
[{"op":"add","path":"/nrfScreeningRulesData","value": {"failureAction":
"SEND_ERROR","nfCallBackUriList": [{"ipv4AddressRange":{"start
```

6

OCNRF Metrics, KPIs, and Alerts

OCNRF Metrics

This section includes information about Metrics for Oracle Communications Network Repository Function.

> **Note:**
>
> Sample OCNRF dashboard for Grafana is delivered to the customer through OCNRF Custom Templates. Metrics and functions used to achieve KPI are already covered in OCNRF Custom Templates. Please view the Oracle Help Center site for the information about OCNRF Custom Templates.

Dimensions Legend for the Metrics:

The following table includes the details about the metrics dimensions:

Dimension	Details
NrfLevel	OCNRF Deployment Name by which OCNRF can be identified nrfName can be configured during helm install using command line.
NfType	Types of Network Functions (NF)
NfInstanceId	Unique identity of the NF Instance
OriginatorNfType	NF which originating the request
HttpStatusCode	HTTP Status Code
ServiceName	Name of the service instance (e.g. "nudm-sdm")
ServiceInstanceId	Unique ID of the service instance within a given NF Instance
UpdateType(Partial/Complete)	NF Update with Put (Complete) or Patch (Partial) methods
OperationType	This dimension is for NFSubscribe Service operation to tell if the request is to create or update the subscription
NotificationEventType	This dimension will indicate subscription request is for which event types. For example:- NF_REGISTERED, NF_DEREGISTERED and NF_PROFILE_CHANGED
TargetNfType	This dimension will indicate request is for which target NF type
RequesterNfType	This dimension will indicate the NF type which originated the request
BucketSize	This dimension will indicate how many profiles are returned in the response of Discovery request. Range is not configurable. Possible values are 0-10, +Inf. According to NF profiles returned, corresponding bucket will be incremented by one. For example, if 2 NF profiles are returned, then bucket 2 will be incremented by one. Profiles getting returned more than 10 will fall in +Inf bucket.

Table 6-1   NRF Common Metrics

Metric	Metric Details	Dimensions	Notes
envoy_http_ingress_http_rq_total	No of Requests		
envoy_cluster_cluster_ocnrf_nfregistration_8080_upstream_rq_201	No of Registration Requests which results in success CREATED response. (NfRegister)		
envoy_cluster_cluster_ocnrf_nfregistration_8080_upstream_rq_200	No of Registration Requests which results in success OK response. (NfUpdate/ NfProfileRetrieval/ NfListRetrieval/ NfHeartbeat)		
envoy_cluster_cluster_ocnrf_nfregistration_8080_upstream_rq_204	No of Registration Requests which results in success NO CONTENT response. (NfDelete/NfHeartbeat)		NFDeregister and NFUpdate (heartbeat) are counted together
envoy_cluster_cluster_ocnrf_nfsubscription_8080_upstream_rq_201	No of Subscription Requests which results in success CREATED response. (NfSubscribe)		
envoy_cluster_cluster_ocnrf_nfsubscription_8080_upstream_rq_204	No of Subscription Requests which results in success NO CONTENT response. (NfUnsubscribe)		
envoy_cluster_cluster_ocnrf_nfdiscovery_8080_upstream_rq_200	No of Discovery Requests which results in success OK response. (NfDiscover)		
envoy_cluster_cluster_ocnrf_nfregistration_8080_upstream_rq_4xx	No of Registration Requests which results in Error Response of category 4xx. (NfRegister/NfUpdate/ NfDelete/ NfProfileRetrieval/ NfListRetrieval)		
envoy_cluster_cluster_ocnrf_nfsubscription_8080_upstream_rq_4xx	No of Subscription Requests which results in Error Response of category 4xx. (NfSubscribe/ NfUnsubscribe)		

Table 6-1   (Cont.) NRF Common Metrics

Metric	Metric Details	Dimensions	Notes
envoy_cluster_cluster_ocnrf_nfdiscovery_8080_upstream_rq_4xx	No of Discovery Requests which results in Error Response of category 4xx. (NfDiscover)		
envoy_cluster_cluster_ocnrf_nfregistration_8080_upstream_rq_5xx	No of Registration Requests which results in Error Response of category 5xx. (NfRegister/NfUpdate/ NfDelete/ NfProfileRetrieval/ NfListRetrieval)		
envoy_cluster_cluster_ocnrf_nfsubscription_8080_upstream_rq_5xx	No of Subscription Requests which results in Error Response of category 5xx. (NfSubscribe/ NfUnsubscribe)		
envoy_cluster_cluster_ocnrf_nfdiscovery_8080_upstream_rq_5xx	No of Discovery Requests which results in Error Response of category 5xx. (NfDiscover)		
ocnrf_nfRegister_rx_requests_total	Number of Registration Requests received.	NrfLevel NfInstanceId OriginatorNfType	
ocnrf_nfRegister_tx_responses_total	Number of Registration Responses sent.	NrfLevel NfInstanceId OriginatorNfType HttpStatusCode	
ocnrf_nfRegister_rx_requests_success_perService_total	Number of Registrations received and processed successfully per Service.	NrfLevel NfInstanceId ServiceName ServiceInstanceId	
ocnrf_nfUpdate_rx_requests_total	Number of Update Requests received.	NrfLevel NfInstanceId OriginatorNfType UpdateType(Partial/ Complete)	

Table 6-1   (Cont.) NRF Common Metrics

Metric	Metric Details	Dimensions	Notes
ocnrf_nfUpdate_tx_responses_total	Number of Update Responses sent.	NrfLevel NfInstanceId OriginatorNfType UpdateType(Partial/ Complete) HttpStatusCode	
ocnrf_nfUpdate_rx_requests_success_perService_total	Number of NfUpdates received and processed successfully per Service.	NrfLevel, Updatetype =(Partial/ Complete), NfInstanceId, ServiceName, ServiceInstanceId	
ocnrf_nfHeartbeat_rx_requests_total	Number of Heartbeat Requests received	NrfLevel NfInstanceId OriginatorNfType	
ocnrf_nfHeartbeat_tx_responses_total	Number of Heartbeat Responses sent	Nrflevel, NfInstanceId, OriginatorNfType, HttpStatusCode	
ocnrf_nfDeregister_rx_requests_total	Number of Deregistration requests received	NrfLevel, NfInstanceId, OriginatorNfType	
ocnrf_nfDeregister_tx_responses_total	Number of Deregistration responses sent	NrfLevel, NfInstanceId, OriginatorNfType, HttpStatusCode	
ocnrf_nfDeregister_rx_requests_success_perService_total	Number of Deregistrations requests received and process successfully per Service	NrfLevel, ServiceName, ServiceInstanceId, NfInstanceId	
ocnrf_nfListRetrieval_rx_requests_total	Number of nfListRetrieval requests received	NrfLevel, OriginatorNfType	

ORACLE®

Table 6-1   (Cont.) NRF Common Metrics

Metric	Metric Details	Dimensions	Notes
ocnrf_nfListRetrieval_tx_responses_total	Number of nfListRetrieval responses sent	NrfLevel, OriginatorNfType, HttpStatusCode	
ocnrf_nfProfileRetrieval_rx_requests_total	Number of nfProfileRetrieval requests received	NrfLevel, NfInstanceId	
ocnrf_nfProfileRetrieval_tx_responses_total	Number of nfProfileRetrieval responses sent	NrfLevel, NfInstanceId, HttpStatusCode	
ocnrf_heartbeat_missed_total	Number of heartbeats missed.	NrfLevel, NfType, NfInstanceId	
ocnrf_nfStatusSubscribe_rx_requests_total	Number of NStatusfSubscribe requests received ocnrf_nfStatusSubscribe_rx_requests_total	NrfLevel, OriginatorNfType	
ocnrf_nfStatusSubscribe_tx_responses_total	Number of NfStatusSubscribe responses sent	NrfLevel, OriginatorNfType, HttpStatusCode	
ocnrf_nfStatusUnsubscribe_rx_requests_total	Number of NfStatusUnsubscribe requests received	NrfLevel, OriginatorNfType	
ocnrf_nfStatusUnsubscribe_tx_responses_total	Number of NfStatusUnsubscribe responses sent	NrfLevel, OriginatorNfType, HttpStatusCode	
ocnrf_nfStatusNotify_tx_requests_total	Number of NfStatusNotify requests sent	NrfLevel, NotificationEventType, TargetNfType	
ocnrf_nfStatusNotify_rx_responses_total	Number of NfStatusNotify responses received	NrfLevel, NotificationEventType, TargetNfType, HttpStatusCode	

Table 6-1 (Cont.) NRF Common Metrics

Metric	Metric Details	Dimensions	Notes
ocnrf_nfDiscover_rx_requests_total	Number of NfDiscover Requests received	NrfLevel, TargetNfType, RequesterNfType	
ocnrf_nfDiscover_tx_responses_total	Number of NfDiscover responses sent	NrfLevel, TargetNfType, RequesterNfType, HttpResponseCode	
ocnrf_nfDiscover_rx_requests_success_perService_total	Number of NfDiscover requests received and processed successfully per Service	NrfLevel, RequesterNfType, ServiceName	
ocnrf_nfDiscover_profiles_discovered_count_total	Number of Profiles returned in discovery response	NrfLevel, TargetNfType, BucketSize	
ocnrf_active_registrations_count	Number of active registered NFs at any point of time	NfType, NrfLevel	
ocnrf_message_processing_time_seconds	Time taken by NRF to process messages (NfRegister/NfUpdate/ NfDelete/ fProfileRetrieval/ NfListRetrieval/ NfHeartbeat/ NfDiscover/ NfSubscribe/ NfUnsubscribe)	NrfLevel, OriginatorNfType, ServiceOperation	Latency calculated by this metric doesn't include time taken by OCNRF API gateway

In the above NRF Metrics table, 4xx and 5xx are the error codes in REST API. For the details about the error codes, refer to 4xx and 5xx Error Codes

Table 6-2 NF Screening specific metrics

Metric	Metric Details	Service Operation	Dimensions
ocnrf_nfScreening_callbackUri_requestFailed_total	The total number of requests for which screening failed against Callback URI screening list.	NFRegister, NFUpdate	NRF level NF type
ocnrf_nfScreening_callbackUri_requestRejected_total	The total number of requests rejected because screening failed against Callback URI screening list.	NFRegister, NFUpdate	NRF level NF type

Table 6-2   (Cont.) NF Screening specific metrics

Metric	Metric Details	Service Operation	Dimensions
ocnrf_nfScreening_nfFqdn_request Failed_total	The total number of requests for which screening failed against NF FQDN screening list.	NFRegister, NFUpdate	NRF level NF type
ocnrf_nfScreening_nfFqdn_request Rejected_total	The total number of requests rejected because screening failed against NF FQDN screening list.	NFRegister, NFUpdate	NRF level NF type
ocnrf_nfScreening_nfIpEndPoint_r equestFailed_total	The total number of requests for which screening failed against NF IP endpoint screening list.	NFRegister, NFUpdate, NFSubscribe	NRF level NF type
ocnrf_nfScreening_nfIpEndPoint_r equestRejected_total	The total number of requests rejected because screening failed against NF IP endpoint screening list.	NFRegister, NFUpdate, NFSubscribe	NRF level NF type
ocnrf_nfScreening_nfTypeRegister _requestFailed_total	The total number of NFRegister requests rejected as NF type was not allowed to register with NRF.	NFRegister, NFUpdate	NRF level NF type
ocnrf_nfScreening_nfTypeRegister _requestRejected_total	The total number of NFRegister requests for which screening failed against NF type screening list.	NFRegister, NFUpdate	NRF level NF type
ocnrf_nfScreening_notApplied_Int ernalError_total	The total number of times screening not applied due to internal error.	NFRegister	NRF level NF type
ocnrf_nfScreening_plmnId_request Failed_total	The total number of requests for which screening failed against PLMN id screening list.	NFRegister	NRF level NF type
ocnrf_nfScreening_plmnId_request Rejected_total	The total number of requests rejected because screening failed against PLMN id screening list.	NFRegister, NFUpdate, NFSubscribe	NRF level NF type

> **Note:**
>
> In the above NF Screening metrics table, the dimension NF Type is a requester NF Type.

OCNRF KPIs

This section includes information about KPIs for Oracle Communications Network Repository Function (OCNRF).

> **Note:**
>
> Sample OCNRF dashboard for Grafana is delivered to the customer through OCNRF Custom Templates. Metrics and functions used to achieve KPI are already covered in OCNRF Custom Templates. Please view the Oracle Help Center site for the information about OCNRF Custom Templates.

Table 6-3   OCNRF KPIs

KPI Details	Metric Used to Achieve this KPI	Service Operation	KPI	Response Code	Notes
Requests/sec	envoy_http_ingress_http_rq_total	All	NRF Ingress Request Rate	Not Applicable	
No of Requests of each type/No of Successful Responses per second	envoy_cluster_cluster_ocnrf_nfregistration_8080_upstream_rq_201	NFRegister	Creates Registration	201	
	envoy_cluster_cluster_ocnrf_nfregistration_8080_upstream_rq_200	NFUpdate	Updates Registration	200	
	envoy_cluster_cluster_ocnrf_nfregistration_8080_upstream_rq_204	NFDeregister	Deletes Registration	204	NFDeregister and NFUpdate (heartbeat) are counted together
	envoy_cluster_cluster_ocnrf_nfsubscription_8080_upstream_rq_201	NFStatusSubscribe	Creates Subscription	201	
	envoy_cluster_cluster_ocnrf_nfsubscription_8080_upstream_rq_204	NFStatusUnsubscribe	Deletes Subscription	204	
	envoy_cluster_cluster_ocnrf_nfdiscovery_8080_upstream_rq_200	NFDiscover	Success Discovery	200	
No of Requests of each type/No of 4xx responses	envoy_cluster_cluster_ocnrf_nfregistration_8080_upstream_rq_4xx	NFRegister/ NFUpdate/ NFDeregister	4xx	4xx	

Table 6-3   (Cont.) OCNRF KPIs

KPI Details	Metric Used to Achieve this KPI	Service Operation	KPI	Response Code	Notes
	envoy_cluster_cluster_ocnrf_nfsubscription_8080_upstream_rq_4xx	NFStatusSubscribe/ NFStatusUnsubscribe	4xx	4xx	
	envoy_cluster_cluster_ocnrf_nfdiscovery_8080_upstream_rq_4xx	NFDiscover	4xx	4xx	
No of Requests of each type/No of 5xx responses	envoy_cluster_cluster_ocnrf_nfregistration_8080_upstream_rq_5xx	NFRegister/ NFUpdate/ NFDeregister	5xx	5xx	
	envoy_cluster_cluster_ocnrf_nfsubscription_8080_upstream_rq_5xx	NFStatusSubscribe/ NFStatusUnsubscribe	5xx	5xx	
	envoy_cluster_cluster_ocnrf_nfdiscovery_8080_upstream_rq_5xx	NFDiscover	5xx	5xx	

OCNRF Alerts

This section includes information about alerts for OCNRF.

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

Table 6-4   OCNRF Alert Details

Alert	Severity	Alert Details
Alert if Ingress traffic reaches 95% of max MPS	Critical	Traffic Rate is above critical threshold
Alert if Ingress traffic reaches 90% of max MPS	Major	Traffic Rate is above major threshold
Alert if Ingress traffic reaches 80% of max MPS	Minor	Traffic Rate is above minor threshold
Alert if error rate exceeds 0.1% of the total transactions	Warning	Transaction Error rate is above 0.1 Percent of Total Transactions
Alert if error rate exceeds1% of the total transactions	Warning	Transaction Error rate is above 1 Percent of Total Transactions
Alert if error rate exceeds 10% of the total transactions	Minor	Transaction Error rate is above 10 Percent of Total Transactions
Alert if error rate exceeds 25% of the total transactions	Major	Transaction Error rate is above 25 Percent of Total Transactions

ORACLE

Table 6-4   (Cont.) OCNRF Alert Details

Alert	Severity	Alert Details
Alert if error rate exceeds 50% of the total transactions	Critical	Transaction Error rate is above 50 Percent of Total Transactions

OCNRF Alert Configuration

OCNRF Alert Configuration

Follow the steps below for OCNRF Alert configuration in Prometheus:

> **Note:**
>
> 1. By default `Namespace` for OCNRF is `ocnrf` that must be update as per the deployment.
>
> 2. The `ocnrf-config-1.2.0.0.0.zip` file can be downloaded from OHC. Unzip the `ocnrf-config-1.2.0.0.0.zip` package after downloading to get `NrfAlertrules.yaml` file.

1. Take Backup of current configuration map of Prometheus:

   ```
   kubectl get configmaps _NAME_-server -o yaml -n _Namespace_ > /tmp/
   tempConfig.yaml
   ```

2. Check and Add NRF Alert file name inside Prometheus configuration map:

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

3. Update configuration map with updated file name of NRF alert file:

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

4. Add NRF Alert rules in configuration map under file name of NRF alert file:

   ```
   kubectl patch configmap _NAME_-server -n _Namespace_--type merge --patch
   "$(cat ~/NrfAlertrules.yaml)"
   ```

> **Note:**
>
> The Prometheus server takes an updated configuration map that is automatically reloaded after approximately 60 seconds. Refresh the Prometheus GUI to confirm that the NRF Alerts have been reloaded.

A

4xx and 5xx Error Codes

This appendix provides the list of 4xx and 5xx error codes.

4xx: Client Error

Message	Description
400 Bad Request	The server did not understand the request.
403 Forbidden	Access is forbidden to the requested page.
404 Not Found	The server can not find the requested page.
405 Method Not Allowed	The method specified in the request is not allowed.
409 Conflict	The request could not be completed because of a conflict.
415 Unsupported Media Type	The server will not accept the request, because the mediatype is not supported.

5xx: Server Error

Message	Description
500 Internal Server Error	The request was not completed. The server met an unexpected condition.
501 Not Implemented	The request was not completed. The server did not support the functionality required.
502 Bad Gateway	The request was not completed. The server received an invalid response from the upstream server.
503 Service Unavailable	The request was not completed. The server is temporarily overloading or down.
504 Gateway Timeout	The gateway has timed out.
505 HTTP Version Not Supported	The server does not support the "http protocol" version.