Oracle® Communications

Cloud Native Core, Network Repository Function REST Specification Guide

Release 23.4.6
F89026-16
March 2025

Oracle Communications Cloud Native Core, Network Repository Function REST Specification Guide, Release 23.4.6

F89026-16

Contents

My Oracle Support

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

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

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

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

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

Acronyms

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

Table Acronyms

Term	Definition
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.
CBCF	Cell Broadcast Center Function
CNE	Cloud Native Environment
CDS	Cache data Service. Caching Microservice which is responsible for syncing and caching the data.
Dimension	Dimension is a tag of Metric filter. For Example, "ocnrf_nfRegister_rx_requests_total {{ OriginatorNfType }} {{NrfLevel }} {{NfInstanceId }}" In the example above, OriginatorNfType, NrfLevel, and NfInstanceId are dimensions.
DNS	Domain Name System
DRA	Diameter Routing Agent
FQDN	Fully Qualified Domain Name
ICSCF	Interrogating Call Session Control Function
IMS_AS	IP Multimedia Subsystem Application Server
K8s	Kubernetes
KPI	Key Performance Indicator
MME	Mobility Management Entity
MMI	Machine Machine Interface
MPS	Messages Per Second
NDB	Network Database
NF	Network Function
NSSAAF	Network Slice-Specific Authentication and Authorization 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 (For Example, compute, storage, and networking resources) which form a deployed Network Slice.

Table (Cont.) Acronyms

Term	Definition
NF Consumer	A generic way to refer to an NF which consumes services provided by another NF. For Example: An AMF acts as a Consumer NF that 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. For Example: A PCF acts as a Producer NF that provides AMPolicy Services to the AMF.
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.
SCP	Service Communication Proxy
SCEF	Service Capability Exposure Function
SCSAS	Security Assurance Specification
SEPP	Security Edge Protection Proxy
SLF	Subscriber Location Function
SMF	Session Management Function
SOR_AF	Steering of Roaming Application Function
SPAF	Service Provider Application Function
URI	Uniform Resource Identifier
UCMF	UE Capability Management Function

What's New in This Guide

This section lists the documentation updates for Release 23.4.x.

Release 23.4.6- F89026-16, March 2025

Updated the `abatementValue` from 180 to 280 for L2 in the `ocnrf-nfdiscovery` service name in the Overload Level Threshold Configuration in Overload Level Threshold Configuration in Perf-Info section.

Release 23.4.6- F89026-15, January 2025

- Updated the description and the default value of the `allowDuplicateSubscriptions` attribute in the NF Management Options section.
- Updated the description of the `nrfSupportForProfileChangesInResponse` attribute in the NF Management Options section.
- Updated the `audienceType` attribute description in the NF Access Token Options section.

Release 23.4.5- F89026-14, October 2024

- Updated the value of `monitoringInterval` from 100 to 1200 in the sample body in the Pod Protection Options section.
- Updated the default value of `monitoringInterval` from 100 to 1200 in the Pod Protection Options section.
- Updated the Query Example with API for the following attributes in the Query Result Attributes Supported by Subscription-Details URI table in the NRF State Data Retrieval APIs section.
  - reqNfFqdn
  - nfStatusNotificationUri
  - validityTime
- Updated the description for the following attributes in the CCA Header Validation section.
  - `enabled`
  - `minExpiryTime`
  - `subKey`
  - `validationRule`

Release 23.4.4- F89026-13, August 2024

There are no changes made to this document in this release.

Release 23.4.3- F89026-12, July 2024

There are no changes made to this document in this release.

Release 23.4.2- F89026-11, July 2024

- Modified the description for the `nfNotifyloadThreshold` parameter in the NF Management Options section.

- Updated the value of the `samplingPeriod` attribute from 6000 to 200 in the Policy Mapping Configuration in Ingress Gateway section.

Release 23.4.0- F89026-09, July 2024

- Modified the description for the `nfNotifyloadThreshold` parameter in the NF Management Options section.
- Updated the value of the `samplingPeriod` attribute from 6000 to 200 in the Policy Mapping Configuration in Ingress Gateway section.

Release 23.4.0- F89026-06, May 2024

- Added the L4 level values for `svc_failure_count` and `memory` metrics under `ocnrf-nfregistration` service in the Overload Level Threshold Configuration in Perf-Info section.
- Updated the description for the `nfNotifyloadThreshold` parameter in the NF Management Options section.

Release 23.4.0- F89026-03, February 2024

- Added a note about static and virtual host configuration in the Peer Configuration section.

Release 23.4.0- F89026-02, January 2024

- Added `healthapipath` parameter in the Peer Configuration section.

Release 23.4.0- F89026-01, December 2023

- Added the configuration details for the NRF Growth feature in the following sections:
  - NRF Growth Options
  - Forwarding Options for NRF Growth
- Added a new API for Cache Data Service logging in the Logging Level Options section.
- Updated the URI for alternate-route logging in the Logging Level Options section.
- Updated the description of the `requestRetryDetails.featureStatus` parameter in the NF Management Options section.
- Restructured the Common Services REST APIs chapter to categorized the APIs based on common services.

1 Introduction

This document provides information on how to configure the services and manageable objects in (NRF) using REST API.

NRF is a key component of the 5G Service Based Architecture. 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 is expected to be instantiated, scaled, and terminated with minimal to no manual intervention. In addition to serving as a repository of the services, NRF also supports discovery mechanisms that allow NFs to discover each other and get the updated status of the desired NFs.

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 provider's NF instances in the 5G core network.
- Allows NF instances to track the status of other NF instances.
- Provides Oauth2 based Access Token service for consumer NF authorization.
- Provides specific NF Type selection based on subscriber identity.
- Supports forwarding of messages from one NRF to another NRF.
- Supports georedundancy to ensure service availability.

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

- Management Service
- Discovery Service
- AccessToken Service

> **Note**
>
> The performance and capacity of the NRF system may vary based on the call model, Feature or Interface configuration, and underlying CNE and hardware environment.

1.1 References

Following are the reference documents:

- *Oracle Communications Cloud Native Core, Cloud Native Environment Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*
- *Oracle Communications Cloud Native Core, Network Repository Function User Guide*

- *Oracle Communications Cloud Native Configuration Console User Guide*

2
NRF REST Specifications

This chapter provides information about REST specifications used in Oracle Communications Cloud Native Core, Network Repository Function (NRF).

NRF can be configured using Helm configurations, REST APIs, and Cloud Native Configuration Console (CNC Console). The NRF deployment configurations are performed during NRF installation using Helm and a few configurations are modified using REST APIs. REST configurations can also be performed using the CNC Console.

For Helm configurations, see *Oracle Communications Cloud Native Core, Network Repository Function Installation, Upgrade, and Fault Recovery Guide*.

For the configurations using CNC Console, see *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

2.1 Mandatory Configurations

Following are the mandatory parameters that must be configured before using NRF:

- nrfPlmnList: PLMN(s) served by NRF.
- ocnrfHost: NRF Host's FQDN.
- ocnrfPort: NRF Host's Port.

For configuring the mandatory parameters, see General Options.

> **Note**
>
> M, O, and C in the Presence column denote as follows:
>
> - M: Mandatory
> - O: Optional
> - C: Conditional

2.2 Service API Interfaces

This section lists the API interface details for each NRF services.

API details

apiRoot is concatenation of the following parts:

- scheme:- http, https
- the fixed string "://"
- authority (host and optional port) host and port will be CNC Console host and port details

Table 2-1 Service API Interfaces

Resource Name	Resource URI	HTTP Method	Data Model for Request	Data Model for Response	Description
ocnrfConfigurations	{apiRoot}/nrf-configuration/v1/	GET	Not applicable	ocnrfConfigurations	Retrieves all of the NRF configurations in single GET request. This includes all Options and NFScreeningRules.
generalOptions	{apiRoot}/nrf-configuration/v1/generalOptions	GET	Not applicable	generalOptions	Retrieves NRF general options configuration.
generalOptions	{apiRoot}/nrf-configuration/v1/generalOptions	PUT	generalOptions	generalOptions	Updates NRF general options configuration.
nfScreeningOptions	{apiRoot}/nrf-configuration/v1/nfScreeningOptions	GET	Not applicable	nfScreeningOptions	Retrieves NF Screening options configuration.
nfScreeningOptions	{apiRoot}/nrf-configuration/v1/nfScreeningOptions	PUT	nfScreeningOptions	nfScreeningOptions	Updates NF Screening options configuration.
nfManagementOptions	{apiRoot}/nrf-configuration/v1/nfManagementOptions	GET	Not applicable	nfManagementOptions	Retrieves NRF Management options configuration.
nfManagementOptions	{apiRoot}/nrf-configuration/v1/nfManagementOptions	PUT	nfManagementOptions	nfManagementOptions	Updates NRF Management options configuration.
nfDiscoveryOptions	{apiRoot}/nrf-configuration/v1/nfDiscoveryOptions	GET	Not applicable	nfDiscoveryOptions	Retrieves NRF Discovery options configuration.
nfDiscoveryOptions	{apiRoot}/nrf-configuration/v1/nfDiscoveryOptions	PUT	nfDiscoveryOptions	nfDiscoveryOptions	Updates NRF Discovery options configuration.
nfAccessTokenOptions	{apiRoot}/nrf-configuration/v1/nfAccessTokenOptions	GET	Not applicable	nfAccessTokenOptions	Retrieves NRF Access Token options configuration.
nfAccessTokenOptions	{apiRoot}/nrf-configuration/v1/nfAccessTokenOptions	PUT	nfAccessTokenOptions	nfAccessTokenOptions	Updates NRF Access Token options configuration.
forwardingOptions	{apiRoot}/nrf-configuration/v1/forwardingOptions	GET	Not applicable	forwardingOptions	Retrieves NRF Forwarding options configuration.
forwardingOptions	{apiRoot}/nrf-configuration/v1/forwardingOptions	PUT	forwardingOptions	forwardingOptions	Updates NRF Forwarding options configuration.

Table 2-1 (Cont.) Service API Interfaces

Resource Name	Resource URI	HTTP Method	Data Model for Request	Data Model for Response	Description
slfOptions	{apiRoot}/nrf-configuration/v1/slfOptions	GET	Not applicable	slfOptions	Retrieves NRF SLF options configuration.
slfOptions	{apiRoot}/nrf-configuration/v1/slfOptions	PUT	slfOptions	slfOptions	Updates NRF SLF options configuration.
slfOptions	{apiRoot}/nrf-configuration/v1/slfOptions	PATCH	slfOptions	slfOptions	Partially updates specific NRF SLF options configuration.
geoRedundancyOptions	{apiRoot}/nrf-configuration/v1/geoRedundancyOptions	GET	Not applicable	geoRedundancyOptions	Retrieves NRF Georedundancy options configuration.
geoRedundancyOptions	{apiRoot}/nrf-configuration/v1/geoRedundancyOptions	PUT	geoRedundancyOptions	geoRedundancyOptions	Updates NRF Georedundancy options configuration.
dnsNAPTRUpdateOptions	{apiRoot}/nrf-configuration/v1/dnsNaptrUpdateOptions	GET	dnsNAPTRUpdateOptions	dnsNAPTRUpdateOptions	Retrieves NAPTR record from DNS configuration.
dnsNAPTRUpdateOptions	{apiRoot}/nrf-configuration/v1/dnsNaptrUpdateOptions	PUT	dnsNAPTRUpdateOptions	dnsNAPTRUpdateOptions	Updates NAPTR record in DNS configuration.
podProtectionOptions	{apiRoot}/nrf-configuration/v1/nfSubscription/podProtectionOptions	GET	podProtectionOptions	Pod Protection Options	Retrieves NRF pod protection options configuration.
podProtectionOptions	{apiRoot}/nrf-configuration/v1/nfSubscription/podProtectionOptions	PUT	podProtectionOptions	Pod Protection Options	Enables or Disables NRF pod protection feature.
nfAuthenticationOptions	{apiRoot}/nrf-configuration/v1/nfAuthenticationOptions	GET	Not applicable	nfAuthenticationOptions	Retrieves NRF Authentication options configuration.
nfAuthenticationOptions	{apiRoot}/nrf-configuration/v1/nfAuthenticationOptions	PUT	nfAuthenticationOptions	nfAuthenticationOptions	Updates NRF Authentication options configuration.
logLevelOptions	{apiRoot}/nrf-configuration/v1/nfAccessToken/logging	GET	Not applicable	logging	Retrieves NRF Log Level options related to nfAccessToken configuration.
logLevelOptions	{apiRoot}/nrf-configuration/v1/nfAccessToken/logging	PUT	logging	logging	Updates NRF Log Level options related to nfAccessToken configuration.

Table 2-1 (Cont.) Service API Interfaces

Resource Name	Resource URI	HTTP Method	Data Model for Request	Data Model for Response	Description
logLevelOptions	{apiRoot}/nrf-configuration/v1/nfDiscovery/logging	GET	Not applicable	logging	Retrieves NRF Log Level options related to nfDiscovery configuration.
logLevelOptions	{apiRoot}/nrf-configuration/v1/nfDiscovery/logging	PUT	logging	logging	Updates NRF Log Level options related to nfDiscovery configuration.
logLevelOptions	{apiRoot}/nrf-configuration/v1/nfRegistration/logging	GET	Not applicable	logging	Retrieves NRF Log Level options related to nfRegistration configuration.
logLevelOptions	{apiRoot}/nrf-configuration/v1/nfRegistration/logging	PUT	logging	logging	Updates NRF Log Level options related to nfRegistration configuration.
logLevelOptions	{apiRoot}/nrf-configuration/v1/nfSubscription/logging	GET	Not applicable	logging	Retrieves NRF Log Level options related to nfSubscription configuration.
logLevelOptions	{apiRoot}/nrf-configuration/v1/nfSubscription/logging	PUT	logging	logging	Updates NRF Log Level options related to nfSubscription configuration
logLevelOptions	{apiRoot}/nrf-configuration/v1/nrfAuditor/logging	GET	Not applicable	logging	Retrieves NRF Log Level options related to nrfAuditor configuration.
logLevelOptions	{apiRoot}/nrf-configuration/v1/nrfAuditor/logging	PUT	logging	logging	Updates NRF Log Level options related to nrfAuditor configuration.
logLevelOptions	{apiRoot}/nrf-configuration/v1/nrfConfiguration/logging	GET	Not applicable	logging	Retrieves NRF Log Level options related to nrfConfiguration configuration.
logLevelOptions	{apiRoot}/nrf-configuration/v1/nrfConfiguration/logging	PUT	logging	logging	Updates NRF Log Level options related to nrfConfiguration configuration.
logLevelOptions	{apiRoot}/nrf/nf-common-component/v1/igw/logging	GET	Not applicable	logging	Retrieves NRF Log Level options related to Ingress Gateway configuration.
logLevelOptions	{apiRoot}/nrf/nf-common-component/v1/igw/logging	PUT	Not applicable	logging	Updates NRF Log Level options related to Ingress Gateway configuration.
logLevelOptions	{apiRoot}/nrf/nf-common-component/v1/egw/logging	GET	Not applicable	logging	Retrieves NRF Log Level options related to Egress Gateway configuration.

Table 2-1 (Cont.) Service API Interfaces

Resource Name	Resource URI	HTTP Method	Data Model for Request	Data Model for Response	Description
logLevelOptions	{apiRoot}/nrf/nf-common-component/v1/egw/logging	PUT	Not applicable	logging	Updates NRF Log Level options related to Egress Gateway configuration.
logLevelOptions	{apiRoot}/nrf/nf-common-component/v1/appinfo/logging	GET	Not applicable	logging	Retrieves NRF Log Level options related to App Info configuration.
logLevelOptions	{apiRoot}/nrf/nf-common-component/v1/appinfo/logging	PUT	Not applicable	logging	Updates NRF Log Level options related to App Info configuration.
allLoggingOptions	{apiRoot}/nrf-configuration/v1/all/logging	GET	Not applicable	array(allLogging Options)	Returns logging options for all NRF microservices and common services.
nrfGrowth	{apiRoot}/nrf-configuration/v1/nrfGrowth/featureOptions	GET	featureOptions	NRF Growth Options	Retrieves NRF growth feature configuration.
nrfGrowth	{apiRoot}/nrf-configuration/v1/nrfGrowth/featureOptions	PUT	featureOptions	NRF Growth Options	Updates NRF growth feature configuration.
nrfForwardingOptions	{apiRoot}/nrf-configuration/v1/nrfGrowth/nrfForwardingOptions	GET	nrfForwardingOptions	Forwarding Options for NRF Growth	Retrieves NRF growth feature forwarding configuration.
nrfForwardingOptions	{apiRoot}/nrf-configuration/v1/nrfGrowth/nrfForwardingOptions	PUT	nrfForwardingOptions	Forwarding Options for NRF Growth	Updates NRF growth feature forwarding configuration.
PeerConfiguration	{apiRoot}/nrf/nf-common-component/v1/egw/peerconfiguration	PUT	Not applicable	array (PeerConfiguration)	Updates NRF Egress Peer configuration.
PeerConfiguration	{apiRoot}/nrf/nf-common-component/v1/egw/peerconfiguration	GET	Not applicable	array (PeerConfiguration)	Retrieves NRF Egress Peer configuration.
PeerConfiguration	{apiRoot}/nrf/nf-common-component/v1/egw/peerconfiguration	PATCH	Not applicable	array (PeerConfiguration)	Modifies NRF Egress Peer Set configuration.

Table 2-1 (Cont.) Service API Interfaces

Resource Name	Resource URI	HTTP Method	Data Model for Request	Data Model for Response	Description
PeerSetConfiguration	{apiRoot}/nrf/nf-common-component/v1/egw/peersetconfiguration	PUT	Not applicable	array (PeerSetConfiguration)	Updates NRF Egress Peer Set configuration.
PeerSetConfiguration	{apiRoot}/nrf/nf-common-component/v1/egw/peersetconfiguration	GET	Not applicable	array (PeerSetConfiguration)	Retrieves NRF Egress Peer Set configuration.
PeerSetConfiguration	{apiRoot}/nrf/nf-common-component/v1/egw/peersetconfiguration	PATCH	Not applicable	array (PeerSetConfiguration)	Modifies a set of NRF Egress Peer Set configuration.
RoutesConfiguration	{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration	PUT	Not applicable	array (RoutesConfiguration)	Updates NRF Egress routing configuration.
RoutesConfiguration	{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration	GET	Not applicable	array (RoutesConfiguration)	Retrieves NRF Egress routing configuration.
RoutesConfiguration	{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration	PATCH	Not applicable	array (RoutesConfiguration)	Modifies a set of NRF Egress routing configurations.
roamingOptions	{apiRoot}/nrf-configuration/v1/roamingOptions	GET	Not applicable	roamingOptions	Retrieves NRF roaming configuration details.
roamingOptions	{apiRoot}/nrf-configuration/v1/roamingOptions	PUT	roamingOptions	roamingOptions	Updates NRF roaming configuration details.
controlledShutdownOptions	{apiRoot}/nrf-configuration/v1/controlledShutdownOptions	GET	Not applicable	Controlled Shutdown Options	Retrieves the operational state.
controlledShutdownOptions	{apiRoot}/nrf-configuration/v1/controlledShutdownOptions	PUT	Not applicable	Controlled Shutdown Options	Updates the operational state.

Table 2-1 (Cont.) Service API Interfaces

Resource Name	Resource URI	HTTP Method	Data Model for Request	Data Model for Response	Description
screening-rules	{apiRoot}/nrf-configuration/v1/screening-rules	GET	Not applicable	ScreeningRulesResult	Returns all the screening rules.
screening-rules	{apiRoot}/nrf-configuration/v1/screening-rules	GET	nfScreeningRulesListType or/and nfScreeningRulesListStatus	ScreeningRulesResult	Returns screening rules corresponding to the specified NF Screening Rule List Type. Query:- {apiRoot}/nrf-configuration/v1/screening-rules?nfScreeningRulesListStatus=<NfScreeningRulesListStatus>&nfScreeningRulesListType=<NfScreeningRulesListType>
screening-rules	{apiRoot}/nrf-configuration/v1/screening-rules/{nfScreeningRulesListType}	PUT	NfScreeningRules	NfScreeningRules	Replaces the complete specified NF Screening Rule List Type.
screening-rules	{apiRoot}/nrf-configuration/v1/screening-rules/{nfScreeningRulesListType}	PATCH	PatchDocument	NfScreeningRules	Partially updates the specified NF Screening Rule List Type (except read-only attributes).

2.2.1 Responses Supported by Service API Interfaces

Table 2-2 Response Body

Data Type	Presence	Cardinality	Response Codes	Description
ProblemDetails	C	1	500 Internal Server Error	Internal error occurred while processing the service API.
ProblemDetails	C	1	400 Bad Request	JSON body sent by the client is not correct according to the data model defined.
As per Data Model Defined	C	1	200 OK	Response body contains all the stored values from NRF.

2.2.2 Common Data Types

Common data types

Table 2-3 Common Data Types

DataType	Reference
NFType	3GPP TS 29.510
NFServiceVersion	3GPP TS 29.510
UriScheme	3GPP TS 29.510
Fqdn	3GPP TS 29.510
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

Table 2-4 NfConfig

Attribute	DataType	Presence	Description
apiVersions	array (NFServiceVersion)	M	API Version of NF
scheme	UriScheme	M	URI schema supported by NF
host	string	M	Host of NF
port	integer	O	Port of NF **Default value**: 80, if the scheme is HTTP 443, if the scheme is HTTPS
apiPrefix	string	O	ApiPrefix
priority	integer	M	Priority of NF
nfInstanceId	string	M	NF Instance Id of NF

Table 2-5 ErrorInfo

Attribute	DataType	Presence	Description
errorCondition	ErrorCondition	ReadOnly	Error Conditions for each configuration. See specific configuration sections for error conditions.
responseCode	integer	M	This response code is used when the corresponding error condition occurs.

Table 2-5 (Cont.) ErrorInfo

Attribute	DataType	Presence	Description
errorResponse	string	M	This response description is used when the corresponding error condition occurs.
retryAfter	string	C	The attribute indicates the time interval after which the NF retry the request. retryAfter header is added only for responseCodes - 503, 413, 429, 3xx. The value is in pHqMrS format. Where p,q,r are integers and H,M,S or h,m,s denote hours, minutes & seconds respectively. Range: 60s-1h Default Value: 5m
redirectUrl	string	C	The attribute indicates the NF to redirect its request to this URI. Location header in redirectUrl is added only for responseCodes - 3xx. redirectUrl must be in URI format. It is mandatory to configure redirectUrl when responseCode is configured.

Table 2-6 ResponseHttpStatusCodes

Attribute	DataType	Description
pattern	string	It is a regular expression that provides a mechanism to select specific strings from a set of character strings. Sample: "pattern":"^[3,5][0-9]{2}$"
codeList	array (integer)	It contains a list of HTTP response status codes. Sample: "codeList": [404,400]

> **Note**
>
> Either pattern or codeList must be present.

Table 2-7 ScreeningRulesResult

Attribute name	DataType	Presence	Cardinality	Description
nfScreeningRulesList	array (NfScreeningRules)	M	0..N	It contains an array of NF Screening Rules List. An empty array means NF Screening list is not configured.

2.3 General Options

This section provides REST API configuration parameter details to configure NRF general options.

URI: *{apiRoot}/nrf-configuration/v1/generalOptions*

Method: PUT and GET

- **PUT**: Updates NRF general options configuration.
- **GET**: Retrieves NRF general options configuration.

Content Type: application/json

Body:

```
{
    "nrfPlmnList": [{
        "mcc": "310",
        "mnc": "14"
    }],
    "ocnrfHost": "ocnrf-ingressgateway.ocnrf.svc.cluster.local",
    "ocnrfPort": 80,
    "ocnrfScheme": "http",
    "enableF3": true,
    "enableF5": true,
    "maximumHopCount": 3,
    "defaultLoad": 5,
    "defaultPriority": 100,
    "defaultPriorityAssignment": false,
    "defaultLoadAssignment": false,
    "add3gppSbiCorrelationInfoHeader": "ENABLED",
    "ocnrfUserAgentHeader": ""
}
```

Configuration Attributes

> **Note**
>
> - If any attribute is not present in the JSON request body while updating, the existing value in the database is preserved and used. At least one attribute is included during the PUT request.
> - nrfPlmnList, ocnrfHost, and ocnrfPort are mandatory values that must be configured before using NRF.

Table 2-8 Configuration Attributes for GeneralOptions

Parameter	Description	Details
`nrfPlmnList`	This value contains at least one PLMN supported by NRF, and this value must be set before using NRF.	**DataType**: array (PlmnId) **Constraints**: NA **Default Value**: See PLMN ID.

`ocnrfHost`	ocnrfHost needs to be NRF's external routable FQDN (for example, ocnrf.oracle.com) OR external routable IpAddress (for example, 10.75.212.60) OR for routing within the same Kubernetes cluster use full NRF Ingress Gateway's Service FQDN as below format:
	`<helm-releasename>ingressgateway.<namespace>.svc.<cluster-domainname>`
	Example:
	`ocnrfingressgateway.nrf-1.svc.cluster.local`
	where,
	helm-releasename: the helm release name (deployment name that will be used during "helm install").
	namespace: the namespace in which NRF is deployed.
	cluster-domainname: the Kubernetes dnsDomain name (dnsDomain can be found using `kubectl -n kube-system get configmap kubeadmconfig -o yaml | grep -i dnsDomain`).
	This value is used in UriList of NfListRetrieval Service Operation response.
	Note: The value of this attribute can be FQDN, IPv4 or IPv6.
		DataType: string
		Constraints: None
		Default Value: ocnrf-ingressgateway.ocnrf.svc.cluster.local

`ocnrfPort`	Indicates the NRF Host's Port	**DataType**: integer **Constraints**: None **Default Value**: 80
`ocnrfScheme`	Indicates the NRF Host's Scheme	**DataType**: string **Constraints**: http or https **Default Value**: http
`enableF3`	Indicates the specification to which NRF is compliant. If this flag is set to `true`, NRF functions as per 3GPP TS 29510 v15.3 specification. If it is set to false, NRF functions as per 3GPP TS 29510 v15.2.	**DataType**: boolean **Constraints**: true or false **Default Value**: true

Table 2-8 (Cont.) Configuration Attributes for GeneralOptions

Parameter	Description	Details
`enableF5`	Indicates the specification to which NRF is compliant. If this flag is set to `true`, NRF functions as per 3GPP TS 29510 v15.5 specification. If it is set to false, NRF functions as per 3GPP TS 29510 v15.2 or v15.3 specification (depends on enableF3 flag).	**DataType**: boolean **Constraints**: true or false **Default Value**: true
`defaultLoad`	defaultLoad value is set in NF load attribute of NFProfile, if `defaultLoadAssignment` attribute is set to `true`. In case NFProfile does not have load attribute, this value is sent in NFDiscover response and NFProfile in NFNotify operation.	**DataType**: integer **Constraints**: 0 - 100 **Default Value**: 5
`defaultLoadAssignment`	Value of default NF load is set in NF Load attribute of NFProfile while sending in NFDiscover response and NFProfile sent in NFNotify operation, in case NFProfile does not have load attribute.	**DataType**: boolean **Constraints**: true or false **Default Value**: false
`defaultPriority`	This attribute is default value of NF Priority and will be used if NFProfile does not have priority attribute set by NF. This attribute value is set in NF Priority of NFProfile, if `defaultPriorityAssignment` attribute is set to `true`.	**DataType**: integer **Constraints**: 0 - 65535 **Default Value**: 100
`defaultPriorityAssignment`	Value of default NF Priority is set in NF Priority attribute of NFProfile while sending in NFDiscover response and NFProfile sent in NFNotify operation, in case NFProfile does not have Priority attribute.	**DataType**: boolean **Constraints**: true or false **Default Value**: false
`maximumHopCount`	Indicates the maximum number of nodes with which NRF can communicate for providing service for a request.	**DataType**: integer **Constraints**: 1 - 5 **Default Value**: 3
`add3gppSbiCorrelationInfoHeader`	This attribute indicates whether the `3gpp-Sbi-Correlation-Info` can be added to the request. If this flag is set to ENABLED, then `3gpp-Sbi-Correlation-Info` header is added, if not present. This configuration is applicable to NFDiscover service operation only and when 3gpp-Sbi-Correlation-Info header is not coming in the incoming NF Discover service operation request. Along with populate 3gpp-Sbi-Correlation-Info header if not present, value is decided using SUPI and GPSI attributes only from NFDiscover query.	**DataType**: String **Constraints**: ENABLED/ DISABLED **Default Value**: ENABLED

Table 2-8 (Cont.) Configuration Attributes for GeneralOptions

Parameter	Description	Details
ocnrfUserAgentHeader	This attribute indicates the value of the User-Agent header that is added to the outgoing requests for SLF query and NFStatusNotify. The recommended value starts with NRF followed by -<operator specific string/value> Few examples, NRF-<NrfInstanceId>, For example: NRF-4947a69a-f61b-4bc1-b9da-47c9c5d14b64 NRF-< FQDN>, For example: NRF-nrf05.testnetwork.org NRF-<NrfInstanceId> < FQDN>, For example: NRF-4947a69a-f61b-4bc1-b9da-47c9c5d14b64 nrf05.testnetwork.org **Note**: 1. String containing only blank spaces is not supported. However, it supports blank spaces in between. 2. The feature is disabled for an empty string. 3. 3GPP validation is not performed for the configured value.	**Data Type**: String **Constraints**: The length of the value can be 1-300 characters. **Default Value**: Empty String

Table 2-9 PLMN ID

Parameter	Description	Details
mcc	Provides unique Mobile Country Code (MCC).	**Data Type**: string **Default**: 310
mnc	Provides unique Mobile Network Code (MNC).	**Data Type**: string **Default**: 14

> **Note**
>
> For more information on the parameters, see Configuration Attributes for GeneralOptions.

2.4 NF Management Options

This section provides REST API configuration parameter details to configure NRF management options.

URI: *{apiRoot}/nrf-configuration/v1/nfManagementOptions*

Method: PUT and GET

- **PUT**: Updates NRF Management options configuration.
- **GET**: Retrieves NRF Management options configuration.

Content Type: application/json

Body:


```
{
    "nfHeartBeatTimers": [{
            "nfType": "ALL_NF_TYPE",
            "minHbTimer": "30s",
            "maxHbTimer": "5m",
            "defaultHbTimer": "30s",
            "nfHeartBeatMissAllowed": 3
        },
        {
            "nfType": "AMF",
            "minHbTimer": "10s",
            "maxHbTimer": "120s",
            "defaultHbTimer": "20s",
            "nfHeartBeatMissAllowed": 1
        }
    ],
    "nfNotifyLoadThreshold": 5,
    "nrfSupportForProfileChangesInResponse": true,
    "defaultSubscriptionValidityTime": "24h",
    "nrfSupportForProfileChangesInNotification": false,
    "nfProfileSuspendDuration": "168h",
    "acceptAdditionalAttributes": false,
    "allowDuplicateSubscriptions": true,
    "requestRetryDetails": {
        "featureStatus": "DISABLED",
        "retryCount": 3,
        "requestTimeout": 3000,
        "errorResponseCodeList": ["408", "409", "5xx"],
        "exceptionResponseList": ["java.util.concurrent.TimeoutException",
"java.net.ConnectException"]
          },
    "subscriptionLimit": {
        "featureStatus": "DISABLED",
        "globalMaxLimit": 850,
        "rejectSubscriptionRenewalWhenLimitBreached": "ENABLED",
        "errorResponses": [{
            "errorCondition": "Subscription_Global_Limit_Breached",
            "responseCode": 500,
            "errorCause": "INSUFFICIENT_RESOURCES",
            "errorResponse": "Subscription global limit breached",
            "retryAfter": "5m",
            "redirectUrl": ""
        }],
        "limitThresholds": [{
            "level": "WARN",
            "onset": 50,
            "abatement": 45
        }, {
            "level": "MINOR",
            "onset": 60,
            "abatement": 55
        }, {
            "level": "MAJOR",
            "onset": 70,
```

```
            "abatement": 65
        }, {
            "level": "CRITICAL",
            "onset": 90,
            "abatement": 75
        }]
    }
}
```

Configuration Attributes

> **Note**
>
> If any attribute is not present in the JSON request body while updating, the existing value in the database is preserved and used. At least one attribute is included during the PUT request.

Table 2-10 nfManagementOptions

Parameter	Description	Details
`nfHeartbeatTimers`	This attribute is used to configure the heartbeat related information of the NF. It allows configuring the heartbeat information per NFType. By default, `ALL_NF_TYPE` nfType relevant configuration is set for nfHeartbeatTimers.	**DataType**: array (HeartbeatInfo) **Constraints**: See HeartbeatInfo table for details. **Default Value**: NA
`nfNotifyLoadThreshold`	NRF generates the notification trigger when the difference between the last notified load value and the current reported load value is equal or greater than the configured value of `nfNotifyloadThreshold` attribute. **Note**: • NRF generates the notification trigger if the nfProfile level load is added or removed. • This feature applies to nfProfile level load only. NRF triggers a notification for every change in the service level load. • This feature is applicable with NfHeartBeat service operation.	**DataType**: integer **Constraints**: 0 - 99 **Default Value**: 5
`nrfSupportForProfileChangesInResponse`	When this flag is set to true and the nfProfile contains *nfProfileChangesSupportInd* set to true, NRF will send only the changed attributes in the nfProfile in the response. When this value is set to false, the complete nfProfile will be sent in the response.	**DataType**: boolean **Constraints**: true or false **Default Value**: true

Table 2-10 (Cont.) nfManagementOptions

Parameter	Description	Details
`defaultSubscriptionValidityTime`	If the ValidityTime attribute is not received in SubscriptionData during NFStatusSubscribe, this default value is used for the calculation of validity time (current time + default duration). If the ValidityTime attribute is received in SubscriptionData during NFStatusSubscribe, this is the minimum value that is used for validation and limits purposes. It means, if the value provided is less than (current time + minimum possible range value), then the minimum range value is considered as validity time for subscription. Similarly, in case the validity time is more than (current time + maximum possible range value), then the maximum range value is considered as validity time for subscription. The value is in pHqMrS format, where p, q, r are integers and H, M, S or h, m, s denotes hours, minutes, and seconds respectively.	**DataType**: string **Constraints**: 10s - 720h **Default Value**: 24h
`nrfSupportForProfileChangesInNotification`	NRF sends profileChanges attribute instead of NFProfile in the notification if this flag is set to **true**.	**DataType**: boolean **Constraints**: true or false **Default Value**: false
`nfProfileSuspendDuration`	Indicates the duration for which the NF is suspended, before it is deleted from the NRF database. The value is in pHqMrS format, where p, q, r are integers and H, M, S or h, m, s denotes hours, minutes, and seconds respectively.	**DataType**: string **Constraints**: 10s - 744h **Default Value**: 168h
`acceptAdditionalAttributes`	NRF preserves additional attributes that are not defined by 3GPP in NFProfile or NFService based on this attribute value.	**DataType**: boolean **Constraints**: true or false **Default Value**: false

Table 2-10 (Cont.) nfManagementOptions

Parameter	Description	Details
`allowDuplicateSubscriptions`	This attribute specifies if NRF allows creation of duplicate subscriptions. • If this value is set as false, for every subscription create request, all the attributes of subscriptionData, except the validityTime attribute, is checked against all the existing subscriptions to see if there is an exact match: – If this value is set as true, for every subscription request, NRF will create a new subscription without checking if there is already a subscription request present. – If a duplicate subscription is found, NRF returns "`201 Created along with the existing SubscriptionId and the existing validityTime`". By this NRF will be accepting the subscription create request, but will not create a duplicate subscription. – If a duplicate subscription is not found, NRF creates the subscription and returns "`201 Created with the new SubscriptionId`". For more use cases, see "Use Cases for Allow Duplicate Subscriptions"section in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*. **Note:** If the value of `allowDuplicateSubscriptions` is set as false, NRF would check for duplicate subscription by matching the current subscription request with every subscription present in NRF. Hence, this causes a performance degradation of around 50% during NFStatusSubscribe and NfStatusNotify service operation.	**DataType**: boolean **Constraints**: true or false **Default Value**: true
`requestRetryDetails`	This attribute configuration indicates the *NfStatusNotify* service operation request retry. After NRF upgrade from 22.2.x to 22.3.x, if you are enabling notification retry feature, `DefaultRouteRetry` must be configured in Egress Gateway configuration. For more information to enable, see *Oracle Communications Cloud Native Core, Network Repository Function User Guide*	**DataType**: array (Table 2-12) **Constraints**: See Table 2-12 table for details. **Default Value**: NA
`subscriptionLimit`	To configure Subscription Limit feature in NfManagementOptions. For more information, see Subscription Limit.	**DataType**: array () **Constraints**: NA **Default Value**: NA

HeartbeatInfo

Table 2-11 HeartbeatInfo

Attribute	Description	Details
`nfType`	All nftypes supported in 3GPP TS 29.510 Release 15.5 and Release 16.0. In addition to this, *ALL_NF_TYPE* and *CUSTOM_NF_TYPE* is also supported. *ALL_NF_TYPE* is the NF Type to be used to specify the default configuration that is to be used when nfType specific configuration is not present. **Notes**: • By default, the values will pre-loaded for *ALL_NF_TYPE*. • ALL_NF_TYPE element cannot be deleted. • *CUSTOM_NF_TYPE* is the NFType to be used to specify the configuration for custom NF types.	**DataType**: string **Constraints**: NFType **Default Value**: ALL_NF_TYPE
`minHbTimer`	The minimum Heartbeat Timer allowed for the NF. The value is in pHqMrS format, where p,q,r are integers and H,M,S or h,m,s denotes hours, minutes, and seconds respectively.	**DataType**: string **Constraints**: 10s-24h **Default Value**: 30s
`maxHbTimer`	The maximum Heartbeat Timer allowed for the NF. The value is in pHqMrS format, where p, q, r are integers and H, M, S or h, m, s denotes hours, minutes, and seconds respectively.	**DataType**: string **Constraints**: 10s-24h **Default Value**: 5m
`defaultHbTimer`	This default Heartbeat Timer value is used when the network functions do not provide the Heartbeat Timer value in NFProfile. The value is in pHqMrS format, where p, q, r are integers and H, M, S or h, m, s denotes hours, minutes, and seconds respectively.	**DataType**: string **Constraints**: minHbTimer and maxHbTimer attributes **Default Value**: 30s
`nfHeartbeatMissAllowed`	The allowed number of missed HeartBeat(s) after which the NFProfile is marked as suspended. If the value is set to 0, NF profiles for which even single heartbeat is missed will be marked as suspended.	**DataType**: integer **Constraints**: 0-15 **Default Value**: 3

Table 2-12 RequestRetryProfile

Attribute	Description	Details
`featureStatus`	This flag enables or disables the *NfStatusNotify* service operation request retry. If any attribute is null in request body, then the previously saved values are retained and considered. If featureStatus value is ENABLED, errorResponseCodeList and exceptionResponseList cannot be simultaneously null, either of the list must be present. If featureStatus value is ENABLED, value of the `notificationRequestTimeout` parameter is calculated using the following formula: ((retryCount+1)*requestTimeOut) + 1000mS (DELTA_VALUE). However, if featureStatus value is DISABLED, both errorResponseCodeList and exceptionResponseList can be simultaneously empty. featureStatus value cannot be ENABLED, if both errorResponseCodeList and exceptionResponseList are null.	**DataType**: string **Constraints**: ENABLED or DISABLED **Default Value**: DISABLED
`retryCount`	The number of retries that happens at Egress Gateway upon the following scenarios: • 4xx, 5xx Response Error Codes: In case of the error response from the notification callback server, http status codes are matched with the errorResponseCodeList. • Connection Failure/Timeout In case of connection failure/timeout, error occurred at Egress Gateway is matched with the exceptionResponseList. • Request Timeout In case of request timeout, error occurred at Egress Gateway is matched with the exceptionResponseList.	**DataType**: integer **Constraints**: 1 - 5 **Default Value**: 3
`requestTimeout`	This configuration decides the request timeout if response from notification callback server is not received. **Note: a.** The value here corresponds in milliseconds. **b.** Upon requestTimeout value elapsed for unsuccessful outbound request, retry will happen depending upon java.util.concurrent.TimeoutException configured in the exceptionResponseList attribute.	**DataType**: integer **Constraints**: 100 - 5000 **Default Value**: 3000
`errorResponseCodeList`	This configuration is the list of HTTP error codes that are authorized for retry. In case of the 4xx, 5xx Response Error Codes from the notification callback server, the HTTP status codes from error response are matched with this configuration.	**DataType**: array(string) **Constraints**: maximum 10 values can be configured **Default Value**: ["408","409","5xx"]

Table 2-12 (Cont.) RequestRetryProfile

Attribute	Description	Details
exceptionResponseList	This configuration is the list of exceptions that are authorized for retry. Below is the set of possible exceptions: • java.net.UnknownHostException: Exception is raised when the address of a host could not be determined, or host address is invalid • javax.net.ssl.SSLHandshakeException:Exception is raised when the client and server could not negotiate the desired level of security. • java.nio.channels.ClosedChannelException: Exception is raised when an attempt is made to invoke or complete an I/O operation upon channel that is closed, or at least closed to that operation • java.net.ConnectException: Exception is raised when an error occurred while attempting to connect a socket to a remote address and port • java.util.concurrent.RejectedExecutionException: Exception is raised when a task cannot be accepted by the executor • java.util.concurrent.TimeoutException: Exception is raised when the NFStatusNotify request is timed out • java.net.SocketTimeoutException: Exception is raised when a timeout has occurred on a socket read or accept. This occurs in case of connection is not established within time.	**DataType**: array(string) **Constraints**: maximum 10 exceptions can be configured **Default Value**: [java.util.concurrent.TimeoutException,java.net.ConnectException]

Subscription Limit

Table 2-13 Subscription Limit

Attribute	Description	Details
featureStatus	This attribute is used to enable or disable the Subscription Limit feature.	**DataType**: string **Constraints**: ENABLED or DISABLED **Default Value**: DISABLED
globalMaxLimit	This attribute is used to set the maximum number of subscriptions allowed for the NRF. **Note**: The value of this attribute must be same across all georedundant sites.	**DataType**: integer **Constraints**: 0 to 1000 **Default Value**: 850
rejectSubscriptionRenewalWhenLimitBreached	This flag is used to indicate whether Subscription Renewal is allowed when the Global Subscription limit is breached.	**DataType**: string **Constraints**: ENABLED or DISABLED **Default Value**: ENABLED
limitThresholds	This attribute is used to configure the subscription limit thresholds for which alerts are raised. **Note**: Duplicate level name is not allowed in limitThresholds. For more information, see Limit Threshold.	**DataType**: array list **Constraints**: NA **Default Value**: NA

Table 2-13 (Cont.) Subscription Limit

Attribute	Description	Details
errorResponses	Indicates the error response that is generated when a subscription request is rejected. For more information, see Table 2-16.	**DataType**: array list **Constraints**: NA **Default Value**: NA

Table 2-14 Limit Threshold

Attributes	Description	Details
level	This attribute indicates the name of the level. For more information about the onset and abatement values, see Table 2-15. **Note**: Duplicate level name is not allowed.	**DataType**: string **Constraints**: NA **Default Value**: NA
onset	This attribute describes onset value in percentage.	**DataType**: integer **Constraints**: NA **Default Value**: NA
abatement	This attribute describes abatement value in percentage.	**DataType**: integer **Constraints**: NA **Default Value**: NA

Table 2-15 Onset and abatement at different level

Level	Onset	Abatement
WARN	50	45
MINOR	60	55
MAJOR	70	65
CRITICAL	90	75

Table 2-16 PreLoaded records for errorResponses

errorCondition	responseCode	errorResponse	errorCause	retryAfter	redirectUrl
Subscription_Global_Limit_Breached	500	Subscription global limit breached	"INSUFFICIENT_RESOURCES"	5m	""

Sample cURL Command

The following example shows how an API Invoker is onboarded by submitting a GET request on the REST resource using cURL.

```
curl -X 'GET' \
  '{apiRoot}/nrf-configuration/v1/nfManagementOptions' \
  -H 'accept: application/json'
```

2.5 NF Discovery Options

This section provides REST API configuration parameter details to configure NRF discovery options.

URI: *{apiRoot}/nrf-configuration/v1/nfDiscoveryOptions*

Method: PUT and GET

- **PUT**: Updates NRF discovery options configuration
- **GET**: Retrieves NRF discovery options configuration

Content Type: application/json

Body:

```
{
   "profilesCountInDiscoveryResponse": 3,
   "discoveryResultLoadThreshold": 0,
    "discoveryValidityPeriodCfg":
    [
      {
         "nfType":"ALL_NF_TYPE",
         "validityPeriod":"1h",
         "emptyListValidityPeriod":"30s"
      },
     {
         "nfType":"AMF",
         "validityPeriod":"1h",
         "emptyListValidityPeriod":"20s"
      }
    ],
      "emptyDiscoveryResponseConfig":{
        "emptyListFeatureStatus": "DISABLED",
        "emptyListConfig":  [
            {
               "nfType": "AMF",
               "featureStatus": "DISABLED"
            },
            {
               "nfType": "SMF",
               "featureStatus": "ENABLED"
            }]
      },
    "extendedPreferredLocality": {
        "featureStatus": "DISABLED",
        "locationTypes": ["SAP", "NEC", "Category-x","Category-y","Category-
xy"],
        "locationTypeMapping": [{
                "nfType": "PCF",
                "nfServices": ["am_policy", "bdt_policy"],
                "locationType": "Category-x"
            },
            {
                "nfType": "PCF",
```

```
                    "nfServices": ["sm_policy"],
                    "locationType": "Category-y"
                },
                {
                    "nfType": "PCF",
                    "nfServices": ["*"],
                    "locationType": "Category-xy"
                },
                {
                    "nfType": "AMF",
                    "nfServices": ["*"],
                    "locationType": "SAP"
                }
            ],
            "preferredLocationDetails": [{
                    "preferredLocation": "Azusa",
                    "targetLocationType": "Category-x",
                    "maxNFProfilesFromFirstMatchLoc": 0,
                    "targetPreferredLocations": [{
                        "priority": 1,
                        "location": "Azusa"
                    }, {
                        "priority": 2,
                        "location": "Vista"
                    }, {
                        "priority": 3,
                        "location": "Ohio"
                    }]
                },
                {
                    "preferredLocation": "Azusa",
                    "targetLocationType": "Category-y",
                    "maxNFProfilesFromFirstMatchLoc": 0,
                    "targetPreferredLocations": [{
                        "priority": 1,
                        "location": "RKL"
                    }, {
                        "priority": 2,
                        "location": "CSP"
                    }, {
                        "priority": 3,
                        "location": "West-Region-Edge-Set01"
                    }]
                }
            ],
            "locationSets": [
                {
                    "locationSetName": "West-Region-Edge-Set01",
                    "locations": ["London", "New York"]
                }
            ]
        }
    }
```

Configuration Attributes

> **Note**
>
> If any attribute is not present in the JSON request body while it is being updated, the existing value in its database is preserved and used. At least one attribute is included during the PUT request.

Table 2-17 nfDiscoveryOptions

Attribute Name	Description	
`profilesCountInDiscoveryResponse`	This value restricts NF profile count in the NFDiscover response. If the value of this attribute is 0, it means this functionality is disabled, and all the NF profiles after the discovery filtering are returned in the NFDiscover response. **Note**: This attribute is not considered if the Limit attribute is present in SearchData URI.	**DataType**: integer **Constraints**: 0 to 20 **Default Value**: 3
`discoveryResultLoadThreshold`	NFDiscover response contains NF profiles with load attribute value less than or equal to this configured value. In case there are no profiles matching this criteria, then the profiles with load greater than the configured value are included in the response. Value 0 indicates this feature is disabled.	**DataType**: integer **Constraints**: 0 to 100 **Default Value**: 0
`discoveryValidityPeriodCfg`	This attribute mentions the validity period of a discovery request for a specific target-nf-type. The NF that sent the discovery request must perform a discovery action again to get the latest values. By default, the validityPeriod information for ALL_NF_TYPE is present.	**DataType**: array (DiscoveryValidityPeriodCfg) **Constraints**: See DiscoveryValidityPeriodCfg table for details. **Default Value**: NA
`emptyDiscoveryResponseConfig`	This attribute provides the configuration for the EmptyList feature.	**DataType**: List <emptyDiscoveryResponseConfig> **Default Value**: Empty array
`extendedPreferredLocality`	This attribute is used for the Extended Preferred Locality feature.	**DataType**: array (ExtendedPreferredLocality) **Constraints**: See ExtendedPreferredLocality table for details. **Default Value**: NA

Table 2-18 emptyDiscoveryResponseConfig

Parameter	Description	Details
`emptyListFeatureStatus`	This attribute defines if the empty list feature is enabled. If the value is set to ENABLED, then NRF checks if the particular target-nf-type is configured in `emptyListConfig`. If the value is set to DISABLED, then NRF does not send any Producer NF profile in the response.	**DataType**: string **Constraints**: ENABLED or DISABLED **Default Value**: DISABLED
`emptyListConfig`	This attribute lists the configuration for the EmptyList feature for a specific NF type.	**DataType**: List <emptyListConfig> **Default Value**: Empty array

Table 2-19 emptyListConfig

Parameter	Description	Details
`nfType`	This attribute indicates the NF type of a particular Network Function. This value is matched with the target-nf-type in the Discovery Query. In addition to this, CUSTOM_<NFType> is also supported. **Note**: CUSTOM_<NFType> is the NFType used to specify the configuration for custom NF types.	**DataType**: NFType **Constraints**: Valid NFType **Default Value**: NA
`featureStatus`	This attribute describes the status of a particular nfType. If the value is set to ENABLED, NRF sends a discovery response with a new validity period as mentioned in `emptyListValidityPeriod` attribute. If the value is set to DISABLED, NRF sends an empty response. **Note**: When `emptyListFeatureStatus` is DISABLED, NRF sends an empty discovery response even though the `featureStatus` is ENABLED for that specific target-nf-type.	**DataType**: string **Constraints**: ENABLED or DISABLED **Default Value**: DISABLED

Table 2-20 ExtendedPreferredLocality

Attribute	Description	Details
`featureStatus`	This decides extendedPreferredLocality feature is enabled or not. locationTypes, locationTypeMapping, and preferredLocationDetails are configured before or during enabling the feature.	**DataType**: string **Constraints**: ENABLED and DISABLED **Default Value**: DISABLED

Table 2-20 (Cont.) ExtendedPreferredLocality

Attribute	Description	Details
`locationTypes`	This attribute decides different location types. • Maximum 25 location types can be configured. • locationType can have a minimum of 3 characters and a maximum of 36 characters. It can have only alphanumeric and special characters '-' and '_'. locationType must not start and end with special characters. • Duplicate Location types are not allowed. • Case sensitive, it means Category-x and Category-X are different.	**DataType**: array (string) **Constraints**: NA **Default Value**: Empty array
`locationTypeMapping`	This attribute specifies which NF Type (along with NF services) is mapped to which Location Type. See LocationTypeMapping table for details.	**DataType**: array (LocationTypeMapping) **Constraints**: Maximum 100 locationTypeMapping values can be configured. **Default Value**: Empty array
`preferredLocationDetails`	This attribute specifies the preferred location (derived from discovery search query) and locationType from locationTypeMapping attribute maps to extended preferred location(s). See PreferredLocationDetails table for details.	**DataType**: array (PreferredLocationDetails) **Constraints**: Maximum 650 preferredLocationDetails values can be configured. **Default Value**: Empty array
`locationSets`	This attribute specifies a set of locations that define the preferred locality for NfDiscovery. See locationSets table for details.	**DataType**: array (locationSets) **Constraints**: Maximum of 255 location sets can be configured. **Default Value**: Empty array

Table 2-21 LocationTypeMapping

Attribute	Description	Details
`nfType`	This attribute indicates the NF type of a particular Network Function. This value is derived from the target-nf-type of Discovery Search query.	**DataType**: string **Constraints**: It can be any one of 3GPP defined ones or values starting with CUSTOM_ **Default Value**: NA **Presence**: M
`nfServices`	This attribute indicates the NF Services that belong to the Network Function.	**DataType**: array (string) **Constraints**: • This value can be '*' only. '*' denotes that mapped location type supports all services for nfType. Along with this, in case `nf-services` from Discovery Search Query do not match with any of nfServices configured or service-names attribute is not present in Discovery Search Query, record with '*' can be mapped. • Same nfService(s) cannot be mapped to different location type. **Default Value**: NA **Presence**: M

Table 2-21 (Cont.) LocationTypeMapping

Attribute	Description	Details
`locationType`	This attribute indicates the location type to which NF is mapped.	**DataType**: string **Constraints**: It is one of locationTypes attributes. **Default Value**: NA **Presence**: M

Table 2-22 PreferredLocationDetails

Attribute	Description	Details
`preferredLocation`	This value is matched with preferredLocation from consumer NF in Discovery Search Query.	**DataType**: string **Constraints**: NA **Default Value**: NA **Presence**: M
`targetLocationType`	This value is derived from the mapped locationType attribute value of the LocationTypeMapping table.	**DataType**: string **Constraints**: This value is one of the configured in locationTypes attributes. **Default Value**: NA **Presence**: M
`maxNFProfilesFromFirstMatchLoc`	This attribute is used to configure the maximum number of NF profiles that can be selected from the first matching location from the `targetPreferredLocations`. It is recommended to have the value of this configuration to be less than or equal to `profilesCountInDiscoveryResponse` which limits the overall count of NF profiles in NFDiscover service operation response. **Note:** • The value of this attribute will become 0 during upgrade scenarios. • If the value of this attribute is 0, the feature is disabled.	**DataType**: integer **Constraints**: 0-20 **Default Value**: NA **Presence**: M . If the value of this attribute is 0, the feature is disabled.
`targetPreferredLocations`	The preferred locations are configured by the operator for particular `preferredLocation` and `targetLocationType`. See TargetPreferredLocations table for details.	**DataType**: string **Constraints**: There can be minimum 1 and maximum 3 TargetPreferredLocations in this list. **Default Value**: NA **Presence**: M

Table 2-23 TargetPreferredLocations

Attribute	Description	Details
`priority`	Indicates the priority of PreferredLocation.	**DataType**: integer **Constraints**: Duplicate priority is not allowed. **Default Value**: NA **Presence**: M

Table 2-23 (Cont.) TargetPreferredLocations

Attribute	Description	Details
`location`	The operator can configure the location or set of locations. In case of location set, configure locationSet.	**DataType**: string **Constraints**: Location can be same as preferredLocation mentioned in PreferredLocationDetails. **Default Value**: NA **Presence**: M

Table 2-24 DiscoveryValidityPeriodCfg

Attribute	Description	Details
`nfType`	This indicates all the nfTypes that are supported in 3GPP TS 29.510 Rel 16.3.0. In addition to this, *ALL_NF_TYPE* and *CUSTOM_<NFType>* are also supported. *ALL_NF_TYPE* is the NF Type used to specify the default configuration that is to be used when nfType specific configuration is not present. **Notes**: • By default, the values are preloaded for *ALL_NF_TYPE*. For more information, see "NfTypeValidityPeriod Loaded Data" table. • ALL_NF_TYPE element cannot be deleted. • *CUSTOM_<NFType>* is the NFType to be used to specify the configuration for custom NF types.	**DataType**: NfType **Cardinality:** 0..1 **Default Value**: NA
`validityPeriod`	This attribute mentions the validity period of a discovery request of a specific target-nf-type after which requester NF must perform discovery again to get the latest values. The value is in pHqMrS format, where p, q, r are integers and H, M, S or h, m, s denote hours, minutes, and seconds respectively.	**DataType**: string **Range:** 0s to 720h **Default Value**: NA
`emptyListValidityPeriod`	This attribute mentions the validity period for an empty list response of a discovery request of a specific target-nf-type. This value is sent as `validityPeriod` in the discovery response when NRF is generating empty response, irrespective of `emptyListFeatureStatus` is ENABLED or DISABLED. In case the specific nfType is not configured, then `emptyListValidityPeriod` configured for ALL_NF_TYPE is considered in the discovery response. Upon expiry of the `emptyListValidityPeriod`, Producer NF must send the discovery request again. The value is in pHqMrS format, where p, q, r are integers and H, M, S or h, m, s denote hours, minutes, and seconds respectively.	**DataType**: string **Range:** 0s to 720h **Default Value**: NA

Table 2-25 locationSets

Attribute Name	Description	Details
`locationSetName`	An identifier to distinguish among other locations. It must be alphanumeric and allows only **-** and _ as special characters. **Note**: It is recommended to append **Set** as a keyword for every `locationSetName` specifying group of locations, for easy identification between a single location and group of locations. For example: West-Region-Edge-Set01	**DataType**: string **Constraints**: 0 or 1 The length of the location name can be in the range of 5 to 100 characters. **Default Value**: NA **Presence**: M
`locations`	Set of unique locations, where location is of type string.	**DataType**: list (String) **Constraints**: 0 to 10 **Default Value**: NA **Presence**: M

Table 2-26 NfTypeValidityPeriod Loaded Data

Attribute	Default Loaded Value
`nfType`	ALL_NF_TYPE
`validityPeriod`	1h
`emptyListValidityPeriod`	30s

2.6 NF Access Token Options

This section provides REST API configuration parameter details to configure NRF Access Token options.

URI: *{apiRoot}/nrf-configuration/v1/nfAccessTokenOptions*

Method: PUT and GET

- **PUT**: Updates NRF Access Token options configuration.
- **GET**: Retrieves NRF Access Token options configuration.

Content Type: application/json

Body:

```
{
   "oauthTokenExpiryTime": "1h",
   "authorizeRequesterNf": "ENABLED",
   "logicalOperatorForScope": "AND",
   "audienceType": "NF_INSTANCE_ID",
   "authFeatureConfig": {
      "featureStatus": "DISABLED",
      "authRulesConfig": [
            {
               "targetNfType":"PCF",
               "requesterNfType":"AMF",
               "serviceNames":[
```

```
                    "npcf-am-policy-control",
                    "npcf-eventexposure"
                ]
            },
            {
                "targetNfType":"UDM",
                "requesterNfType":"AMF",
                "serviceNames":[
                    "*"
                ]
            }
        ],
      "errorResponses": [{
         "errorCondition": "RequesterNf_Unauthorized",
         "responseCode": 400,
         "errorResponse": "The Consumer NfType is not authorized to receive
access token for the requested Nftype.",
         "errorCause": "UNSPECIFIED_MSG_FAILURE",
         "retryAfter": "5m",
         "redirectUrl": ""
     }]
   },
   "tokenSigningDetails": {
        "currentKeyID": "a14ef8e1bc5c",
        "addkeyIDInAccessToken": true,
        "oauthTokenIssuerId": "6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c",
        "defaultK8SecretDetails": {
            "k8SecretName": "ocnrf",
            "k8SecretNameSpace": "ocnrf"
        },
        "keyDetailsList": [
            {
                "keyID": "a14ef8e1bc5c",
                "algorithm": "ES256",
                "privateKey": {
                    "k8SecretName": "ocnrf",
                    "k8SecretNameSpace": "ocnrf",
                    "fileName": "ec_private_key_pkcs8.pem"
                },
                "certificate": {
                    "k8SecretName": "ocnrf",
                    "k8SecretNameSpace": "ocnrf",
                    "fileName": "ecdsa_ocnrfapigatewayTestCA.cer"
                }
            }
        ]
    },
   "errorResponses": [
        {
            "errorCondition": "Invalid_Key_Details",
            "responseCode": 500,
            "errorResponse": "Configured Key ID details are invalid and
cannot be used",
            "errorCause": "UNSPECIFIED_NF_FAILURE",
            "retryAfter": "5m",
            "redirectUrl": ""
```

```
            },
            {
                "errorCondition": "Current_Key_Id_Not_Configured",
                "responseCode": 500,
                "errorResponse": "Current Key ID is not configured",
                "errorCause": "UNSPECIFIED_NF_FAILURE",
                "retryAfter": "5m",
                "redirectUrl": ""
            }
        ]
}
```

Configuration Attributes

> **Note**
>
> If any attribute is not present in the JSON request body while updating, the existing value in the database is preserved and used. At least one attribute is included during the PUT request.

Table 2-27 nfAccessTokenOptions

Attribute Name	Description	Details
`oauthTokenExpiryTime`	Oauth token expiry time. The value is in pHqMrS format, where p,q,r are integers and H,M,S or h,m,s denotes hours, minutes and seconds respectively. **Note**: In case NRF signed certificate expiry duration is less than this attribute, then certificate expiry duration is used in Access Token Expiry Time.	**DataType**: string **Constraints**: 1s - 168h **Default Value**: 1h
`authorizeRequesterNf`	This attribute validates the requester NF is registered with NRF or not. NRF issues the access token only to the registered requester NFs. If NF is registered, then check if NFtype in Access Token Request is same as in NF profile registered with NRF and requesterPlmn received in the Access Token Request request is same as Registered Profile. If the value is set as DISABLED, NRF will issue token to non-registered NFs as well.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: ENABLED
`audienceType`	This value decides the aud attribute (Type:-Audience) in AccessTokenClaim as per 3GPP specification. NRF considers this value only if targetNfType and targetNfInstanceId both are not received in AccessTokenRequest. **Possible values are:** • NF_INSTANCE_ID - NF Instance Id(s) in aud attribute of the AccessTokenClaim. • NF_TYPE - NF Type in aud attribute of the AccessTokenClaim.	**DataType**: string **Constraints**: NF_INSTANCE_ID,NF_TYPE **Default Value**: NF_INSTANCE_ID

Table 2-27 (Cont.) nfAccessTokenOptions

Attribute Name	Description	Details
`logicalOperatorForScope`	The value decides whether values in scope will have relationship AND or OR. If the value is set as AND, while looking for producer network function profiles, token will be issued for profiles matching all the services-names present in scope. If the value is set as OR, token will be issued for profiles matching any of the services-names present in scope.	**DataType**: string **Constraints**: AND, OR **Default Value**: AND
`authFeatureConfig`	The attribute contains the parameters required to enable and configure NfAccessToken Authorization feature.	**DataType**: array (AuthFeatureConfig) **Constraints**: See AuthFeatureConfig table for details. **Default Value**: NA
`tokenSigningDetails`	This attribute allows user to configure all of the details required to sign the token generated by NRF.	**DataType**: array (TokenSigningDetails) **Constraints**: See TokenSigningDetails for details. **Default Value**: null (None of token signing details are configured)
`errorResponses`	This attribute allows user to update details for different error conditions.	**DataType**: array (ErrorInfo) **Constraints**: See table Preloaded Values for details. **Default Value**: NA

Table 2-28 PreLoaded records for errorResponses

errorCondition	responseCode	errorResponse	errorCause	retryAfter	redirectUrl
Invalid_Key_Details	500	Configured Key ID details are invalid and cannot be used	UNSPECIFIED_NF_FAILURE	5m	See ErrorInfo.
Current_Key_Id_Not_Configured	500	Current Key ID is not configured	UNSPECIFIED_NF_FAILURE	5m	See ErrorInfo.

AuthFeatureConfig

Table 2-29 AuthFeatureConfig

Attribute	Description	Details
`featureStatus`	Enables or disables the NfAccessToken Authorization Feature.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
`authRulesConfig`	The attribute defines a mapping across Requester NF Type, Target NF Type, and the allowed services. This attribute should be configured if the `authFeatureStatus` is set to 'ENABLED'.	**DataType**: array (AuthConfig) **Constraints**: NA **Default Value**: NA

Table 2-29 (Cont.) AuthFeatureConfig

Attribute	Description	Details
`errorResponses`	This attribute defines the error responses which are sent during NRF AccessToken Authorization failure scenarios. This attribute allows to update the error response code and error response description. This attribute must be configured if the `authFeatureStatus` is set to 'ENABLED'. By default, the RequesterNF_Unauthorized condition is preloaded.	**DataType**: array (ErrorInfo) **Constraints**: See PreLoaded records for AuthFeatureConfig errorResponses. **Default Value**: NA

Table 2-30 PreLoaded records for AuthFeatureConfig errorResponses

errorCondition	response Code	errorResponse	errorCause	retryAfter	redirectUrl
RequesterNf_Unauthorized	400	The RequesterNfType is not authorized to receive access token for the targetNfType.	UNSPECIFIED_MSG_FAILURE	5m	See ErrorInfo.

> **Note**
>
> The attributes featureStatus, authRulesConfig, and errorResponses can be configured independently in any order. However, when the feature is enabled, it is expected that the authRulesConfig is configured or present in the current request.

Table 2-31 AuthConfig

Attribute	Description	Details
`targetNfType`	The attribute defines the NF Type of the target NF.	**DataType**: string **Constraints**: NFType **Default Value**: NA
`requesterNfType`	The attribute defines the NF Type of the requester NF that is authorized to access the target NF Type and its services.	**DataType**: string **Constraints**: NFType **Default Value**: NA
`serviceNames`	This attribute defines the NF services that are authorized to be accessed by the requester NF type. The value "*" indicates that all the services are authorized to be accessed by the requester NF Type. If "*" is to be used, The services contain only a single entry in the list with this value.	**DataType**: array (string) **Constraints**: None **Default Value**: NA

> **Note**
>
> It is mandatory to configure all the attributes together.

Table 2-32 TokenSigningDetails

Attribute	Description	Details
`currentKeyID`	KeyID value corresponding to the token signing details used to sign the token. Mandatory attribute for Access Token Service to work. Newly added KeyID details can be used after values are validated by NRF. Newly added KeyID cannot be configured as currentKeyID in same request.	**DataType**: string **Constraints**: Once currentKeyID is configured, this value cannot be null. **Default Value**: NA
`addkeyIDInAccessToken`	The value of this attribute decides if the KeyID value can be added to AccessToken Response or not. If value is true, then currentKeyID value will be added in AccessToken Response. If value is false, then value will not be added in AccessToken Response. Value for this attribute cannot be set to true if currentKeyID is not set.	**DataType**: boolean **Constraints**: true, false **Default Value**: false
`oauthTokenIssuerId`	This attribute is NRF Instance ID that is used for signing AccessTokenClaim (IE of AccessTokenClaim). If NRF needs to issue AccessTokenClaim using its own NF instance ID then the nrfInstanceId configured in the global section (`global.nrfInstanceId`) needs to configured here. In case of fresh Install, this value is populated with `global.nrfInstanceId` automatically. If NRF needs to issue AccessTokenClaim using a common or virtual then a common or virtual NF instance ID needs to be configured here (along with the common or virtual PrivateKey and Certificate Pair). The same NF instance id and PrivateKey and Certificate Pair has to be configured in all other NRFs so that tokens issued by all the NRFs can be validated using a Single NfInstanceId and KeyPair. While upgrading from 1.12.x to 1.14.x, this value is taken from `nfaccesstoken.oauth.nrfInstanceId` helm attribute. In case this value is not set in helm custom values.yaml, value is taken from `global.nrfInstanceId` helm attribute.	**DataType**: string **Constraints**: Mandatory attribute for Access Token Service to work. It should be in UUID format. **Default Value**: NA
`defaultK8SecretDetails`	This attribute decides the default Kubernetes secret details and these details value are used in case individual key details for secret name and secret namespace are not configured.	**DataType**: DefaultK8SecretDetails **Constraints**: See DefaultK8SecretDetails table for details. This value is mandatory in case secret namespace and secret name of any individual key details are not configured. **Default Value**: NA

Table 2-32 (Cont.) TokenSigningDetails

Attribute	Description	Details
keyDetailsList	This attribute provides details of oauth key details which is used by NRF to sign the token. For more information about adding Keys status, see *Oracle Communications Cloud Native Core, Network Repository Function User Guide*. Any key ID details cannot be removed if it is getting used as currentKeyID. See OauthKeyDetails table for details.	**DataType**: array (OauthKeyDetails) **Constraints**: Maximum 25 key details can be configured. In case any of key details need to be modified, then complete keyDetailList is used for updates. Change of specific keyId detail is not supported. **Default Value**: NA

Table 2-33 DefaultK8SecretDetails

Attribute	Description	Details
k8SecretName	Default Kubernetes secret name.	**DataType**: string **Constraints**: Mandatory, if default Kubernetes secret details are configured. Both k8SecretName and k8SecretNameSpace are configured together. **Default Value**: NA
k8SecretNameSpace	Default Kubernetes secret namespace.	**DataType**: string **Constraints**: Mandatory, if default Kubernetes secret details are configured. Both k8SecretName and k8SecretNameSpace are configured together. **Default Value**: NA

Table 2-34 OauthKeyDetails

Attribute	Description	
keyID	Unique value in list of keys. Key details are known by this value.	**DataType**: string **Constraints**: Mandatory attribute, keyID length must not exceed 36 characters. **Default Value**: NA
algorithm	Algorithm value is used to sign the oauth token.	**DataType**: string **Constraints**: Mandatory attribute, algorithm can be only ES256 and RS256. **Default Value**: NA
privateKey	NRF Private key details. Both k8SecretName and k8SecretNameSpace are configured together.	**DataType**: OauthSecretFiles **Constraints**: Mandatory attribute, see OauthSecretFiles for details. **Default Value**: NA
certificate	NRF Public certificate details. Both k8SecretName and k8SecretNameSpace are configured together.	**DataType**: OauthSecretFiles **Constraints**: Mandatory attribute, see OauthSecretFiles for details. **Default Value**: NA

Table 2-35 OauthSecretFiles

Attribute	Description	Details
k8SecretName	Kubernetes secret name where key and certificate details are stored.	**DataType**: string **Constraints**: Optional, but if this attribute is present then k8SecretNameSpace must be present. **Default Value**: NA
k8SecretNameSpace	Kubernetes namespace for secret name.	**DataType**: string **Constraints**: Optional, but if this attribute is present then k8SecretName must be present. **Default Value**: NA
fileName	Filename of Key or certificate.	**DataType**: string **Constraints**: Mandatory attribute. **Default Value**: NA

2.7 NRF-NRF Forwarding Options

This section provides REST API configuration parameter details to configure NRF forwarding options.

URI: *{apiRoot}/nrf-configuration/v1/forwardingOptions*

Method: PUT and GET

- **PUT**: Updates NRF forwarding options configuration.
- **GET**: Retrieves NRF forwarding options configuration.

Content Type: application/json

Body:

```
{
   "profileRetrievalStatus": "DISABLED",
    "subscriptionStatus": "DISABLED",
    "discoveryStatus": "DISABLED",
    "accessTokenStatus": "ENABLED",
    "nrfHostConfig": [
        {
            "nfInstanceId": "c56a4180-65aa-42ec-a945-5fd21dec0538",
            "apiVersions": [
                {
                    "apiVersionInUri": "v1",
                    "apiFullVersion": "15.5.0"
                }
            ],
            "scheme": "http",
            "host": "ocnrf-1-ingressgateway.ocnrf.svc.cluster.local",
            "priority": 100,
            "port": 80
        }
    ],
    "nrfRerouteOnResponseHttpStatusCodes": {
```

```
            "pattern": "^[3,5][0-9]{2}$|408$",
            "codeList": null
        },
        "errorResponses": [
            {
                "errorCondition": "NRF_Not_Reachable",
                "responseCode": 504,
                "errorResponse": "NRF not reachable",
                "errorCause": "UNSPECIFIED_NF_FAILURE",
                "retryAfter": "5m",
                "redirectUrl": ""
            },
            {
                "errorCondition": "NRF_Forwarding_Loop_Detection",
                "responseCode": 508,
                "errorResponse": "Loop Detected",
                "errorCause": "UNSPECIFIED_NF_FAILURE",
                "retryAfter": "5m",
                "redirectUrl": ""
            }
        ],
        "forwardingRulesFeatureConfig": {
            "featureStatus": "ENABLED",
            "forwardingRulesConfig": [
                {
                    "targetNfType": "UDM",
                    "serviceNames": [
                        "nudm-uecm"
                    ],
                    "serviceNamesMatchType": "ANYONE"
                },
                {
                    "targetNfType": "*",
                    "serviceNames": [
                        "UDMname14","UDMname15"
                    ],
                    "serviceNamesMatchType": "EXACT"
                }
            ]
}
}
```

Configuration Attributes

> **Note**
>
> If any attribute is not present in the JSON request body while updating, the existing value in the database is preserved and used. The profileRetrievalStatus, subscriptionStatus, discoveryStatus, and accessTokenStatus attributes are mandatory in the PUT request.

Table 2-36 forwardingOptions

Attribute Name	Description	Details
`nrfHostConfig`	This attribute is used to configure Primary and Secondary NRF details used for forwarding various requests. It allows to configure details of NRF like apiVersion, scheme, host, port, and so on. The only supported value for apiVersionInUri is v1. Hence the apiVersions attribute must have at least one data record with apiVersionInUri attribute values set as v1. This configuration allows you to configure more than two NRF Details. NRF with highest priority is considered as Primary NRF for forwarding messages. NRF with second highest priority is considered as Secondary NRF for forwarding. To reset this attribute, please send empty array, for example:- "nrfHostConfig": [ ] If this attribute is already set then there is no need to provide the value again. **Note**: The value of this attribute can be FQDN, IPv4 or IPv6.	**DataType**: array (NFConfig) **Constraints**: NA **Default Value**: NA
`nrfRerouteOnResponseHttpStatusCodes`	This configuration is used to determine if the service operation message needs to be forwarded to Secondary NRF. The primary NRF receives a response. If the response status code matches the configured response status code list, then NRF reroutes the request to the secondary NRF. Refer nrfHostConfig for details for Primary and Secondary NRF details.	**DataType**: ResponseHttpStatusCodes **Constraints**:: pattern or specific code list **Default Value**: "pattern": "^[3,5][0-9]{2}$\|408$"
`profileRetrievalStatus`	This attribute controls the forwarding of NFProfileRetrieval service operation messages. If the flag is set to true and NRF is unable to complete the request due to the unavailability of any matching profile, then NRF forwards the NfProfileRetrieval request to the configured NRF host(s) and relays the response received from forwarding NRF to the Consumer NF. If the flag is set to false, NRF will not forward the NfProfileRetrieval request. It returns a response to the consumer NF without forwarding it.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED

Table 2-36 (Cont.) forwardingOptions

Attribute Name	Description	Details
`subscriptionStatus`	This attribute controls the forwarding of NFStatusSubscribe, and NFStatusUnsubscribe service operation messages. If the flag is set to true and NRF cannot complete the request due to the unavailability of any matching profile, then NRF forwards the NfStatusSubscribe or NfStatusUnSubscribe request to the configured NRF host(s) and relays the response received from forwarding NRF to the Consumer NF. If the flag is false, NRF will not forward the NFStatusSubscribe or NFStatusUnSubscribe request. It returns a response to the consumer NF without forwarding it. **Note**: NFStatusSubscribe forwarding is supported only if Subscription Condition is NfInstanceIdCond in the NFStatusSubscribe request.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
`discoveryStatus`	This attribute controls the forwarding of NFDiscover service operation messages. If the flag is set to ENABLED and NRF is not able to complete the request due to unavailability of any matching profile, then NRF forwards the NFDiscover request to the configured NRF host(s) and relays the response received from forwarding NRF to the Consumer NF. If flag is DISABLED, NRF will not forward the NFDiscover request in any case. It will return a response to consumer NF without forwarding it.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
`accessTokenStatus`	This attribute controls the forwarding of AccessToken service operation messages. If the flag is set to ENABLED and NRF is not able to complete the request due to unavailability of any matching Producer NF, then NRF forwards the AccessToken request to the configured NRF host(s) and relays the response received from forwarding NRF to the Consumer NF. If flag is DISABLED, NRF will not forward the AccessToken request in any case. It will return a response to consumer NF without forwarding it.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
`forwardingRulesFeatureConfig`	This attribute provide details for Forwarding Rules feature configuration.	**DataType**: ForwardingRulesFeatureConfig **Constraints**: NA **Default Value**: NA
`errorResponses`	This attribute defines the error responses which may be sent during NRF Forwarding scenarios. This attribute will allow to update the error response code and error response description for preloaded error conditions.	**DataType**: array (ErrorInfo) **Constraints**: NA **Default Value**: NA

Table 2-37 ForwardingRulesFeatureConfig

Attribute	Description	
`featureStatus`	This attribute enables or disables the evaluation of forwarding eligibility of a service request based on target NF type and service names configured in forwardingRulesConfig. This flag can be ENABLED only if `discoveryStatus` or `accessTokenStatus` attributes are ENABLED. **Note**: Once `featureStatus` flag is **ENABLED**, both `discoveryStatus` and `accessTokenStatus` forwarding cannot be disabled.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
`forwardingRulesConfig`	While enabling the forwarding rules feature, this attribute is configured prior or during enabling the feature.	**DataType**: array (ForwardingRulesConfig) **Constraints**: Maximum of 50 forwarding rules can be configured. **Default Value**: Empty array

Table 2-38 ForwardingRulesConfig

Attribute	Description	Details
`targetNfType`	This attribute defines target NF type in a forwarding rule.	**DataType**: string **Constraints**: It has to be either 3gpp defined NF type, custom NF type following regular expression (^CUSTOM_([A-Za-z0-9_]+)), or a wildcard *. **Default Value**: NA
`serviceNames`	List of services allowed for target NF type.	**DataType**: array (string) **Constraints**: Cannot be empty, either wildcard or any service name(s) must be defined. **Note**: Maximum of 20 service names can be configured for each forwarding rule. **Default Value**: NA
`serviceNamesMatchType`	This attribute provides details on how the service names are evaluated, based on the defined constraints. **Exact**: Service Names in incoming request must be present in the configured service names. **Anyone**: Service Names in incoming request must match with any one of the configured service names. **Note**: If serviceNames is a wildcard attribute, then this attribute can be skipped.	**DataType**: string **Constraints**: EXACT, ANYONE **Default Value**: NA

Table 2-39 Preloaded records for errorResponses

errorCondition	responseCode	errorResponse	errorCause	retryAfter	redirectUrl
NRF_Not_Reachable	504	NRF not reachable	UNSPECIFIED_NF_FAILURE	5m	See ErrorInfo.

Table 2-39 (Cont.) Preloaded records for errorResponses

errorCondition	responseCode	errorResponse	errorCause	retryAfter	redirectUrl
NRF_Forwarding_Loop_Detection	508	Loop Detected	UNSPECIFIED_NF_FAILURE	5m	See ErrorInfo.

2.8 SLF Options

This section provides REST API configuration parameter details to configure NRF Subscriber Location Function (SLF) options.

URI: *{apiRoot}/nrf-configuration/v1/slfOptions*

Method: PUT and GET

- **PUT**: Updates NRF SLF options configuration.
- **GET**: Retrieves NRF SLF options configuration.

Content Type: application/json

Body:

```
{
    "slfLookupConfig": [{
        "nfType": "UDM",
        "preferredSubscriberIdType": "SUPI",
        "skipSLFLookupParameters": ["group-id-list"],
        "exceptionListForMissingMandatoryParameter": ["routing-indicator"]
    }],
    "slfHostConfig": [{
        "nfInstanceId": "c56a4180-65aa-42ec-a945-5fd21dec0538",
        "apiVersions": [{
            "apiVersionInUri": "v1",
            "apiFullVersion": "15.5.0"
        }],
        "scheme": "http",
        "host": "ocudrSlf-1-ingressgateway.ocnrf.svc.cluster.local",
        "priority": 100,
        "port": 80
    }],
    "rerouteOnResponseHttpStatusCodes": {
        "pattern":"^[3,5][0-9]{2}$",
        "codeList": null
    },
    "featureStatus": "ENABLED",
    "slfConfigMode": "DISCOVERED_SLF_CONFIG_MODE",
    "preferredSLFLocality": "",
    "populateSlfCandidateList": false,
    "slfDiscoveredCandidateList": [],
    "useOAuthToken": false,
    "accessTokenCacheEnabled": true,
    "preferredPortFromIPEndpoint": DISABLED,
    "preferredRoutingParameter": [
```

```
            {
               "priority": 1,
               "routingParameter": "Ipv4Address"
            },
            {
               "priority": 2,
               "routingParameter": "Ipv6Address"
            },
            {
               "priority": 3,
               "routingParameter": "Fqdn"
            }],
    "errorResponses": [{
            "errorCondition": "SLF_Missing_Mandatory_Parameters",
            "responseCode": 400,
            "errorResponse": "Mandatory parameter missing for SLF Lookup",
            "errorCause": "MANDATORY_QUERY_PARAM_MISSING",
            "retryAfter": "5m",
            "redirectUrl": ""
        }, {
            "errorCondition": "SLF_Subscriber_Not_Provisioned",
            "responseCode": 200,
            "errorResponse": "Subscriber not provisioned in SLF",
            "errorCause": "",
            "retryAfter": "5m",
            "redirectUrl": ""
        }, {
            "errorCondition": "SLF_Not_Reachable",
            "responseCode": 504,
            "errorResponse": "SLF not reachable",
            "errorCause": "UNSPECIFIED_NF_FAILURE",
            "retryAfter": "5m",
            "redirectUrl": ""
        },
        {
            "errorCondition": "SLF_OAuthToken_Failure",
            "responseCode": 500,
            "errorResponse": "SLF OauthToken Failure Occurred",
            "errorCause": "UNSPECIFIED_NF_FAILURE",
            "retryAfter": "5m",
            "redirectUrl": ""
        }
    ]
}
```

Configuration Attributes

> **Note**
>
> If any attribute is not present in the JSON request body while updating, the existing value in the database is preserved and used. At least one attribute is included during the PUT request.

Table 2-40 slfOptions

Attribute Name	Description	Details
`featureStatus`	Enables or disables the SLF Feature. • If the GroupId is already present in Search Query, then the value of featureStatus is also ENABLED. NRF uses the Group Id received and not communicate to SLF or UDR. • If Subscriber Id is present in Search Query, it is ignored and not be used to perform discovery search in NRF.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
`slfConfigMode`	This attribute decides whether the SLF lookup can be performed based on preconfigured slfHostConfig configuration or UDR registered at the NRF. STATIC_SLF_CONFIG_MODE: If this value is set, the SLF lookup is performed based on preconfigured slfHostConfig. DISCOVERED_SLF_CONFIG_MODE: If this value is set, the SLF lookup is performed based on SLF or UDR registered with NRF. **Note**: The `slfHostConfig` must be configured before or while setting the `slfConfigMode` as STATIC_SLF_CONFIG_MODE when featureStatus is ENABLED. Once the featureStatus is ENABLED, and the `slfConfigMode` is set to STATIC_SLF_CONFIG_MODE, the `slfHostConfig` cannot be empty. If `slfConfigMode` is set to DISCOVERED_SLF_CONFIG_MODE, `slfHostConfig` is not considered for discovery query. The `slfConfigMode` can be set to DISCOVERED_SLF_CONFIG_MODE only if there is atleast one slfCandidate present in the `slfDiscoveredCandidateList`. To trigger the population of `slfDiscoveredCandidateList`, `populateSlfCandidateList` must be set to **true**.	**DataType**: string **Constraints**: STATIC_SLF_CONFIG_MODE, DISCOVERED_SLF_CONFIG_MODE **Default Value**: STATIC_SLF_CONFIG_MODE
`preferredSLFLocality`	Indicates the preferred locality for SLF or UDR. When this attribute is configured, the SLF or UDR profile belonging to this locality is given the highest priority.	**DataType**: string **Constraints**: NA **Default Value**: ""
`populateSlfCandidateList`	This attribute triggers the creation of the `slfDiscoveredCandidateList` when set to **true**. **Note**: If `slfConfigMode` is set to **DISCOVERED_SLF_CONFIG_MODE**, this attribute cannot be set back to **false**.	**DataType**: boolean **Constraints**: true, false **Default Value**: false

Table 2-40 (Cont.) slfOptions

Attribute Name	Description	Details
`slfDiscoveredCandidateList`	This attribute contains the list of SLF or UDR profiles registered with the NRF which is used to send the SLF query when the SLF feature is enabled and `slfConfigMode` is set to DISCOVERED_SLF_CONFIG_MODE. While updating the existing SLF options, remove the `slfDiscoveredCandidateList` and `slfConfigMode` attributes from the body, and then perform a PUT request. To enable the SLF feature in dynamic SLF mode, see the "Subscriber Location Function" section in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*. **Note**: This is a read-only attribute. For more information, see slfDiscoveredCandidate.	**DataType**: array **Constraints**: NA **Default Value**: []
`slfHostConfig`	This attribute is used to configure Primary and Secondary SLF details for forwarding various requests. It allows to configure SLF details such as apiVersion, scheme, host, port, and so on. The only supported value for apiVersionInUri is v1. Hence the apiVersions attribute must have at least one data record with apiVersionInUri attribute value set as v1. This configuration allows you to configure more than two SLF Details. SLF with the highest priority is considered as Primary SLF for forwarding requests. SLF with the second highest priority is considered as Secondary SLF for forwarding. If supportedNfTypeList is set, then operator must set this attribute. This is because this value is used to contact the network function hosting the SLF. To reset this attribute, send an empty array, for example: "slfHostConfig": [ ] If this attribute is already set, then there is no need to provide the value again. **Note**: The value of this attribute can be FQDN, IPv4, or IPv6. This attribute is used only if `slfConfigMode` is set to STATIC_SLF_CONFIG_MODE. For more information, see Table 2-4.	**DataType**: array **Constraints**: NA **Default Value**: []
`slfLookupConfig`	This attribute defines details for SLF lookup. Different exception lists, preferredSubscriberIdType, and SLF lookup skip can be configured per NFType. While enabling the SLF feature, this attribute is configured before or while enabling the feature.	**DataType**: array **Constraints**: Maximum 30 configurations can be done, see table SLFLookupConfig for details. **Default Value**: []

Table 2-40 (Cont.) slfOptions

Attribute Name	Description	Details

`rerouteOnResponseHttpStatusCodes`	This attribute is used to determine if SLF retry must be performed to an alternate SLF based on the response code received from the SLF. The alternate SLF is picked from the SLF Host Config, if `slfConfigMode` is set to STATIC_SLF_CONFIG_MODE or from the slfDiscoveredCandidateList if `slfConfigMode` is set to DISCOVERED_SLF_CONFIG_MODE. For more information, see ResponseHttpStatusCodes.	**DataType**: string
		Constraints: pattern or codeList.
		Default Value:
		`{`
		`"pattern": "^[3,5][0-9]{2}$",`
		`"codeList": null`
		`}`

`useOAuthToken`	This attribute is used while performing the SLF query to SLF or UDR to query the Access Token service to get Oauth access token details. If the value of this attribute is true, the SLF function of NRF accesses the Access Token service. If the value of this attribute is false, the SLF function does not access the Access Token service and performs the SLF query without Oauth Access Token.	**DataType**: boolean **Constraints**: true or false **Default Value**: false
`preferredPortFromIPEndpoint`	This attribute indicates if the ports defined in the ipEndpoints of the NfService must be used for routing where chosen routing parameter is not having port explicitly defined. If set to **ENABLED**, the port is picked from the ipEndpoints. If no port is present in the ipEndPoints, then NRF will fall back to use the scheme for determining the port. If set to **DISABLED**, the scheme is used for determining the port. If NfService.scheme is set to http, port 80 is used If NfService.scheme is set to https, port 443 is used. **Note**: For port selection, the first ipEndPoints configured with **only port** (without IPv4Addresses/IPv6Addresses) is used as port for routing.	**DataType**: boolean **Constraints**: DISABLED, ENABLED **Default Value**: DISABLED
`preferredRoutingParameter`	This attribute indicates the priority order of the routing parameters, Ipv4Address, Ipv6Address, and Fqdn, to be used for routing SLF requests. If the most preferred attribute is not present in the NfProfile, then the next available preferred attribute is selected for routing. **Note**: The highest preference is given for the routing parameter to which the lowest priority value is set. For more information, see Table 2-41.	**DataType**: array **Constraints**: • Array size should be equal to 3. • No duplicate priority values are allowed. • Routing Parameter value must be Ipv4Address, Ipv6Address or Fqdn. **Default Value**: As mentioned in Table 2-41.

Table 2-40 (Cont.) slfOptions

Attribute Name	Description	Details
accessTokenCacheEnabled	When this attribute is set to true, NRF will cache the oAuth2 token for SLF communication and use it until the token is expired. When this attribute is set to false, for each SLF query NRF generates a new oAuth2 token. **Note**: The operator must set this attribute to true after a successful NRF upgrade.	**DataType**: boolean **Constraints**: true or false **Default Value**: true
errorResponses	This attribute defines the error responses which may be sent during SLF processing. This attribute allows the operator to update the error response code and error response description for preloaded error conditions. For more information, see PreLoaded records for errorResponses.	**DataType**: array **Constraints**: NA **Default Value**: As mentioned in PreLoaded records for errorResponses.

Table 2-41 PreferredRoutingParameter

Parameter	Description	Details
priority	Indicates the priority for the routing parameter. **Note**: No duplicate priority values are allowed.	**DataType**: string **Constraints**: 1, 2, 3 **Default Value**: NA
routingParameter	Indicates the routing parameter assigned for the specific priority. **Note**: Routing parameter value should be either Ipv4Address, Ipv6Address or Fqdn.	**DataType**: integer **Constraints**: Ipv4Address, Ipv6Address, Fqdn **Default Value**: NA

Table 2-42 PreLoaded records for errorResponses

errorCondition	responseCode	errorResponse	errorCause	retryAfter	redirectUrl
SLF_Missing_Mandatory_Parameters	400	Mandatory parameter missing for SLF Lookup	MANDATORY_QUERY_PARAM_MISSING	5m	See ErrorInfo.
SLF_Not_Reachable	504	SLF not reachable	UNSPECIFIED_NF_FAILURE	5m	See ErrorInfo.
SLF_Subscriber_Not_Provisioned	200	Subscriber not provisioned in SLF	""	5m	See ErrorInfo.
SLF_OAuthToken_Failure	500	SLF OauthToken Failure Occurred	UNSPECIFIED_NF_FAILURE	5m	See ErrorInfo.

Table 2-43 SLFLookupConfig

Attribute	Description	Details
nfType	NF Type for which SLF need to be supported.	**DataType**: string **Constraints**: This value can be 3GPP defined NF type or Custom NFtype. **Default Value**: NA

Table 2-43 (Cont.) SLFLookupConfig

Attribute	Description	Details
preferredSubscriberIdType	This attribute is only used to pick one Subscriber Identifier if more than one subscriber identifiers (SUPI, GPSI) are present in the NFDiscover service operation message.	**DataType**: string **Constraints**: SUPI or GPSI **Default Value**: NA
exceptionListForMissingMandatoryParameter	This attribute indicates that if one of the elements of this arrayed attribute is present in Discovery Search Query, then the discovery query is processed without rejection even if the mandatory parameter for SLF query (SUPI or GPSI) is NOT present in the discovery request.	**DataType**: array (string) **Constraints**: Maximum 10 elements can be present in this list attribute. **Default Value**: NA
skipSLFLookupParameters	This attribute indicates that if one of the elements of this arrayed attribute is present in the discovery search query then SLF lookup is skipped, and 3GPP defined discovery lookup is performed by ignoring subscriber Id attributes (SUPI or GPSI) from the discovery message.	**DataType**: array (string) **Constraints**: Maximum 10 elements can be present in this list attribute. **Default Value**: NA

Table 2-44 slfDiscoveredCandidate

Attribute	Datatype	Description
nfInstanceId	String	The NfInstanceId of the registered UDR profile.
nfSetIdList	array(String)	The NfSetIdList of the registered UDR profile. **Note**: Only the first NfSetId in this list is included in the *3gpp-Sbi-Discovery-target-nf-set-id header.*
capacity	Integer	If the profile is present in the 'nudr-group-id-map' service, the capacity is set from the 'nudr-group-id-map', else it is set from the NfProfile. If not present in either NfProfile or 'nudr-group-id-map' service, the capacity will not be set.
priority	Integer	If the profile is present in the 'nudr-group-id-map' service, the priority is set from the 'nudr-group-id-map', else it is set from the NfProfile. If not present in either NfProfile or 'nudr-group-id-map' service, the priority will not be set.
load	Integer	If the profile is present in the 'nudr-group-id-map' service, the load is set from the 'nudr-group-id-map', else it is set from the NfProfile. If not present in either NfProfile or 'nudr-group-id-map' service, the load will not be set.
fqdn	String	The FQDN of the registered UDR profile **Note**: The order of priority for picking the *fqdn* from the registered UDR profile is as follows: • Fqdn from 'nudr-group-id-map' service if present. • Fqdn from NfProfile if present. See the **Note**.
ipv4Address	array(String)	The IPv4 address of the registered UDR profile **Note**: The order of priority for picking the *ipv4Address* from the registered UDR profile is as follows: • ipEndpoint.ipv4Address from 'nudr-group-id-map' service if present. • ipv4Addresses from NfProfile if present. • Only the first ipv4Address in the list is currently used. See the **Note**.

Table 2-44 (Cont.) slfDiscoveredCandidate

Attribute	Datatype	Description
`ipv6Address`	array(String)	The IPv6 address of the registered UDR profile **Note**: The order of priority for picking the *ipv6Address* from the registered UDR profile is as follows: • ipEndpoint.ipv6Address from 'nudr-group-id-map' service if present. • ipv6Addresses from NfProfile if present. • Only the first ipv6Address in the list is currently used. See the **Note**.
`oauth2Required`	Boolean	This attribute indicates if Oauth2-based authorization is required. This value is set to `slfOptions.useOAuthToken`.
`locality`	String	The locality of the registered UDR profile.

> **Note**
>
> Atleast one of the attributes fqdn, ipv4Address or ipv6Address must be included by the registered UDR.
>
> For routing, the service level attributes is considered first. If none of the attributes are present at service level, then the profile level attributes are considered.
>
> The Producer NF is selected in the following sequence of the three attributes:
>
> `ipv4Adress, ipv6Address, fqdn`

2.9 Georedundancy Options

This section provides REST API configuration parameter details to configure NRF georedundancy options.

URI: *{apiRoot}/nrf-configuration/v1/geoRedundancyOptions*

Method: PUT and GET

- **PUT**: Updates NRF georedundancy options configuration.
- **GET**: Retrieves NRF georedundancy options configuration.

Content Type: application/json

Body:

```
{
            "featureStatus": "DISABLED",
            "monitorDBReplicationStatusInterval": "5s",
            "monitorNrfServiceStatusInterval": "5s",
            "replicationDownTimeTolerance": "10s",
            "replicationLatency": "5s",
            "replicationLatencyThreshold": "20s",
            "replicationStatusUri": "null",
```

```
            "replicationUpTimeTolerance": "10s",
            "siteNameToNrfInstanceIdMappingList":
[{"siteName":"Site-2","nrfInstanceId":"6faf1bbc-6e4a-4454-a507-
a14ef8e1bc5d"}],
            "useRemoteDataWhenReplDown": true
}
```

Configuration Attributes

> **Note**
>
> If any attribute is not present in the JSON request body while updating, the existing value in the database is preserved and used. At least one attribute is included during the PUT request.

Table 2-45 geoRedundancyOptions

Attribute Name	Description	Details
`featureStatus`	Enables or disables the georedundancy feature in NRF.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
`replicationLatency`	The default replication Latency of the replication channel. This value will be used only if the actual replication channel latency is not reported by the DBTier Replication Service. This value must always be lesser than the configured value of replicationLatencyThreshold. The value is in pHqMrS format, where p, q, r are integers and H, M, S or h,m,s denotes hours, minutes, and seconds respectively.	**DataType**: string **Constraints**: 1s - 10m **Default Value**: 5s
`monitorNrfServiceStatusInterval`	The attribute defines the time interval for monitoring the aggregated Nf_Management service status (combined status of nfRegistration, nfSubscription, and nrfAuditor service). The value is in pHqMrS format, where p, q, r are integers and H, M, S or h,m,s denotes hours, minutes, and seconds respectively.	**DataType**: string **Constraints**: 1s - 10m **Default Value**: 5s
`monitorDBReplicationStatusInterval`	This attribute defines the time interval for monitoring the DB replication status. The value is in pHqMrS format, where p, q, r are integers and H, M, S or h,m,s denotes hours, minutes, and seconds respectively.	**DataType**: string **Constraints**: 1s - 10m **Default Value**: 5s
`replicationDownTimeTolerance`	The attribute defines the minimum time for the reported replication channel status to remain as DOWN before NRF considers the replication status is DOWN. The value is in pHqMrS format, where p, q, r are integers and H, M, S or h,m,s denotes hours, minutes, and seconds respectively.	**DataType**: string **Constraints**: 1s - 3m **Default Value**: 10s

Table 2-45 (Cont.) geoRedundancyOptions

Attribute Name	Description	Details
`replicationUpTimeTolerance`	The attribute defines the minimum time for the reported replication channel status to remain as UP before NRF considers the replication status is UP. The value is in pHqMrS format, where p, q, r are integers and H, M, S or h,m,s denotes hours, minutes, and seconds respectively.	**DataType**: string **Constraints**: 1s - 3m **Default Value**: 10s
`replicationLatencyThreshold`	The attribute defines the maximum allowed replication channel latency beyond which the replication channel status is considered as DOWN. The value is in pHqMrS format, where p, q, r are integers and H, M, S or h,m,s denotes hours, minutes, and seconds respectively.	**DataType**: string **Constraints**: 1s - 10m **Default Value**: 20s
`useRemoteDataWhenReplDown`	This attribute specifies whether the remote NRF records are considered for NRF service operations when the DB replication is down. **Note**: This attribute only impacts the read-only service requests. The service requests in this category are NfDiscovery, NfAccessToken, NfProfileRetrieval, and NfListRetrieval. When replication channel status is down, the mate NRF records are not used for processing the service requests.	**DataType**: boolean **Constraints**: true or false **Default Value**: true
`siteNameToNrfInstanceIdMappingList`	The attribute specifies the list of NRF Instance Id and the corresponding DBTier site name of the remote site(s). The attribute *"nrfInstanceId"* is configured as per the value of *global.nrfInstanceId* of the ***REMOTE*** site NRF. The attribute *"siteName"* is configured as per the value of the remote DBTier site name. Following is the sample configuration at site Chicago which is georedundant with sites Atlantic (SiteName: atlantic, NrfInstanceId: 723da493-528f-4bed-871a-2376295c0020) and Pacific (SiteName: pacific, NrfInstanceId: cfa780dc-c8ed-11eb-b8bc-0242ac130003) `"siteNameToNrfInstanceIdMappingList": [{"siteName":"atlantic", "nrfInstanceId":"723da493-528f-4bed-871a-2376295c0020"}, {"siteName":"pacific", "nrfInstanceId":"cfa780dc-c8ed-11eb-b8bc-0242ac130003"} ]}` **Note**: It is mandatory only if the georedundancy feature is enabled.	**DataType**: array (SiteNameToNrfInstanceIdMapping) **Constraints**: 1..N **Default Value**: NA

Table 2-45 (Cont.) geoRedundancyOptions

Attribute Name	Description	Details
`replicationStatusUri`	The URI that is used to query the DB replication status. **Note**: It is mandatory only if the georedundancy feature is enabled. The URI is defined as: `http://<appinfo-svc>:<appinfo-port>/status/category/replicationstatus` Where, `<appinfo-svc>` is the appinfo service name. `<appinfo-port>` is the port of appinfo service. For more information about using appinfo microservice to fetch the replication status, see "NRF Georedundancy" section in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.	**DataType**: string **Constraints**: NA **Default Value**: null

Table 2-46 SiteNameToNrfInstanceIdMapping

Attribute Name	Description	Details
`siteName`	Represents site name.	**DataType**: string **Presence**: M **Cardinality**: 1
`nrfInstanceId`	Represents nrfInstanceId of NRF.	**DataType**: string **Presence**: M **Cardinality**: 1

2.10 NF Authentication Options

This section provides REST API configuration parameter details to configure NRF NF authentication options.

URI: *{apiRoot}/nrf-configuration/v1/nfAuthenticationOptions*

Method: PUT and GET

- **PUT**: Updates NRF NF authentication options configuration.
- **GET**: Retrieves NRF NF authentication options configuration.

Content Type: application/json

Body:

```
{
    "nfRegistrationStatus": "DISABLED",
    "nfSubscriptionStatus": "DISABLED",
    "nfDiscoveryStatus": "DISABLED",
    "accessTokenStatus": "DISABLED",
    "nfProfileRetrievalStatus": "DISABLED",
    "nfListRetrievalStatus": "DISABLED",
```

```
    "checkIfNfIsRegistered": "DISABLED",
    "errorResponses": [{
    "errorCondition": "Nf_Fqdn_Authentication_Failure",
       "responseCode": 403,
       "errorResponse": "Failed to authenticate NF using FQDN",
       "errorCause": "UNSPECIFIED_MSG_FAILURE",
       "retryAfter": "5m",
       "redirectUrl": ""
   }]
}
```

Configuration Attributes

> **Note**
>
> If any attribute is not present in the JSON request body while updating, the existing value in the database is preserved and used. At least one attribute is included during the PUT request.

Table 2-47 nfAuthenticationOptions

Attribute Name	Description	Details
`nfRegistrationStatus`	This attribute controls the authentication of consumer NF for NFRegister, NFUpdate, and NFDeregister service operations. If the value of this attribute is set as ENABLED, then the identity of consumer NF is validated. If the value of this attribute is set as DISABLED, then the validation is not performed for consumer NF.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
`nfSubscriptionStatus`	This attribute controls the authentication of consumer NF for NFStatusSubscribe and NFStatusUnsubscribe service operations. If the value of this attribute is set as ENABLED, then the identity of consumer NF is validated, and NRF allows the subscription only if the NF is registered with NRF. If the value of this attribute is set as DISABLED, then the validation is not performed for consumer NF.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
`nfDiscoveryStatus`	This attribute controls the authentication of consumer NF for NFDiscover service operation. If the value of this attribute is set as ENABLED, then the identity of consumer NF is validated. If the value of this attribute is set as DISABLED, then the validation is not performed for consumer NF. In case NF identity is not present in discovery request messages then validation is performed as per the `checkIfNfIsRegistered` attribute.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
`accessTokenStatus`	This attribute controls the authentication of consumer NF for AccessToken service operation. If the value of this attribute is set as ENABLED, then the identity of consumer NF is validated. If the value of this attribute is set as DISABLED, then the validation is not performed for consumer NF.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED

Table 2-47 (Cont.) nfAuthenticationOptions

Attribute Name	Description	Details
nfProfileRetrievalStatus	This attribute controls the authentication of consumer NF for NF Profile Retrieval service operation. If the value of this attribute is set as ENABLED, then the identity of consumer NF is validated. If the value of this attribute is set as DISABLED, then the validation is not performed for consumer NF.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
nfListRetrievalStatus	This attribute controls the authentication of consumer NF for NF List Retrieval service operation. If the value of this attribute is set as ENABLED, then the identity of consumer NF is validated. If the value of this attribute is set as DISABLED, then the validation is not performed for consumer NF.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
errorResponses	This attribute defines the error responses which may be sent for NF Authentication scenarios. This attribute will allow to update the response code, error response description, retryAfter, and redirectUrl for preloaded error conditions.	**DataType**: array (ErrorInfo) **Constraints**: ENABLED, DISABLED **Default Value**: See PreLoaded values for more details.
checkIfNfIsRegistered	This attribute controls the mechanism to check if NF is registered or not with NRF. If the value of this attribute is set as ENABLED, then the validation is performed. If the value of this attribute is set as DISABLED, then the validation is not performed for consumer NF. • Discovery request does not contain requester-nf-instance-fqdn and the value of nfDiscoveryAuthenticationStatus is set as ENABLED.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED

Table 2-48 PreLoaded records for errorResponses

errorCondition	responseCode	errorResponse	errorCause	retryAfter	redirectUrl
Nf_Fqdn_Authentication_Failure	403	Failed to authenticate NF using FQDN	UNSPECIFIED_MSG_FAILURE	5m	See ErrorInfo.

2.11 Logging Level Options

NRF allows retrieving or updating of logs levels of each service. It also allows to retrieve the log levels of all the services together, including the common services like Ingress Gateway, Egress Gateway, and appinfo.

Retrieving all service level logs

Configuration Attributes for all service level logs

URI:*{apiRoot}/nrf-configuration/v1/all/logging*

Method: GET

Content Type: application/json

Body:

```
[
{
    "nfAccessToken":"{\"appLogLevel\":\"WARN\",\"packageLogLevel\":
[{\"packageName\":\"root\",\"logLevelForPackage\":\"WARN\"}]}",
    "nfDiscovery":"{\"appLogLevel\":\"WARN\",\"packageLogLevel\":
[{\"packageName\":\"root\",\"logLevelForPackage\":\"WARN\"},
{"packageName":"cache","logLevelForPackage":"WARN"}]}",
    "nfRegistration":"{\"appLogLevel\":\"WARN\",\"packageLogLevel\":
[{\"packageName\":\"root\",\"logLevelForPackage\":\"WARN\"}]}",
    "nfSubscription":"{\"appLogLevel\":\"WARN\",\"packageLogLevel\":
[{\"packageName\":\"root\",\"logLevelForPackage\":\"WARN\"}]}",
    "nrfArtisan": "{\"appLogLevel\":\"WARN\",\"packageLogLevel\":
[{\"packageName\":\"root\",\"logLevelForPackage\":\"WARN\"}]}",
    "nrfAuditor":"{\"appLogLevel\":\"WARN\",\"packageLogLevel\":
[{\"packageName\":\"root\",\"logLevelForPackage\":\"WARN\"}]}",
    "nrfConfiguration":"{\"appLogLevel\":\"WARN\",\"packageLogLevel\":
[{\"packageName\":\"root\",\"logLevelForPackage\":\"WARN\"}]}",
    "nrfCacheData":"{\"appLogLevel\":\"WARN\",\"packageLogLevel\":
[{\"packageName\":\"root\",\"logLevelForPackage\":\"WARN\"}]}",
    "ingressGateway":"{\"appLogLevel\":\"WARN\",\"packageLogLevel\":
[{\"packageName\":\"root\",\"logLevelForPackage\":\"WARN\"},
{\"packageName\":\"oauth\",\"logLevelForPackage\":\"WARN\"}]}",
    "egressGateway":"{\"appLogLevel\":\"WARN\",\"packageLogLevel\":
[{\"packageName\":\"root\",\"logLevelForPackage\":\"WARN\"},
{\"packageName\":\"oauth\",\"logLevelForPackage\":\"WARN\"}]}",
    "appInfo":"{\"appLogLevel\":\"INFO\"}",
    "perfinfo":"{"appLogLevel":"WARN","packageLogLevel":
[{"packageName":"root","logLevelForPackage":"WARN"}]}"
    "altRoute":"{"appLogLevel":"WARN","packageLogLevel":
[{"packageName":"root","logLevelForPackage":"WARN"}]}"
}
]
```

Table 2-49 allLoggingOptions

Attribute	DataType	Presence	Description
nfAccessToken	string	M	Specifies the log level options for nfAccessToken microservice.
nfDiscovery	string	M	Specifies the log level options for nfDiscovery microservice.
nfRegistration	string	M	Specifies the log level options for nfRegistration microservice.
nfSubscription	string	M	Specifies the log level options for nfSubscription microservice.
nrfArtisan	string	M	Specifies the log level options for artisan microservice.
nrfAuditor	string	M	Specifies the log level options for nrfAuditor microservice.
nrfConfiguration	string	M	Specifies the log level options for nrfConfiguration microservice.
nrfCacheData	string	M	Specifies the log level options for Cache Data Service.

Table 2-49 (Cont.) allLoggingOptions

Attribute	DataType	Presence	Description
ingressGateway	string	M	Specifies the log level options for Ingress Gateway.
egressGateway	string	M	Specifies the log level options for Egress Gateway.
appInfo	string	M	Specifies the log level options for appinfo.
altRoute	string	M	Specifies the log level options for alternate-route service
perfinfo	string	M	Specifies the log level options for Perf Info

Retrieving log levels at service level

Configuration example for nfAccessToken Logging

URI: *{apiRoot}/nrf-configuration/v1/nfAccessToken/logging*

Method: PUT

Content Type: application/json

Body:

```
{  "appLogLevel":"WARN",
   "packageLogLevel":[{
         "packageName":"root",
         "logLevelForPackage":"WARN"
         }]
}
```

Configuration example for nfDiscovery Logging

URI: *{apiRoot}/nrf-configuration/v1/nfDiscovery/logging*

Method: PUT

Content Type: application/json

Body:

```
{  "appLogLevel":"WARN",
   "packageLogLevel":[{
         "packageName":"root",
         "logLevelForPackage":"WARN"
         },
         {"packageName":"cache",
         "logLevelForPackage":"WARN"
      }]
}
```

Configuration example for nfRegistration Logging

URI: *{apiRoot}/nrf-configuration/v1/nfRegistration/logging*

Method: PUT

Content Type: application/json

Body:

```
{  "appLogLevel":"WARN",
   "packageLogLevel":[{
         "packageName":"root",
         "logLevelForPackage":"WARN"
         }]
}
```

Configuration example for nfSubscription Logging

URI: *{apiRoot}/nrf-configuration/v1/nfSubscription/logging*

Method: PUT

Content Type: application/json

Body:

```
{  "appLogLevel":"WARN",
   "packageLogLevel":[{
         "packageName":"root",
         "logLevelForPackage":"WARN"
         }]
}
```

Configuration example for nrfArtisan Logging

URI: *{apiRoot}/nrf-configuration/v1/nrfArtisan/logging*

Method: PUT

Content Type: application/json

Body:

```
{  "appLogLevel":"WARN",
   "packageLogLevel":[{
         "packageName":"root",
         "logLevelForPackage":"WARN"
         }]
}
```

Configuration example for nrfAuditor Logging

URI: *{apiRoot}/nrf-configuration/v1/nrfAuditor/logging*

Method: PUT

Content Type: application/json

Body:

```
{  "appLogLevel":"WARN",
   "packageLogLevel":[{
         "packageName":"root",
```

```
        "logLevelForPackage":"WARN"
        }]
}
```

Configuration example for nrfConfiguration Logging

URI: *{apiRoot}/nrf-configuration/v1/nrfConfiguration/logging*

Method: PUT

Content Type: application/json

Body:

```
{  "appLogLevel":"WARN",
   "packageLogLevel":[{
        "packageName":"root",
        "logLevelForPackage":"WARN"
        }]
}
```

Configuration example for nrfCacheData Logging

URI: *{apiRoot}/nrf-configuration/v1/nrfCacheData/logging*

Method: PUT

Content Type: application/json

Body:

```
{
  "appLogLevel": "WARN",
  "packageLogLevel": [
    {
      "packageName": "root",
      "logLevelForPackage": "WARN"
    },
    {
      "packageName": "cache",
      "logLevelForPackage": "WARN"
    }
  ]
}
```

Configuration example for Ingress Gateway Logging

URI: *{apiRoot}/nrf/nf-common-component/v1/igw/logging*

Method: PUT

Content Type: application/json

Body:

```
{  "appLogLevel":"WARN",
   "packageLogLevel":[{
        "packageName":"root",
```

```
         "logLevelForPackage":"WARN"
         },
         {
        "packageName": "oauth",
        "logLevelForPackage": "WARN"
    }]
}
```

Configuration example for Egress Gateway Logging

URI: *{apiRoot}/nrf/nf-common-component/v1/egw/logging*

Method: PUT

Content Type: application/json

Body:

```
{
    "appLogLevel": "WARN",
    "packageLogLevel": [{
        "packageName": "root",
        "logLevelForPackage": "WARN"
    }, {
        "packageName": "oauth",
        "logLevelForPackage": "WARN"
    }]
}
```

Configuration example for AppInfo Logging

URI: *{apiRoot}/nrf/nf-common-component/v1/appinfo/logging*

Method: PUT

Content Type: application/json

Body:

```
{
    "appLogLevel": "INFO"
}
```

Configuration example for Alternate-Route Logging

URI: *{apiRoot}/nrf/nf-common-component/v1/altRoute/logging*

Method: PUT

Content Type: application/json

Body:

```
{  "appLogLevel":"WARN",
   "packageLogLevel":[{
         "packageName":"root",
         "logLevelForPackage":"WARN"
```

```
            }]
}
```

> **Note**
>
> If any attribute is not present in JSON request body while updating, existing value in database will be preserved and used. At least one attribute shall be included during PUT request.

Table 2-50 Logging

Attribute Name	Description	Details
`appLogLevel`	Specifies the log level of the application.	**DataType**: string **Constraints**: INFO,DEBUG,WARN,ERROR,FATAL,OFF,TRACE **Default Value**: WARN
`packageLogLevel`	Specifies a list of individual packages and their respective log levels.	**DataType**: array (PackageLogLevel) **Constraints**: NA **Default Value**: See PackageLogLevel for more details.

Table 2-51 PackageLogLevel

Attribute Name	Description	Details
`packageName`	Specifies the name of the package.	**DataType**: string **Constraints**: root, cache **Note**: `cache` is for nfDiscovery microservice only. **Default Value**: NA
`logLevelForPackage`	Specifies the log level for the given package.	**DataType**: string **Constraints**: INFO,DEBUG,WARN,ERROR,FATAL,OFF,TRACE **Default Value**: WARN

2.12 Roaming Options

This section provides REST API configuration parameter details to configure NRF roaming options.

URI: *{apiRoot}/nrf-configuration/v1/roamingOptions*

Method: PUT and GET

- **PUT**: Updates NRF roaming options configuration.
- **GET**: Retrieves NRF roaming options configuration.

Content Type: application/json

Body:


```
{
    "featureStatus": "DISABLED",
    "genericInterPlmnFqdn": "nrf.5gc.mnc012.mcc345.pub.3gppnetwork.org",
    "userAgentMandatory": true,
    "notificationAPIversion": "v1",
    "3GPPAPIRootScheme": "http",
    "errorResponses": [{
            "errorCondition": "Mandatory_Attributes_Missing",
            "responseCode": 400,
            "errorResponse": "Mandatory attribute(s) for Roaming are not
present",
            "errorCause": "MANDATORY_QUERY_PARAM_MISSING",
            "retryAfter": "5m",
            "redirectUrl": ""
        }, "errorCondition": "Roaming_Attributes_Present_Feature_Disabled",
            "responseCode": 400,
            "errorResponse": "Roaming attributes are present while Roaming
feature is disabled",
            "errorCause": "UNSPECIFIED_MSG_FAILURE",
            "retryAfter": "5m",
            "redirectUrl": ""
        },{
            "errorCondition": "UserAgent_Header_NotPresent",
            "responseCode": 400,
            "errorResponse": "User agent header not present",
            "errorCause": "MANDATORY_IE_MISSING",
            "retryAfter": "5m",
            "redirectUrl": ""
        },{
            "errorCondition": "Loop_Detected",
            "responseCode": 508,
            "errorResponse": "Loop detected during roaming routing",
            "errorCause": "UNSPECIFIED_NF_FAILURE",
            "retryAfter": "5m",
            "redirectUrl": ""
        }]
]
}
```


Configuration Attributes

> **Note**
>
> If any attribute is not present in the JSON request body while updating, the existing value in the database is preserved and used. At least one attribute is included during the PUT request.

Table 2-52 Roaming Options

Attribute	Description	Details
`featureStatus`	Flag to control roaming feature. If the value is set as ENABLED, the roaming specific routing occurs. If the value is set as DISABLED, the roaming specific routing does not occur.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
`genericInterPlmnFqdn`	Generic FQDN required to be added in Inter-PLMN headers request. Every NRF whether it is vNRF or hNRF role, adds the genericInterPlmnFqdn in the header. This helps to detect the loop. This is a mandatory attribute. **Note**: It is recommended to configure this FQDN value as Inter-PLMN format. Example: nrf.5gc.mnc012.mcc345.pub.3gppnetwork.org **Note**: Length of this attribute is limited upto 255 characters.	**DataType**: string **Constraints**: Format: 5gc.mnc<MNC>.mcc<MCC>.3gppnetwork.org **Default Value**: null
`userAgentMandatory`	This flag is to regulate the consideration to relax or reject the request if the user-agent is not present in the request at home NRF for Access Token Request Validation feature processing. Access Token Request Validation feature is disabled then user-agent header presence is not checked. When the Access Token Request Validation feature is enabled, and this flag value is set as true: • If user-agent header is present, then requester NF type is fetched and used in Access Token Request Validation feature processing. • If user-agent header is not present, then message is rejected. When the Access Token Request Validation feature is enabled, and this flag value is set as false: • If user-agent header is not present, then message is accepted and Access Token Request Validation feature processing is skipped as NFType cannot be determined due to unavailability of data in Request message. **Note**: User-agent header is only applicable when the nfType attribute of the access token request is not present at home NRF.	**DataType**: boolean **Constraints**: true, false **Default Value**: true

Table 2-52 (Cont.) Roaming Options

Attribute	Description	Details
`notificationAPIversion`	This attribute specifies the version of Notification Server that is required as part of 3gpp-Sbi-Callback header input. This is a mandatory attribute for Roaming feature. Example: v1 **Note**: Length of this attribute is limited upto 255 characters.	**DataType**: string **Constraints**: NA **Default Value**: null
`3GPPAPIRootScheme`	This attribute defines the scheme which is used while constructing the 3GPP API root header.	**DataType**: string **Constraints**: http, https **Default Value**: http
`errorResponses`	This attribute defines the error responses which may be sent during NRF roaming traffic processing. This attribute allows configuring the error response code and error response description for preloaded error conditions.	**DataType**: array (errorResponses) **Constraints**: See PreLoaded Error Response values. **Default Value**: NA

Table 2-53 Preloaded values for errorResponses

errorCondition	responseCode	errorResponse	errorCause	retryAfter	redirectUrl
Mandatory_Attributes_Missing	400	Mandatory attribute(s) for roaming are not present.	MANDATORY_QUERY_PARAM_MISSING	5m	See ErrorInfo.
UserAgent_Header_NotPresent	400	User agent header not present.	MANDATORY_IE_MISSING	5m	See ErrorInfo.
Loop_Detected	508	Loop detected during roaming message processing.	UNSPECIFIED_NF_FAILURE	5m	See ErrorInfo.
Roaming_Attributes_Present_Feature_Disabled	400	Roaming attributes are present while roaming feature is disabled.	UNSPECIFIED_MSG_FAILURE	5m	See ErrorInfo.

2.13 NF Screening Options

This section provides REST API configuration parameter details to configure NF screening options.

URI: *{apiRoot}/nrf-configuration/v1/nfScreeningOptions*

Method: PUT and GET

- **PUT**: Updates NF screening options configuration.
- **GET**: Retrieves NF screening options configuration.

Content Type: application/json

Body:

```
{
   "featureStatus": "DISABLED",
   "responseCode": 403
}
```

Configuration Attributes

> **Note**
>
> If any attribute is not present in JSON request body while updating, existing value in database will be preserved and used. Atleast one attribute shall be included during PUT request.

Table 2-54 nfScreeningOptions

Attribute Name	Description	Details
`featureStatus`	This attribute indicates if NF Screening feature is enabled or not globally.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
`responseCode`	This attribute indicates HTTP status code which is returned, if incoming request does not qualify NF screening rules.	**DataType**: integer **Constraints**: NA **Default Value**: 403

2.14 NF Screening Rules Configuration

This section provides REST API configuration parameter details to configure NRF NF screening rules options.

Screening Rules List Update

The following NF screening rules update particular rule configuration (except read only attributes).

URI: *http://host:port/nrf-configuration/v1/screening-rules/CALLBACK_URI*

Method: PUT

Content Type: application/json

Body:

Request Body

```
{
    "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]
            }
        ]
    }
}
```

Screening Rules List Get
The following NF screening rules get all the configured rules.

URI: *http://host:port/nrf-configuration/v1/screening-rules/*

Method: 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"
            ]
        }
    }
]
}
```

Screening Rules List Partial Update

The following NF screening rules get a particular configured rule.

URI: *http://host:port/nrf-configuration/v1/screening-rules/CALLBACK_URI*

Method: 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 partial rule update

The following NF screening rules update the partial rule.

URI: *http://host:port/nrf-configuration/v1/screening-rules/CALLBACK_URI*

Method: PATCH

Content-Type: application/json-patch+json

Request Body:

```
[
    {"op":"remove","path":"/globalScreeningRulesData/nfCallBackUriList/2/
```

```
ports/0"},
    {"op":"replace","path":"/globalScreeningRulesData/failureAction","value":
"CONTINUE"},
    {"op":"add","path":"/nrfScreeningRulesData","value": {"failureAction":
"SEND_ERROR","nfCallBackUriList":
      [{"ipv4AddressRange":{"start" : "189.163.192.10","end":
"190.178.127.10"}}]}}
 ]
```

Configuration Attributes

Table 2-55 nfScreeningRules

Attribute name	Data type	Presence	Description
nfScreeningRulesListType	array(NfScreeningRulesListType)	C	ReadOnly. It is returned while retrieving the rule.
nfScreeningType	array(NfScreeningType)	M	Screening type of complete screening list. All the rules can be either Blacklist or Whitelist.
nfScreeningRulesListStatus	array(NfScreeningRulesListStatus)	M	Enables or disables complete screening list.
globalScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if global screening rules need to be configured.
customNfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for custom NF need to be configured.
nrfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for NRF need to be configured.
udmScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for UDM need to be configured.
amfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for AMF need to be configured.
smfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for custom SMF need to be configured.
ausfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for AUSF need to be configured.
nefScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for NEF need to be configured.
pcfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for PCF need to be configured.
nssfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for NSSF need to be configured.
udrScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for UDR need to be configured.
lmfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for IMF need to be configured.
gmlcScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for GMLC need to be configured.
fiveG_EirScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for EIR need to be configured.
seppScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for SEPP need to be configured.

Table 2-55 (Cont.) nfScreeningRules

Attribute name	Data type	Presence	Description
upfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for UPF need to be configured.
n3iwfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for IWF need to be configured.
afScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for AF need to be configured.
udsfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for UDSF need to be configured.
bsfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for BSF need to be configured.
chfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for CHF need to be configured.
nwdafScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for NWDAF need to be configured.
slfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for SLF need to be configured.
cbcfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for CBCF need to be configured.
nssaafScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for NSSAAF need to be configured.
mmeScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for MME need to be configured.
icscfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for ICSCF need to be configured.
scsasScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for SCSAS need to be configured.
draScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for DRA need to be configured.
ucmfScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for UCMF need to be configured.
sorafScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for SOR_AF need to be configured.
spafScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for SPAF need to be configured.
scefScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for SCEF need to be configured.
imsasScreeningRulesData	array(NfScreeningRulesData)	O	This attribute is present if screening rules for IMS_AS need to be configured.

Table 2-56 NfScreeningRulesData

Attribute name	Data type	Presence	Description
failureAction	FailureAction	M	Indicates what action needs to be taken during failure.
nfFqdn	NfFqdn	C	This attribute is present if screeningListType is NF_FQDN.

Table 2-56 (Cont.) NfScreeningRulesData

Attribute name	Data type	Presence	Description
nfCallBackUriList	array(NfCallBackUri)	C	This attribute is present if screeningListType is CALLBACK_URI.
nfIpEndPointList	array(NfIpEndPoint)	C	This attribute is present if screeningListType is NF_IP_ENDPOINT.
plmnList	array(PlmnId)	C	This attribute is present if screeningListType is PLMN_ID.
nfTypeList	array(string)	C	This attribute is present if screeningListType is NF_TYPE_REGISTER.

Table 2-57 NfScreeningRulesListType

Enumeration value	Description
"NF_FQDN"	Screening List type for NF FQDN. This screening rule applies at NF service level of a NF profile in NF_Register service operation.
"NF_IP_ENDPOINT"	Screening list type for IP Endpoint. This screening rule type is applicable for ipv4address, ipv6address attributes at NF Profile level and ipEndPoint attribute at nfServices level for NF_Register service operation.
"CALLBACK_URI"	Screening list type for callback URIs in NF Service and nfStatusNotificationUri in SubscriptionData. This screening rule type is applicable for defaultNotificationSubscription attribute at NF service level for NF_Register service operation. This is also applicable for nfStatusNotificationUri attribute of SubscriptionData for NFStatusSubscribe service operation.
"PLMN_ID"	Screening list type for PLMN ID.
"NF_TYPE_REGISTER"	Screening list type for allowed NF Types to register. NRF supports 3GPP TS 29510 Release 15 and specific Release 16 NF Types. For more information on the supported NF Types list, see "Supported NF Types" section in *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

Table 2-58 NfScreeningType

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. **Note**: It is not supported for NF Screening rule NF_TYPE_REGISTER.
"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 2-59 NfScreeningRulesListStatus

Enumeration value	Description
"ENABLED"	Screening List feature is enabled to apply the rules.
"DISABLED"	Screening List feature is disabled.

Table 2-60 FailureAction

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

Table 2-61 NfFqdn

Attribute Name	Data Type	Presence	Description
fqdn	array(FQDN)	C	Exact FQDN to be matched. This is conditional, at least one attribute is present.
pattern	array(string)	C	Regular expression for FQDN. This is conditional, at least one attribute is present.

Table 2-62 NfIpEndPoint

Attribute Name	Data Type	Presence	Description
ipv4Address	Ipv4Addr	C	IPv4 address to be matched. See the below **NOTE**.
ipv4AddressRange	Ipv4AddressRange	C	Range of IPv4 addresses. See the below **NOTE**.
ipv6Address	Ipv6Addr	C	IPv6 address to be matched. See the below **NOTE**.
ipv6AddressRange	Ipv6AddressRange	C	Range of IPv6 addresses. See the below **NOTE**.
port	array(integer)	O	If this attribute is not configured, then it is not considered for validation.
portRange	array(PortRange)	O	If this attribute is not configured, then it is not considered for validation.

> **Note**
>
> The condition is only one attribute must be present (either ipv4Address, ipv4AddressRange, ipv6Address, or ipv6AddressRange).

Table 2-63 NfCallBackUri

Attribute Name	Data Type	Presence	Description
fqdn	FQDN	C	Exact FQDN to be matched. See the below **NOTE**.
pattern	string	C	Regular Expression for FQDN, Ipv4Address, Ipv6Address. See the below **NOTE**.
ipv4Address	Ipv4Addr	C	IPv4 address to be matched. See the below **NOTE**.
ipv4AddressRange	Ipv4AddressRange	C	Range of IPv4 addresses. See the below **NOTE**.
ipv6Address	Ipv6Addr	C	IPv6 address to be matched. See the below **NOTE**.
ipv6AddressRange	Ipv6AddressRange	C	Range of IPv6 addresses. See the below **NOTE**.
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**
>
> The condition is only one attribute must be present (either ipv4Address, ipv4AddressRange, ipv6Address, or ipv6AddressRange).

Table 2-64 PortRange

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

2.15 DNS NAPTR Update Options Configuration

This section provides REST API configuration parameter details to update Name Authority Pointer (NAPTR) record in Domain Name System (DNS) during Access and Mobility Functions (AMF) registration, update, and deregistration.

2.15.1 DNS NAPTR Configuration in Alternate Route Service

This URI can be used to configure alternate route service for DNS NAPTR.

URI: */{nfType}/nf-common-component/v1/{serviceName}/upstreamdnsconfig*

/{nfType}/nf-common-component/v1/{serviceName}/{instanceId}/upstreamdnsconfig

Method: GET, PUT

Content Type: application/json

Body:

```
{
    "enabled": false,
    "watchSecretTimeout": 2000,
    "fixedTsigKeyMonitorDelay": 5000,
    "tsigKeyNamespace": "ocnrf",
    "tsigKeySecretName": "tsig-secret",
    "host": "10.75.175.222",
    "port": "53",
  "zone": "example.search",
"upstreamDNSTimeout": 10000
}
```

Configuration Attributes

Table 2-65 Upstream DNS Server

Attribute Name	Description	Details
`enabled`	Enables or disables the update or delete of DNS NAPTR record in DNS Server.	**DataType**: boolean **Constraints**: true, false **Default Value**: false
`watchSecretTimeout`	This configuration is to watch event timeout (in second) while reading the secret.	**DataType**: integer **Constraints**: It should be always 2000 or 3000 ms less than the `fixedTsigKeyMonitorDelay` **Default Value**: 2000
`fixedTsigKeyMonitorDelay`	This configuration is for monitoring interval to secret added or updated.	**DataType**: integer **Constraints**: NA **Default Value**: 5000
`tsigKeyNamespace`	Indicates the namespace in which transaction signature key secret is created.	**DataType**: string **Constraints**: NA **Default Value**: ocnrf
`tsigKeySecretName`	Indicates the transaction signature key secret name.	**DataType**: string **Constraints**: NA **Default Value**: tsig-secret
`host`	This configuration is host IP of the DNS server.	**DataType**: string **Constraints**: NA **Default Value**: NA
`port`	This configuration is port of the DNS server.	**DataType**: string **Constraints**: NA **Default Value**: NA
`zone`	This configuration is zone of the DNS server.	**DataType**: string **Constraints**: NA **Default Value**: example.search
`upstreamDNSTimeout`	This configuration is set timeout for a upstream DNS server transaction.	**DataType**: string **Constraints**: NA **Default Value**: 10000

2.15.2 DNS NAPTR Update Options

This URI can be used to configure DNS NAPTR Update at NRF.

URI: *{apiRoot}/nrf-configuration/v1/dnsNaptrUpdateOptions*

Method:

- GET: Retrieves NAPTR record from DNS configuration.
- PUT: Updates NAPTR record in DNS configuration.

Content Type: application/json

Body:

```
{
    "featureStatus": "DISABLED",
    "maxRetryCount":  2,
    "defaultPriority": "100",
    "defaultCapacity":"65435"
}
```

Configuration Attributes

> **Note**
>
> If any attribute is not present in the JSON request body while updating, the existing value in the database is preserved and used. At least one attribute is included during the PUT request.

Table 2-66 DNS-NAPTR Update Options

Attribute Name	Description	Details
`featureStatus`	Enables or disables the DNS NAPTR update. If the value of this attribute is ENABLED, then NRF performs DNS NAPTR update during AMF NFRegister, NFUpdate, NFDeregister service operations, and suspension of NF by NRF. If the value of this attribute is DISABLED, then NRF does not perform any DNS NAPTR update.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
`maxRetryCount`	This attribute defines the maximum number of retries in case DNS NAPTR update fails.	**DataType**: integer **Constraints**: 0-10 **Default Value**: 2
`defaultPriority`	The value of this attribute is considered as NF priority in case NFProfile does not have a priority attribute.	**DataType**: integer **Constraints**: 0 - 65535 **Default Value**: 100
`defaultCapacity`	The value of this attribute is considered as NF capacity in case NFProfile does not have the capacity attribute.	**DataType**: integer **Constraints**: 0 - 65535 **Default Value**: 65435

2.15.3 DNS NAPTR Status API

This section explains the status API for DNS NAPTR records.

URI: *{apiRoot}/nrf-status-data/v1/dnsNAPTRRecords*

Method: GET

Content Type: application/json

Body:

```
{
    "dnsNAPTRRecordStatus": [{
        "amfSetFqdn":
"set1ab.region23.amfset.5gc.mnc014.mcc310.3gppnetwork.org",
        "amfInstanceId": "bbab9915-c8bd-47dd-9438-14709bc2b452",
        "capacity": 63535,
        "priority": 20,
        "amfName": "amf1.cluster1.net2.amf.5gc.mnc014.mcc310.3gppnetwork.org",
        "syncStatus": "DNSRecordCreated",
        "operationType": "update",
        "amfFqdn": "AMF.d5g.oracle.com",
        "arecord": ["192.168.3.110"]
    }]
}
```

Table 2-67 DNS NAPTR Record Status

Attribute Name	Description	Details
`dnsNAPTRRecordStatus`	This attribute defines the DNS NAPTR Record details along with their status. The value for this attribute is populated in GET response.	**DataType**: array (DnsNAPTRUpdateRecord) **Constraints**: NA **Default Value**: Not applicable

Table 2-68 DnsNAPTRUpdateRecord

Attribute Name	Description	Details
`amfSetFqdn`	Domain name or FQDN for this NAPTR	**DataType**: string **Constraints**: NA **Default Value**: NA
`amfInstanceId`	This is 3GPP attribute. This is kept in Status response, to map NF Instance	**DataType**: string **Constraints**: NA **Default Value**: NA
`capacity`	Preference value is used to break the tie when order is same among the two AMF NFs. In the NAPTR record this value is: (Maximum capacity of AMF) - Declared Capacity of AMF) It means value will be 65535 - capacity	**DataType**: integer **Constraints**: NA **Default Value**: NA

Table 2-68 (Cont.) DnsNAPTRUpdateRecord

Attribute Name	Description	Details
priority	Determines which record is processed first. It processes the record with the lowest value first.	**DataType**: integer **Constraints**: NA **Default Value**: NA
amfName	Replacement field specifies the FQDN for the next lookup, if it was not specified in the regular expression	**DataType**: string **Constraints**: NA **Default Value**: NA
syncStatus	This attribute tells about DNS NAPTR record status with NRF. Possible values:- • DNSRecordStatusPending - DNS Record is not created yet or status is not known yet • DNSRecordCreated - DNS Record is created successfully • DNSRecordCreateFailed - DNS Record creation failed • DNSRecordDeleted - DNS Record deleted • DNSRecordDeleteFailed - DNS Record deletion failed • DNSRecordNotFound - DNS Record not found in DNS Server • DNSRecordMismatch - DNS Record mismatch with NRF and DNS Server • DNSRecordError - DNS Record Error other than above	**DataType**: string **Constraints**: NA **Default Value**: NA
operationType	DNS NAPTR record entry with NRF and specific operation type. • update: record with NRF sent to DNS Server for updating the DNS NAPTR record. • delete: record with NRF sent to DNS Server for deleting the DNS NAPTR record.	**DataType**: string **Constraints**: update, delete **Default Value**: NA
amfFqdn	FQDN of the AMF.	**DataType**: string **Constraints**: NA **Default Value**: NA
arecord	Indicates the IPv4 Endpoint Address.	**DataType**: integer **Constraints**: NA **Default Value**: NA
aaaarecord	Indicates the IPv6 Endpoint Address.	**DataType**: integer **Constraints**: NA **Default Value**: NA

Table 2-69 Mapping of DnsNAPTRUpdateRecord with DNS NAPTR and 3GPP Attributes

Attribute in DnsNAPTRUpdateRecord	Mapped Attribute in DNS NAPTR	Mapped Attribute in 3GPP
amfSetFqdn	Lookup domain name	This attribute is built using various attributes from AMF Profile. The AMF Set FQDN is constructed as follows: set<AMF Set Id>.region<AMF Region Id>.amfset.5gc.mnc<MNC>.mcc<MCC>.3gppnetwork.org Where, AMF Set Id:- amfSetId of AMFInfo AMF Region Id:- amfRegionId of AMFInfo MCC and MNC values are extracted from amfName attribute of n2InterfaceAmfInfo (AmfInfo)
amfInstanceId	No mapped attribute	NfInstanceId of AMF (NFProfile)
capacity	Preference In the NAPTR record this value is: (Maximum capacity of AMF) - Declared Capacity of AMF) It means value will be 65535 - capacity	capacity (NFProfile)
priority	Order	priority (NFProfile)
amfName	Replacement	amfName (AmfInfo)
syncStatus	None	None
amfFqdn	None	fqdn (NFProfile)
arecord	IPv4 address	ipv4EndpointAddress (n2InterfaceAmfInfo)
aaaarecord	IPv6 address	ipv6EndpointAddress (n2InterfaceAmfInfo)

> **Note**
>
> There can be additional attributes in DNS NAPTR record with specific values and does not get impacted by 3GPP attribute. For example, `class = 1, ttl = 0, flag = A, regexp = “”, service = x-3gpp-amf:x-n2`.

2.15.4 DNS NAPTR Retrigger API

This section explains the retrigger API for DNS NAPTR records. This API is used to retrigger the operations on any particular `NfInstanceId` provided in DNS NAPTR Status API output.

URI: *{apiRoot}/nrf-status-data/v1/reTriggerNAPTRUpdate*

Method: PUT

Content Type: application/json

Body:

```
{ "reTriggerNAPTRUpdateRecords":["9e21368f-
fd27-4d64-8dc4-70dc1529c319","9e21368f-fd27-4d64-8dc4-70dc1529c319"]}
```

```
Sample response codes with details:
204 STATUS CODE with No content: If all of the NfInstanceIds from the request
are present with OCNRF.
404 STATUS CODE with Problem details: If any one of the NfInstanceId from the
request is not present with OCNRF.

GET Method: It is supported and retained for future use. Currently, 204 No
Content response is sent for GET method.
```

Table 2-70 DNS NAPTR Retrigger API

Attribute Name	Description	Details
`reTriggerNAPTRUpdateRecords`	This attribute defines AMF NFInstanceIds for which retrigger needs to be performed.	**DataType**: array (NFInstanceIds) **Constraints**: NfInstanceIds up to 10. Mandatory attribute. **Default Value**: NA

2.16 Pod Protection Options

This section provides REST API configuration parameter details to configure pod protection options for NRF subscription microservice.

URI: *{apiRoot}/nrf-configuration/v1/nfSubscription/podProtectionOptions*

Method:

- GET: Retrieves NRF pod protection options configuration.
- PUT: Enables or Disables NRF pod protection feature.

Content Type: application/json

Body:

```
{
  "enabled": true,
  "monitoringInterval": 1200,
  "congestionControl": {
    "enabled": true,
    "stateChangeSampleCount": 2,
    "actionSamplingPeriod": 2,
    "states": [
      {
        "name": "Normal",
        "weight": 0,
        "entryAction": [
          {
            "action": "MaxConcurrentStreamsUpdate",
            "arguments": {
              "incrementBy": 30,
              "incrementByActionSamplingPeriod": 3,
              "maxConcurrentStreamsPerCon": 100
            }
          },
          {
```

```
        "action": "AcceptIncomingConnections",
        "arguments": {
          "accept": true
        }
      }
    ]
  },
  {
    "name": "DoC",
    "weight": 1,
    "resourceThreshold": {
      "cpu": 60,
      "pendingMessage": 100
    },
    "entryAction": [
      {
        "action": "AcceptIncomingConnections",
        "arguments": {
          "accept": false
        }
      },
      {
        "action": "MaxConcurrentStreamsUpdate",
        "arguments": {
          "incrementBy": 30,
          "incrementByActionSamplingPeriod": 3,
          "decrementBy": 30,
          "decrementByActionSamplingPeriod": 1,
          "maxConcurrentStreamsPerCon": 50
        }
      }
    ]
  },
  {
    "name": "Congested",
    "weight": 2,
    "resourceThreshold": {
      "cpu": 75,
      "pendingMessage": 150
    },
    "entryAction": [
      {
        "action": "AcceptIncomingConnections",
        "arguments": {
          "accept": false
        }
      },
      {
        "action": "MaxConcurrentStreamsUpdate",
        "arguments": {
          "decrementBy": 30,
          "decrementByActionSamplingPeriod": 1,
          "maxConcurrentStreamsPerCon": 10
        }
      }
    ]
```

```
            }
          ]
        }
      }
```

Table 2-71 Pod Protection Options

Attribute	Description	Details
`enabled`	This attribute indicates if the Pod Protection feature is enabled or disabled.	**DataType**: Boolean **Constraints**: true, false **Default Value**: false
`monitoringInterval`	This attribute indicates the periodicity at which the overload state is monitored. **Unit**: Milliseconds **Note:** This is a read-only attribute.	**DataType**: Integer **Constraints**: NA **Default Value**: 1200
`congestionControl`	This attribute specifies the congestion control configuration. For more information about congestion control parameters, see Table 2-72.	**DataType**: Object **Constraints**: NA **Default Value**: NA

Table 2-72 CongestionControlConfig

Attribute	Description	Details
`enabled`	This attribute allows the configuration of pod protection attributes for the nfSubscription pods. **Note**: This must be set to true for Pod Protection feature.	**DataType**: Boolean **Constraints**: true, false **Default Value**: false
`actionSamplingPeriod`	This attribute indicates the interval at which the configured action must be considered. The actions are configured under `entryAction.action` attribute. The interval is calculated as (`actionSamplingPeriod` * `monitoringInterval`). **Note:** This is a read-only attribute.	**DataType**: Integer **Constraints**: NA **Default Value**: 2
`stateChangeSampleCount`	This attribute indicates the number of times the pod must be in the particular congestion state before transitioning to another state. For example, if the current state is normal, and the new state is DoC, then NRF moves the pod to DoC only if the state is reported for 10 times in 1 second (`stateChangeSampleCount` * `monitoringInterval`). **Note:** This is a read-only attribute.	**DataType**: Integer **Constraints**: NA **Default Value**: 2
`states`	This attribute indicates the congestion states, the thresholds, and corresponding actions. For more information about congestion states, see Table 2-73.	**DataType**: Object **Constraints**: NA **Default Value**: NA

Table 2-73 CongestionStates

Attribute	Description	Details
`name`	The name of the congestion state. • Normal: The pod is not in overload state. • Danger of Congestion (Doc): The pod is about to go into the congested state. Actions configured in the entryAction is performed. • Congested state: The pod is in congested state. Actions configured in the entryAction is performed. **Note:** This is a read-only attribute.	**DataType**: String **Constraints**: Normal, Doc, Congested **Default Value**: NA
`weight`	The weight of the congestion state. The weight indicates the critical of the congestion state. The lower the value, the lower the criticality. **Note:** This is a read-only attribute.	**DataType**: Integer **Constraints**: NA **Default Value**: Normal= 0, DoC= 1, and Congested= 2
`entryAction`	This attribute indicates the actions for the congestion state. For more information about the entry action configuration, see Table 2-74. **Note:** This is a read-only attribute.	**DataType**: List **Constraints**: NA **Default Value**: NA
`resourceThreshold`	This attribute indicates the resource thresholds for the given congestion state. This configuration is mandatory for the 'DoC' and 'Congested' states. For more information about the threshold for each resources, see Table 2-75. **Note:** This is a read-only attribute.	**DataType**: Object **Constraints**: NA **Default Value**: NA

Table 2-74 EntryActionConfig

Attribute	Description	Details
`action`	This attribute indicates the action for the congestion state. • `AcceptIncomingConnections`: The action indicates whether the incoming new connection is accepted or rejected based on the overload state. • `MaxConcurrentStreamsUpdate`: The action indicates whether to increase or decrease the max concurrent stream for all incoming connections till the maxConcurrentStreamsPerCon is reached.	**DataType**: String **Constraints**: MaxConcurrentStreamsUpdate, AcceptIncomingConnections **Default Value**: NA
`arguments`	This attribute indicates the actions for the congestion state. For more information about the arguments, see Table 2-76.	**DataType**: Map <String, Object> **Constraints**: NA **Default Value**: NA

Table 2-75 ResourceThreshold

Attribute	Description	Details
cpu	The CPU threshold is expressed in percentage. **Note:** This is a read-only attribute.	**DataType**: Integer **Constraints**: NA **Default Value**: For DoC, the default CPU is 60. For Congested, the default CPU is 75.
pendingMessageCount	The number of messages pending to be processed, expressed in absolute value. **Note:** This is a read-only attribute.	**DataType**: Integer **Constraints**: NA **Default Value**: For DoC, the default count is 100. For Congested, the default count is 150.

Table 2-76 Possible Arguments

Attribute	Description	Details
accept	The attribute indicates if the incoming connection should be accepted or not. Applicable when the action is AcceptIncomingConnections. true: The incoming connection is accepted. false: The incoming connection is rejected.	**DataType**: Boolean **Constraints**: true, false **Default Value**: For Normal state, the default value is true. For DoC and Congested state, the default value is false.
incrementBy	The attribute indicates the factor by which the current concurrent streams value will be incremented till it reaches maxConcurrentStreamsPerCon. **Note**: This is preconfigured for Normal and DoC state.	**DataType**: Integer **Constraints**: NA **Default Value**: 30
decrementBy	The attribute indicates the factor by which the current concurrent streams value will be decremented till it reaches maxConcurrentStreamsPerCon. **Note**: This is preconfigured for DoC and Congested state.	**DataType**: Integer **Constraints**: NA **Default Value**: 30
maxConcurrentStreamsPerCon	The attribute indicates the maximum number of concurrent streams per connection allowed.	**DataType**: Integer **Constraints**: NA **Default Value**: For Normal, the default value is 100. For DoC, the default count is 50. For Congested, the default count is 10.
decrementByActionSamplingPeriod	The attribute indicates the time interval at which the decrementBy is applied to reach maxConcurrentStreamsPerCon. If not provided, the actionSamplingPeriod is used. **Note**: This is preconfigured for DoC and Congested state.	**DataType**: Integer **Constraints**: NA **Default Value**: 1
incrementByActionSamplingPeriod	The attribute indicates the time interval at which the incrementBy is applied to reach maxConcurrentStreamsPerCon. If not provided, the actionSamplingPeriod is used. **Note**: This is preconfigured for Normal and DoC state.	**DataType**: Integer **Constraints**: NA **Default Value**: 30

2.17 Controlled Shutdown Options

This section provides REST API configuration parameter details to configure Controlled Shutdown options.

URI: *{apiRoot}/nrf-configuration/v1/controlledShutdownOptions*

Method:

- GET: Retrieves the operational state.
- PUT: Updates the operational state.

Content Type: application/json

Body:

```
{
  "operationalState": "NORMAL"
}
```

Table 2-77 Controlled Shutdown Options

Attribute	Description	Details
`operationalState`	The operational state of NRF. If the controlled shutdown feature is disabled (global Helm attribute *global.enableControlledShutdown* is set to *false*), then the operator will not be able to perform a controlled shutdown operation and NRF responds with *403 response*.	**DataType**: string **Constraints**:NORMAL, COMPLETE_SHUTDOWN **Default Value**: NORMAL

2.17.1 Operational State History

The below API is used to retrieve the history of the operational state changes. This API lists the last 5 operational state changes. The topmost entry indicates the latest change in the operational state.

URI: *{apiRoot}/nrf-configuration/v1/operationalStateHistory*

Method: GET: Retrieves operational state history.

Content Type: application/json

Body:

```
{
    "operationalStateHistory":
    [
        {
            "operationalState": "NORMAL",
            "timeStamp": "2022-08-30T08:12:45.88519",
            "status": "SUCCESS"
        },
        {
            "operationalState": "COMPLETE_SHUTDOWN",
```

```
                "timestamp": "2023-02-02 13:42:08.148351775",
                "status": "SUCCESS"
            },
            {
                "operationalState": "NORMAL",
                "timestamp": "2023-02-02 13:35:07.111230791",
                "status": "SUCCESS"
            },
            {
                "operationalState": "COMPLETE_SHUTDOWN",
                "timestamp": "2023-02-02 13:33:22.06605161",
                "status": "SUCCESS"
            }
    ]
}
```


Table 2-78 Operational State History

Attribute	Description	Details
`operationalState`	The operational state value of the NRF. This is a read-only attribute.	**DataType**: string **Constraints**:NORMAL, COMPLETE_SHUTDOWN **Default Value**: NA
`timestamp`	Logs the timestamp when the operational state event took place. This is a read-only attribute.	**DataType**: Date **Constraints**: NA **Default Value:** NA
`status`	Indicates if the operational state was changed successfully. • Success: Successful in switching of operational state • Failure: Fail in switching of operational state. This is a read-only attribute.	**DataType**: string **Constraints**: SUCCESS, FAILURE **Default Value:** NA

2.18 NRF Growth Options

This section provides REST API configuration parameter details to set or retrieve the NRF Growth feature configuration.

URI: *{apiRoot}/nrf-configuration/v1/nrfGrowth/featureOptions*

Method:

- GET: Retrieves NRF Growth configuration.
- PUT: Updates NRF Growth configuration.

Content Type: application/json

Sample Body:


```
{
  "featureStatus": "ENABLED",
  "nfSetId": "set101.nrfset.5gc.mnc012.mcc345",
  "nrfHostConfig": {
```

```
    "hostConfigMode": "STATIC_MODE",
    "staticNrfConfigList": [
      {
        "nfSetId": "set101.nrfset.5gc.mnc012.mcc345",
        "nrfHostConfigList": [
          {
            "nfInstanceId": "6faf1bbc-6e4a-4454-a507-a14ef8e1bc5c",
            "apiVersions": [
              {
                "apiVersionInUri": "v1",
                "apiFullVersion": "15.5.0"
              }
            ],
            "scheme": "http",
            "host": "ocnrf-ingressgateway.ocnrf.svc.cluster.local",
            "priority": 0,
            "port": 80
          }
        ]
      },
      {
        "nfSetId": "set201.nrfset.5gc.mnc012.mcc345",
        "nrfHostConfigList": [
          {
            "nfInstanceId": "6faf1bbc-6e4a-4454-a507-a14ef8e1bc5d",
            "apiVersions": [
              {
                "apiVersionInUri": "v1",
                "apiFullVersion": "15.5.0"
              }
            ],
            "scheme": "http",
            "host": "ocnrf-1-ingressgateway.ocnrf.svc.cluster.local",
            "priority": 1,
            "port": 80
          }
        ]
      }
    ]
  },
  "nrfRerouteProfile": {
    "rerouteAttempts": 1,
    "httpStatusCodeList": {
      "pattern": "^[3,5][0-9]{2}$",
      "codeList": null
    }
  }
}
```

Table 2-79 featureOptions

Attribute	Description	Details
`featureStatus`	This is an optional parameter. This attribute is used to enable or disable the NRF Growth feature. `featureStatus` value cannot be **ENABLED**, if nfSetId is null and nrfHostConfig is not configured. If the value of this attribute is set as **ENABLED**, then the state data from multiple sets is used for processing service requests.	**DataType**: String **Constraints**: NA **Default Value**: DISABLED **Range**: ENABLED or DISABLED
`nfSetId`	This is a mandatory parameter. This attribute indicates the NRF set ID to which the NRF belongs. The value of this is the same for all the georedundant sites. The value for this attribute is set as per 3GPP TS 29.571 v16.7.0.	**DataType**: String **Constraints**: maximum length is 255. **Default Value**: null **Range**: NA
`nrfHostConfig`	This is a mandatory parameter. This attribute is used to configure the NRF host details of the remote NRF set. See Table 2-80 for details.	**DataType**: Array **Constraints**: NA **Default Value**: NA **Range**: NA
`nrfRerouteProfile`	The attribute indicates the number of reroutes to be performed when the NRF of a remote NRF set is unreachable. See Table 2-82 for details.	**DataType**: Array **Constraints**: NA **Default Value**: NA **Range**: NA

Table 2-80 nrfHostConfig

Attribute	Description	Details
`hostConfigMode`	This is a mandatory parameter. The attribute defines the mode in which the configurations are used. **STATIC_MODE**: When this value is set, the NRF hosts configured in the `staticNrfConfigList` is used. **Note**: STATIC_MODE is the only mode supported in this release.	**DataType**: String **Constraints**: NA **Default Value**: STATIC_MODE **Range**: NA
`staticNrfConfigList`	This is a mandatory parameter. This attribute is used to configure the host details of the NRFs from the remote NRF set. The attribute must have at least two entries to enable the feature. One of the entries must contain the details of the local NRF Set. See Table 2-81 for details. **Note**: A maximum of three NRF sets can be configured. In each NRF set, a maximum of four hosts per set are allowed.	**DataType**: Array **Constraints**: NA **Default Value**: NA **Range**: NA

Table 2-81 staticNrfConfigList

Attribute	Description	Details
`nfSetId`	This is a mandatory parameter. This attribute represents the unique ID of the NRF set. The value of this is the same for all the georedundant sites. The value for this attribute is set as per 3GPP TS 29.571 v16.7.0.	**DataType**: String **Constraints**: maximum length is 255. **Default Value**: NA **Range**: NA
`nrfHostConfigList`	This is a mandatory parameter. This attribute is used to configure the NRF host details of the remote NRF set(s). See Table 2-4 for details.	**DataType**: Array **Constraints**: NA **Default Value**: NA **Range**: NA

Table 2-82 nrfRerouteProfile

Attribute	Description	Details
`rerouteAttempts`	This is an optional parameter. This attribute indicates the number of alternate NRF reroutes when the retry attempts are exhausted.	**DataType**: Integer **Constraints**: NA **Default Value**: 1 **Range**: 0-3
`httpStatusCodeList`	This is an optional parameter. This attribute indicates the HTTP status codes to which retry and reroute must be attempted.	**DataType**: Array **Constraints**: NA **Default Value**: "pattern": "^[3,5][0-9]{2}$" **Range**: NA

2.18.1 Forwarding Options for NRF Growth

This section provides REST API configuration parameter details to set or retrieve the forwarding configuration when the NRF Growth feature is enabled.

> **Note**
>
> When the growth feature is enabled, NRF-NRF Forwarding Options configurations are not considered.

URI: *{apiRoot}/nrf-configuration/v1/nrfGrowth/nrfForwardingOptions*

Method:

- GET: Retrieves forwarding options for NRF Growth feature.
- PUT: Updates forwarding options for NRF Growth feature.

Content Type: application/json

Sample Body:

```
{
  "profileRetrievalStatus": "DISABLED",
  "subscriptionStatus": "DISABLED",
```

```
"discoveryStatus": "DISABLED",
"accessTokenStatus": "DISABLED",
"nrfHostConfig": {
  "hostConfigMode": "STATIC_MODE",
  "staticNrfConfigList": [
    {
      "segmentId": "Segment-1",
      "nfSetId": "set401.nrfset.5gc.mnc012.mcc345",
      "priority": 0,
      "nrfHostConfigList": [
        {
          "nfInstanceId": "c56a4180-65aa-42ec-a945-5fd21dec0540",
          "apiVersions": [
            {
              "apiVersionInUri": "v1",
              "apiFullVersion": "15.5.0"
            }
          ],
          "scheme": "http",
          "host": "ocnrf-1-ingressgateway.ocnrf.svc.cluster.local",
          "priority": 0,
          "port": 80
        },
        {
          "nfInstanceId": "c56a4180-65aa-42ec-a945-5fd21dec0541",
          "apiVersions": [
            {
              "apiVersionInUri": "v1",
              "apiFullVersion": "15.5.0"
            }
          ],
          "scheme": "http",
          "host": "ocnrf-2-ingressgateway.ocnrf.svc.cluster.local",
          "priority": 1,
          "port": 80
        }
      ]
    }
  ]
},
"nrfRerouteOnResponseHttpStatusCodes": {
  "pattern": "^[3,5][0-9]{2}$",
  "codeList": null
},
"errorResponses": [
  {
    "errorCondition": "NRF_Not_Reachable",
    "responseCode": 504,
    "errorResponse": "NRF not reachable",
    "errorCause": "UNSPECIFIED_NF_FAILURE",
    "retryAfter": "5m",
    "redirectUrl": ""
  },
  {
    "errorCondition": "NRF_Forwarding_Loop_Detection",
    "responseCode": 508,
```

```
        "errorResponse": "Loop Detected",
        "errorCause": "UNSPECIFIED_NF_FAILURE",
        "retryAfter": "5m",
        "redirectUrl": ""
      }
    ],
    "forwardingRulesFeatureConfig": {
      "featureStatus": "ENABLED",
      "forwardingRulesConfig": [
        {
          "targetNfType": "UDM",
          "serviceNames": [
            "nudm-uecm"
          ],
          "serviceNamesMatchType": "ANYONE"
        },
        {
          "targetNfType": "*",
          "serviceNames": [
            "UDMname14",
            "UDMname15"
          ],
          "serviceNamesMatchType": "EXACT"
        }
      ]
    }
  }
```

Table 2-83 nrfForwardingOptions

Attribute Name	Description	Details
`profileRetrievalStatus`	This attribute controls the forwarding of NFProfileRetrieval service operation messages. If the flag is set to true and NRF is unable to complete the request due to the unavailability of any matching profile, then NRF forwards the NfProfileRetrieval request to the configured NRF host(s) and relays the response received from forwarding NRF to the Consumer NF. If the flag is set to false, NRF will not forward the NfProfileRetrieval request. It returns a response to the consumer NF without forwarding it.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED

Table 2-83 (Cont.) nrfForwardingOptions

Attribute Name	Description	Details
subscriptionStatus	This attribute controls the forwarding of NFStatusSubscribe, and NFStatusUnsubscribe service operation messages. If the flag is set to true and NRF cannot complete the request due to the unavailability of any matching profile, then NRF forwards the NfStatusSubscribe or NfStatusUnSubscribe request to the configured NRF host(s) and relays the response received from forwarding NRF to the Consumer NF. If the flag is false, NRF will not forward the NFStatusSubscribe or NFStatusUnSubscribe request. It returns a response to the consumer NF without forwarding it. **Note**: NFStatusSubscribe forwarding is supported only if Subscription Condition is NfInstanceIdCond in the NFStatusSubscribe request.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
discoveryStatus	This attribute controls the forwarding of NFDiscover service operation messages. If the flag is set to ENABLED and NRF is not able to complete the request due to unavailability of any matching profile, then NRF forwards the NFDiscover request to the configured NRF host(s) and relays the response received from forwarding NRF to the Consumer NF. If flag is DISABLED, NRF will not forward the NFDiscover request in any case. It will return a response to consumer NF without forwarding it.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
accessTokenStatus	This attribute controls the forwarding of AccessToken service operation messages. If the flag is set to ENABLED and NRF is not able to complete the request due to unavailability of any matching Producer NF, then NRF forwards the AccessToken request to the configured NRF host(s) and relays the response received from forwarding NRF to the Consumer NF. If flag is DISABLED, NRF will not forward the AccessToken request in any case. It will return a response to consumer NF without forwarding it.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
nrfHostConfig	This is a mandatory parameter. This attribute is used to configure the NRF host details of the remote NRF set. See Table 2-84 table for details. **Note**: The value of this attribute can be FQDN, IPv4, or IPv6.	**DataType**: array **Constraints**: NA **Default Value**: NA
nrfRerouteOnResponseHttpStatusCodes	This configuration is used to determine if the service operation message needs to be forwarded to remote NRF set. The local NRF set receives a response. If the response status code matches the configured response status code list, then NRF reroutes the request to the remote NRF set. Refer Table 2-84 for details for local and remote NRF set details.	**DataType**: ResponseHttpStatusCodes **Constraints**: pattern or specific code list **Default Value**: "pattern": "^[3,5][0-9]{2}$\|408$"

Table 2-83 (Cont.) nrfForwardingOptions

Attribute Name	Description	Details
errorResponses	This attribute defines the error responses which may be sent during NRF Forwarding scenarios. This attribute will allow to update the error response code and error response description for preloaded error conditions. See Table 2-88 table for details.	**DataType**: array **Constraints**: NA **Default Value**: NA
forwardingRulesFeatureConfig	This attribute provide details for Forwarding Rules feature configuration. See Table 2-86 table for details.	**DataType**: Array **Constraints**: NA **Default Value**: NA

Table 2-84 nrfHostConfig

Attribute	Description	Details
hostConfigMode	This is a mandatory parameter. The attribute defines the mode in which the configurations are used. STATIC_MODE: The NRF hosts configured in the staticNrfConfigList is used.	**DataType**: String **Constraints**: NA **Default Value**: STATIC_MODE **Range**: NA
staticNrfConfigList	This is a mandatory parameter. This attribute is used to configure the host details of the NRFs from the remote NRF set. The attribute must have at least 2 entries to enable the feature. One of the entries must contain the details of the own NRF Set. See Table 2-85 for details. **Note**: A maximum of 3 NRF sets can be configured. In each NRF set, a maximum of 4 hosts are allowed.	**DataType**: Array **Constraints**: NA **Default Value**: NA **Range**: NA

Table 2-85 staticNrfConfigList

Attribute	Description	Details
segmentId	This is a mandatory parameter. This attribute provides details of the segment information in the deployment. This attribute is the unique identifier of the segment.	**DataType**: String **Constraints**: NA **Default** Value: NA **Range**: NA
nfSetId	This is a mandatory parameter. This attribute provides details of the set information in the deployment. This attribute is the unique identifier of a NRF set.	**DataType**: String **Constraints**: NA **Default Value**: NA **Range**: NA

Table 2-85 (Cont.) staticNrfConfigList

Attribute	Description	Details
priority	This is a mandatory parameter. This attribute indicates the priority of the segment and also the NRF set for which the NRF forwarding is applied.	**DataType**: Integer **Constraints**: NA **Default Value**: NA **Range**: NA
nrfHostConfigList	This is a mandatory parameter. This attribute is used to configure the NRF host details of the remote NRF set. This attribute is used to configure local and remote NRF set details used for forwarding various requests. It allows to configure details of NRF like apiVersion, scheme, host, port, and so on. The only supported value for apiVersionInUri is v1. Hence the apiVersions attribute must have at least one data record with apiVersionInUri attribute values set as v1. See Table 2-4 for details. This configuration allows you to configure more than two NRF Details. NRF with the highest priority is considered as local NRF for forwarding messages. NRF with the second highest priority is considered as remote NRF set for forwarding. To reset this attribute, send an empty array, for example: "nrfHostConfig": [ ] If this attribute is already set, then there is no need to provide the value again. **Note**: The value of this attribute can be FQDN, IPv4, or IPv6.	**DataType**: Array **Constraints**: NA **Default Value**: NA **Range**: NA

Table 2-86 ForwardingRulesFeatureConfig

Attribute	Description	
featureStatus	This attribute enables or disables the evaluation of forwarding eligibility of a service request based on target NF type and service names configured in forwardingRulesConfig. This flag can be ENABLED only if discoveryStatus or accessTokenStatus attributes are ENABLED. **Note**: Once featureStatus flag is **ENABLED**, both discoveryStatus and accessTokenStatus forwarding cannot be disabled.	**DataType**: string **Constraints**: ENABLED, DISABLED **Default Value**: DISABLED
forwardingRulesConfig	While enabling the forwarding rules feature, this attribute is configured prior or during enabling the feature. See ForwardingRulesConfig table for details.	**DataType**: array **Constraints**: Maximum of 50 forwarding rules can be configured. **Default Value**: Empty array

Table 2-87 ForwardingRulesConfig

Attribute	Description	Details
`targetNfType`	This attribute defines target NF type in a forwarding rule.	**DataType**: string **Constraints**: It has to be either 3gpp defined NF type, custom NF type following regular expression (^CUSTOM_([A-Za-z0-9_]+)), or a wildcard *. **Default Value**: NA
`serviceNames`	List of services allowed for target NF type.	**DataType**: array (string) **Constraints**: Cannot be empty, either wildcard or any service name(s) must be defined. **Note**: Maximum of 20 service names can be configured for each forwarding rule. **Default Value**: NA
`serviceNamesMatchType`	This attribute provides details on how the service names are evaluated, based on the defined constraints. **Exact**: Service Names in incoming request must be present in the configured service names. **Anyone**: Service Names in incoming request must match with any one of the configured service names. **Note**: If serviceNames is a wildcard attribute, then this attribute can be skipped.	**DataType**: string **Constraints**: EXACT, ANYONE **Default Value**: NA

Table 2-88 Preloaded records for errorResponses

errorCondition	responseCode	errorResponse	errorCause	retryAfter	redirectUrl
NRF_Not_Reachable	504	NRF not reachable	UNSPECIFIED_NF_FAILURE	5m	See Table 2-5.
NRF_Forwarding_Loop_Detection	508	Loop Detected	UNSPECIFIED_NF_FAILURE	5m	See Table 2-5.

3 NRF Configuration Status and Manage APIs

3.1 NRF Configuration Status REST APIs

The configuration status APIs are used to check AccessToken Signing Key Status. For more information on Key-ID for Access Token, see *Oracle Communications Cloud Native Core, Network Repository Function User Guide*.

Table 3-1 Configuration Status REST APIs

API	HTTP method supported	Description	HTTP response codes
{apiRoot}/nrf-status-data/v1/accessTokenSigningDataStatus	GET	This API fetches Access Token Signing Data Status from NRF. NRF provides option to configure access token signing key and certificate details. Using this API, it can be checked that details provided are valid or not and specific key details can be used to sign the token.	200 OK with **AccessTokenSigningDataStatus**, if Access Token Signing data details found. 200 OK with Empty List <**AccessTokenSigningData** > inside **AccessTokenSigningDataStatus**, if Access Token Signing data details not found.

API example

Sample API:- *{apiRoot}/nrf-status-data/v1/accessTokenSigningDataStatus*

```
Method:- GET
Sample response:-
{
"dataTimeStamp":"2020-11-24T15:55:48.000Z",
"accessTokenSigningKeysCount": 2,
"accessTokenSigningData":[{"keyID": "KeyId01", "privateKey":{ "fileName":
"KeyId01-privateKey.pem", "isValid": true, "invalidReason": null },
                                                "certificate":{ "fileName":
"KeyId01-publicCertificate.crt", "isValid": true, "invalidReason": null,
"expiryTime": "2021-11-24T15:55:48.000Z"}
                          },
                          {"keyID": "KeyId02", "privateKey":{ "fileName":
"KeyId02-privateKey.pem", "isValid": false, "invalidReason": "Key file not
found" },
                                                "certificate":{ "fileName":
"KeyId02-publicCertificate.crt", "isValid": false, "invalidReason": "Key file
not found", "expiryTime":null }
                          },
```

```
                                ]
}
```

Data Models

Table 3-2 AccessTokenSigningDataStatus

Attribute	DataType	Description
dataTimeStamp	string	Time stamp when Data was retrieved
accessTokenSigningKeysCount	integer	Count of keys in response
accessTokenSigningData	array(AccessTokenSigningData)	See AccessTokenSigningData for details

Table 3-3 AccessTokenSigningData

Attribute	DataType	Description
keyID	string	Key Id for the Access Token Signing Data
privateKey	AccessTokenSigningDataDetails	Private key details corresponding to KeyId
certificate	AccessTokenSigningDataDetails	Public Certificate details corresponding to KeyId

Table 3-4 AccessTokenSigningDataDetails

Attribute	DataType	Description
fileName	string	File Name of the Private Key and Public Certificate
isValid	boolean (true or false)	Details provided are valid to use or not.
invalidReason	string	Indicates the reason for key or certificate invalidity when is `isValid` value is set to false.
expiryTime	string	Indicates the validity of the certificate. This attribute is applicable only for certificate.

4 NRF State Data Retrieval APIs

REST API Details

"`apiRoot`" is concatenation of the following parts:

- **scheme**: http, https
- the fixed string "://"
- authority (host and optional port)
  host and port will be CNC Console host and port details

Table 4-1 API Details

API	HTTP method supported	Description	HTTP response codes
{apiRoot}/nrf-state-data/v1/nf-details	GET	This API fetches NF Profile related data. Both query request attributes and query result attributes can be used together to get specific results	200 OK with **NFProfileDetails**, if NF Details found 200 OK with Empty List <**NFProfileDetails**>, if NF Details not found 400 BAD Request, if request is not proper 404 NOT FOUND - If no NF Details found for input attributes and query request attribute is used to get details based on specific attribute 500 INTERNAL ERROR - If any internal error occurred while accessing NRF state data
{apiRoot}/nrf-state-data/v1/subscription-details	GET	This API fetches Subscription related data. Both query request attributes and query result attributes can be used together to get specific results	200 OK - **SubscriptionDetails** , if subscription details found 200 OK - Empty List <**SubscriptionDetails** >, if subscription details not found 400 BAD Request, if request is not proper 404 NOT FOUND - If no NF Details found for input attributes and query request attribute is used to get details based on specific attribute 500 INTERNAL ERROR - If any internal error occurred while accessing NRF state data

Details of nf-details URI

Query request parameters supported by nf-details URI

> **Note**
>
> - These attributes can be used to get results based on specific input attributes. These attribute are optional to API. In case no query request and result attribute is mentioned, then only NFinstance Ids will be provided for all of the NF Profiles.
> - In case complete profile is needed, query request attributes shall contain specific attributes **nf-instance-id** or **nf-fqdn**. Otherwise only NFinstance Ids will be provided for all of the NF Profiles by default and additionally query result attributes can be provided to get additional details.
> - At-most one query request attribute is supported, data is returned based on the request attribute.

Table 4-2 Query request attributes supported by nf-details URI

Name	Data Type	Details	Query Example with API
nf-instance-id	string	NF Instance Id of Network Function	{apiRoot}/nrf-state-data/v1/nf-details**?nf-instance-id=<NF Instance ID>**
nf-fqdn	string	NF Profile FQDN	{apiRoot}/nrf-state-data/v1/nf-details**?nf-fqdn=<Profile level NF FQDN>**
nf-status	string	NF Profile Status	{apiRoot}/nrf-state-data/v1/nf-details**?nf-status=<Profile level NF Status>**

Query result parameters supported by nf-details URI

> **Note**
>
> - These attributes can be used to get specific attributes of NF Profile in the response. User can mention them as wish to see in the response.
> - These attributes can be mention in query using **result-attributes=**<Requested Attribute 1>,<Requested Attribute 2>
> - Different query result attributes can be mention together comma (,) separated in query to get specific results.
> - In case no result attribute is mentioned, then only **nf-instance-id** and **nf-fqdn** will be returned.
> - Some of the attributes are optional in NFProfile. In case specific result attribute is asked but it is not present in NFProfile then its value will be marked UNKNOWN.

Table 4-3 Query result attributes supported by nf-details URI

Name	Details	Query Example with API
fqdn	NF Profile FQDN required in Result attributes	{apiRoot}/nrf-state-data/v1/nf-details**?nf-instance-id=<NF Instance ID>&result-attributes=fqdn**
nfType	NF Type required in Result attributes	{apiRoot}/nrf-state-data/v1/nf-details**?nf-instance-id=<NF Instance ID>&result-attributes=nftype**

Table 4-3 (Cont.) Query result attributes supported by nf-details URI

Name	Details	Query Example with API
nfServices	NF Services (ServiceInstanceId and Service Name) required in Result attributes	{apiRoot}/nrf-state-data/v1/nf-details**?nf-instance-id=<NF Instance ID>&result-attributes=nfServices**
nfStatus	NF Profile Status required in Result attributes	{apiRoot}/nrf-state-data/v1/nf-details**?nf-fqdn=<NF FQDN>&result-attributes=nfStatus**

Response structure supported by nf-details URI

Table 4-4 NFProfileDetails

Name	Details	Response Example

dataTimeStamp	Timestamp when data was returned	<pre>{
		
		"dataTimeStamp":"2020-11-24T15:55:48.
		000Z",
		"nfProfileDataCount":3,
		"nfProfileData":
		[{"nfInstanceId":"13515195-
		c537-4645-9b97-96ec797fbbbe","fqdn":"
		ocamf1.oracle.com"},
		
		{"nfInstanceId":"23515195-
		c537-4645-9b97-96ec797fbbbe","fqdn":"
		ocpcf1.oracle.com"},
		
		{"nfInstanceId":"33515195-
		c537-4645-9b97-96ec797fbbbe","fqdn":"
		ocudr1.oracle.com"}
		]
		
		}</pre>

nfProfileDataCount	Count of NF profile data elements in response	same as above
nfProfileData	NF Profile data attributes requested in Query result attributes	same as above

Details of subscription-details URI

Query request parameters supported by subscription-details URI

> **Note**
>
> - These attributes can be used to get results based on specific input attributes. These attribute are optional to API. In case no query request and result attribute is mentioned, then only subscription ids will be provided for all of the Subscriptions.
> - In case complete subscription is needed, query request attributes shall contain specific attributes **subscription-id**. Otherwise only Subscription Ids will be provided by default and additionally query result attributes can be provided to get additional details.
> - At-most one query request attribute is supported, data is returned based on the request attribute.

Table 4-5 Query request attributes supported by subscription-details URI

Name	Data Type	Details	Query Example with API
subscription-id	string	Subscription Id for which data is required	{apiRoot}/nrf-state-data/v1/subscription-details**?subscription-id=<SUBSCRIPTION ID>**
nf-status-notification-uri	string	NF Status Notification URI for which data is required	{apiRoot}/nrf-state-data/v1/subscription-details?**nf-status-notification-uri=<NF Status Notification URI>**

Query result parameters supported by subscription-details URI

> **Note**
>
> - These attributes can be used to get specific attributes of Subscription in the response. User can mention them as wish to see in the response.
> - These attributes can be mention in query using **result-attributes=**<Requested Attribute 1>,<Requested Attribute 2>
> - Different query result attributes can be mention together comma (,) separated in query to get specific results.
> - In case no result attribute is mentioned, then only default attributes (subscriptionId) will be returned.
> - Some of the above attributes are Optional in Subscription data. In case specific result attribute is asked but it is not present in Subscription then its value will be marked UNKNOWN.

Table 4-6 Query result attributes supported by subscription-details URI

Name	Details	Query Example with API
reqNfFqdn	Requestor NF FQDN required in Result attributes	{apiRoot}/nrf-state-data/v1/subscription-details?**result-attributes=reqNfFqdn**
reqNfType	Requestor NF Type required in Result attributes	{apiRoot}/nrf-state-data/v1/subscription-details?**subscription-id=<SUBSCRIPTION ID>&result-attributes=reqNfType**

Table 4-6 (Cont.) Query result attributes supported by subscription-details URI

Name	Details	Query Example with API
nfStatusNotificationUri	NF Status Notification URI required in Result attributes	{apiRoot}/nrf-state-data/v1/subscription-details?**subscription-id=<SUBSCRIPTION ID>&result-attributes=nfStatusNotificationUri**
validityTime	Validity Time required in Result attributes	{apiRoot}/nrf-state-data/v1/subscription-details?**subscription-id=<SUBSCRIPTION ID>&result-attributes=validityTime**

Response structure supported by subscription-details URI

Table 4-7 SubscriptionDetails

Name	Details	Response Example

dataTimeStamp	Timestamp when data was returned	{
		"dataTimeStamp":
		"2020-11-24T15:55:48.000Z",
		"subscriptionDataCount": 3,
		"subscriptionData":
		[{"subscriptionId":"a1c5600116e9403bb
		032b214d564b729","reqNfFqdn":"amf1.or
		acle.com","reqNfType":"AMF"},
		
		{"subscriptionId":"b1c5600116e9403bb0
		32b214d564b729","reqNfFqdn":"pcf1.ora
		cle.com","reqNfType":"PCF"},
		
		{"subscriptionId":"c1c5600116e9403bb0
		32b214d564b729","reqNfFqdn":"amf2.ora
		cle.com","reqNfType":"UNKNOWN"}
		]
		}

subscriptionDataCount	Count of subscription data elements in response	same as above
subscriptionData	Subscription data attributes requested in query result attributes	same as above

4.1 Sample Queries

Following are the queries with examples.

Query#1 - Fetches all of the NF InstanceIds

Sample query: `{apiRoot}/nrf-state-data/v1/nf-details`

Sample response:

```
{
    "dataTimeStamp":"2020-11-24T15:55:48.000Z",
```

```
    "nfProfileDataCount":3,
    "nfProfileData":[{"nfInstanceId":"13515195-
c537-4645-9b97-96ec797fbbbe","fqdn":"ocamf1.oracle.com"},
                     {"nfInstanceId":"23515195-
c537-4645-9b97-96ec797fbbbe","fqdn":"ocpcf1.oracle.com"},
                     {"nfInstanceId":"33515195-
c537-4645-9b97-96ec797fbbbe","fqdn":"ocudr1.oracle.com"}
                    ]

}
```

Query#2 - Fetches all of the NF InstanceIds along with requested additional result attributes
Sample query: `{apiRoot}/nrf-state-data/v1/nf-details`

Sample response:

```
{
    "dataTimeStamp":"2020-11-24T15:55:48.000Z",
    "nfProfileDataCount":3,
    "nfProfileData":[{"nfInstanceId":"13515195-
c537-4645-9b97-96ec797fbbbe","fqdn":"ocpcf1.oracle.com","nfType":"PCF"},
                     {"nfInstanceId":"23515195-
c537-4645-9b97-96ec797fbbbe","fqdn":"ocpcf2.oracle.com","nfType":"PCF"},
                     {"nfInstanceId":"33515195-
c537-4645-9b97-96ec797fbbbe","fqdn":"ocudr1.oracle.com","nfType":"UDR"}
                    ]
}
```

Query#3 - Fetches complete NF Profile based on NF Instance ID
Sample query: `{apiRoot}/nrf-state-data/v1/nf-details?nf-instance-id=<NF Instance ID>`

Sample response:

```
{
    "dataTimeStamp":"2020-11-24T15:55:48.000Z",
    "nfProfileDataCount":1,
    "nfProfileData":[{<< !!!!Complete NF Profile!!!! >>}]
}
```

Query#4 - Fetches NF Profile FQDN attribute value based on NF Instance ID
Sample query: `{apiRoot}/nrf-state-data/v1/nf-details?nf-instance-id=<NF Instance ID>&result-attributes=fqdn`

Sample response:

```
{
    "dataTimeStamp":"2020-11-24T15:55:48.000Z",
    "nfProfileDataCount":1,
    "nfProfileData":[{"nfInstanceId":"33515195-
c537-4645-9b97-96ec797fbbbe","fqdn":"ocpcf1.oracle.com"}]
}
```

Query#5 - Fetches NF Services attribute of NF Profile based on NF Instance ID
Sample query: {apiRoot}/nrf-state-data/v1/nf-details?nf-instance-id=<NF Instance ID>&result-attributes=nfServices

Sample response:

```
{
    "dataTimeStamp":"2020-11-24T15:55:48.000Z",
    "nfProfileDataCount":1,
    "nfProfileData":[{"nfInstanceId":"33515195-
c537-4645-9b97-96ec797fbbbe","nfServices":
[{"<serviceInstanceId":"aaaa","serviceName":"ABC"},

            {"<serviceInstanceId":"bbbb","serviceName":"XYZ"}]
                    }]
 }
```

Query#6 - Fetches complete NF Profile based on NF FQDN
Sample query: {apiRoot}/nrf-state-data/v1/nf-details?nf-fqdn=<NF FQDN>

Sample response:

```
{
    "dataTimeStamp":"2020-11-24T15:55:48.000Z",
    "nfProfileDataCount":1,
    "nfProfileData":[{<< !!!!Complete NF Profile!!!! >>}]
}
```

Query#7 - Fetches NF Status attribute value of NF Profile based on NF Instance ID
Sample query: {apiRoot}/nrf-state-data/v1/nf-details?nf-instance-id=<NF Instance ID>&result-attributes=nfStatus

Sample response:

```
{
    "dataTimeStamp":"2020-11-24T15:55:48.000Z",
    "nfProfileDataCount":1,
    "nfProfileData":[{"nfInstanceId":"33515195-
c537-4645-9b97-96ec797fbbbe","nfStatus":"Suspended"}]
}
```

Query#8 - Fetches NFInstance Id and its NF Status attribute value based on NF FQDN
Sample query: {apiRoot}/nrf-state-data/v1/nf-details?nf-fqdn=<NF FQDN>&result-attributes=nfStatus

Sample response:

```
{
    "dataTimeStamp":"2020-11-24T15:55:48.000Z",
    "count":3,
    "nfProfileData":[{"nfInstanceId":"13515195-
c537-4645-9b97-96ec797fbbbe","nfStatus":"Suspended"}]
}
```

Query#9 - Fetches NF Profile details based on NF Status
Sample query: `{apiRoot}/nrf-state-data/v1/nf-details?nf-status=<NF Status>&result-attributes=fqdn,nfType`

Sample response:

```
{
    "dataTimeStamp": "2020-11-24T15:55:48.000Z",
    "nfProfileDataCount": 3,
    "nfProfileData": [{"nfInstanceId":"13515195-c537-4645-9b97-96ec797fbbbe",
"fqdn":"ocpcf1.oracle.com", "nfType":"PCF"},
                      {"nfInstanceId":"23515195-c537-4645-9b97-96ec797fbbbe",
"fqdn":"ocpcf2.oracle.com", "nfType":"PCF"},
                      {"nfInstanceId":"33515195-c537-4645-9b97-96ec797fbbbe",
"fqdn":"ocudr1.oracle.com", "nfType":"UDR"}
                     ]
}
```

Query#10 - Fetches all of the subscriptions
Sample query: `{apiRoot}/nrf-state-data/v1/subscription-details`

Sample response:

```
{
    "dataTimeStamp": "2020-11-24T15:55:48.000Z",
    "subscriptionDataCount": 3,
    "subscriptionData": [{"subscriptionId":
"a1c5600116e9403bb032b214d564b729"},
                         {"subscriptionId":
"b1c5600116e9403bb032b214d564b729"},
                         {"subscriptionId":
"c1c5600116e9403bb032b214d564b729"}
                        ]
}
```

Query#11- Fetches all of the subscriptions along with requested result attributes
Sample query: `{apiRoot}/nrf-state-data/v1/subscription-details?result-attributes=reqNfFqdn,reqNfType`

Sample response:

```
{
    "dataTimeStamp": "2020-11-24T15:55:48.000Z",
    "subscriptionDataCount": 3,
    "subscriptionData":
[{"subscriptionId":"a1c5600116e9403bb032b214d564b729","reqNfFqdn":"amf1.oracle
.com","reqNfType":"AMF"},

{"subscriptionId":"b1c5600116e9403bb032b214d564b729","reqNfFqdn":"pcf1.oracle.
com","reqNfType":"PCF"},

{"subscriptionId":"c1c5600116e9403bb032b214d564b729","reqNfFqdn":"amf2.oracle.
com","reqNfType":"UNKNOWN"}
                         ]
}
```

Query#12 - Fetches Subscription Data based on Subscription ID
Sample query: {apiRoot}/nrf-state-data/v1/subscription-details?subscription-id=<SUBSCRIPTION ID>

Sample response:

```
{
  "dataTimeStamp":"2020-11-24T15:55:48.000Z",
  "subscriptionDataCount":3,
  "subscriptionData":[<< !!!!Complete Subscription Data!!!!>>]
}
```

Query#13 - Fetches specific attributes of Subscription Data based on Subscription ID
Sample query: {apiRoot}/nrf-state-data/v1/subscription-details?subscription-id=<SUBSCRIPTION ID>&result-attributes=reqNfFqdn,reqNfType

Sample response:

```
{
    "dataTimeStamp": "2020-11-24T15:55:48.000Z",
    "subscriptionDataCount": 3,
    "subscriptionData":
[{"subscriptionId":"a1c5600116e9403bb032b214d564b729","reqNfFqdn":"amf1.oracle
.com","reqNfType":"AMF"},

{"subscriptionId":"b1c5600116e9403bb032b214d564b729","reqNfFqdn":"pcf1.oracle.
com","reqNfType":"PCF"},

{"subscriptionId":"c1c5600116e9403bb032b214d564b729","reqNfFqdn":"amf2.oracle.
com","reqNfType":"UNKNOWN"}
                        ]
}
```

Query#14 - Fetches Subscription IDs based on nf-status-notification-uri
Sample query: {apiRoot}/nrf-state-data/v1/subscription-details?nf-status-notification-uri=<NF Status Notification URI>&result-attributes=reqNfFqdn,reqNfType

Sample response:

```
{
    "dataTimeStamp": "2020-11-24T15:55:48.000Z",
    "subscriptionDataCount": 2,
    "subscriptionData":
[{"subscriptionId":"a1c5600116e9403bb032b214d564b729","reqNfFqdn":"amf1.oracle
.com","reqNfType":"AMF"},

{"subscriptionId":"b1c5600116e9403bb032b214d564b729","reqNfFqdn":"UNKNOWN","re
qNfType":"PCF"}
                        ]
}
```

5
Common Services REST APIs

This chapter provides information about the REST specifications for Common Services used in NRF. You can use the following APIs to update configurations related to the Overload Control feature.

5.1 Perf-Info Configuration

This section explains REST API configurations required at Perf-Info to enable Overload control feature.

5.1.1 Overload Level Threshold Configuration in Perf-Info

The following URI can be used for configuring overload threshold in Ingress Gateway.

URI: *{apiRoot}/nrf/nf-common-component/v1/perfinfo/overloadLevelThreshold*

Method:

- GET: Get Overload Threshold Value of the required service (Backend service).
- PUT: Update the Overload Threshold Value of the required service (Backend service).
- DELETE: Delete the Overload Threshold Value of the required service (Backend service).

Content Type: application/json

Body:

```
[
  {
    "svcName": "ocnrf-nfdiscovery",
    "metricsThresholdList": [
      {
        "metricsName": "svc_failure_count",
        "levelThresholdList": [
          {
            "level": "L1",
            "onsetValue": 250,
            "abatementValue": 200
          },
          {
            "level": "L2",
            "onsetValue": 300,
            "abatementValue": 280
          },
          {
            "level": "L3",
            "onsetValue": 600,
            "abatementValue": 480
          },
          {
```

```
        "level": "L4",
        "onsetValue": 900,
        "abatementValue": 720
      }
    ]
  },
  {
    "metricsName": "memory",
    "levelThresholdList": [
      {
        "level": "L1",
        "onsetValue": 70,
        "abatementValue": 65
      },
      {
        "level": "L2",
        "onsetValue": 80,
        "abatementValue": 75
      },
      {
        "level": "L3",
        "onsetValue": 90,
        "abatementValue": 85
      },
      {
        "level": "L4",
        "onsetValue": 95,
        "abatementValue": 91
      }
    ]
  },
  {
    "metricsName": "cpu",
    "levelThresholdList": [
      {
        "level": "L1",
        "onsetValue": 65,
        "abatementValue": 60
      },
      {
        "level": "L2",
        "onsetValue": 75,
        "abatementValue": 70
      },
      {
        "level": "L3",
        "onsetValue": 80,
        "abatementValue": 75
      },
      {
        "level": "L4",
        "onsetValue": 90,
        "abatementValue": 85
      }
    ]
  },
```

```
      {
        "metricsName": "svc_pending_count",
        "levelThresholdList": [
          {
            "level": "L1",
            "onsetValue": 5000,
            "abatementValue": 4500
          },
          {
            "level": "L2",
            "onsetValue": 6000,
            "abatementValue": 5300
          },
          {
            "level": "L3",
            "onsetValue": 7000,
            "abatementValue": 6200
          },
          {
            "level": "L4",
            "onsetValue": 8000,
            "abatementValue": 7200
          }
        ]
      }
    ]
  },
  {
    "svcName": "ocnrf-nfregistration",
    "metricsThresholdList": [
      {
        "metricsName": "svc_failure_count",
        "levelThresholdList": [
          {
            "level": "L1",
            "onsetValue": 250,
            "abatementValue": 200
          },
          {
            "level": "L2",
            "onsetValue": 300,
            "abatementValue": 280
          },
          {
            "level": "L3",
            "onsetValue": 600,
            "abatementValue": 480
          },
          {
            "level": "L4",
            "onsetValue": 900,
            "abatementValue": 720
          }
        ]
      },
      {
```

```
    "metricsName": "memory",
    "levelThresholdList": [
      {
        "level": "L1",
        "onsetValue": 70,
        "abatementValue": 65
      },
      {
        "level": "L2",
        "onsetValue": 80,
        "abatementValue": 75
      },
      {
        "level": "L3",
        "onsetValue": 90,
        "abatementValue": 85
      },
      {
        "level": "L4",
        "onsetValue": 95,
        "abatementValue": 91
      }
    ]
  },
  {
    "metricsName": "cpu",
    "levelThresholdList": [
      {
        "level": "L1",
        "onsetValue": 65,
        "abatementValue": 60
      },
      {
        "level": "L2",
        "onsetValue": 75,
        "abatementValue": 70
      },
      {
        "level": "L3",
        "onsetValue": 80,
        "abatementValue": 75
      },
      {
        "level": "L4",
        "onsetValue": 90,
        "abatementValue": 85
      }
    ]
  },
  {
    "metricsName": "svc_pending_count",
    "levelThresholdList": [
      {
        "level": "L1",
        "onsetValue": 5000,
        "abatementValue": 4500
```

```
            },
            {
              "level": "L2",
              "onsetValue": 6000,
              "abatementValue": 5300
            },
            {
              "level": "L3",
              "onsetValue": 7000,
              "abatementValue": 6200
            },
            {
              "level": "L4",
              "onsetValue": 8000,
              "abatementValue": 7200
            }
          ]
        }
      ]
    },
    {
      "svcName": "ocnrf-nfaccesstoken",
      "metricsThresholdList": [
        {
          "metricsName": "svc_failure_count",
          "levelThresholdList": [
            {
              "level": "L1",
              "onsetValue": 250,
              "abatementValue": 200
            },
            {
              "level": "L2",
              "onsetValue": 300,
              "abatementValue": 280
            },
            {
              "level": "L3",
              "onsetValue": 600,
              "abatementValue": 480
            },
            {
              "level": "L4",
              "onsetValue": 900,
              "abatementValue": 720
            }
          ]
        },
        {
          "metricsName": "memory",
          "levelThresholdList": [
            {
              "level": "L1",
              "onsetValue": 70,
              "abatementValue": 65
            },
```

```
        {
          "level": "L2",
          "onsetValue": 80,
          "abatementValue": 75
        },
        {
          "level": "L3",
          "onsetValue": 90,
          "abatementValue": 85
        },
        {
          "level": "L4",
          "onsetValue": 95,
          "abatementValue": 91
        }
      ]
    },
    {
      "metricsName": "cpu",
      "levelThresholdList": [
        {
          "level": "L1",
          "onsetValue": 65,
          "abatementValue": 60
        },
        {
          "level": "L2",
          "onsetValue": 75,
          "abatementValue": 70
        },
        {
          "level": "L3",
          "onsetValue": 80,
          "abatementValue": 75
        },
        {
          "level": "L4",
          "onsetValue": 92,
          "abatementValue": 87
        }
      ]
    },
    {
      "metricsName": "svc_pending_count",
      "levelThresholdList": [
        {
          "level": "L1",
          "onsetValue": 100,
          "abatementValue": 58
        },
        {
          "level": "L2",
          "onsetValue": 190,
          "abatementValue": 125
        },
        {
```

```
            "level": "L3",
            "onsetValue": 250,
            "abatementValue": 200
          },
          {
            "level": "L4",
            "onsetValue": 400,
            "abatementValue": 300
          }
        ]
      }
    ]
  }
]
```

Table 5-1 Overload Level Threshold

Attribute Name	Description	Details
`svcName`	Name of the backend service (svcName).	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
`metricsThresholdList`	List of criteria used to calculate the load level.	**DataType**: array **Mandatory(M)/Optional(O)/Conditional(C)**: M
`metricsThresholdList.metricsName`	Name of overload indicator such as cpu, svc_failure_count, svc_pending_count, and memory.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
`metricsThresholdList.levelThresholdList`	List of threshold values.	**DataType**: array **Mandatory(M)/Optional(O)/Conditional(C)**: M
`metricsThresholdList.levelThresholdList.level`	Specifies the name of the level. The name specified in this parameter must match the level name in Ingress Gateway's `ocdiscardpolicies`.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
`metricsThresholdList.levelThresholdList.abatementValue`	The overload condition is considered as cleared, if the load level for the indicator mentioned in `metricsThresholdList.metricsName` is below the abatement value.	**DataType**: integer **Mandatory(M)/Optional(O)/Conditional(C)**: M
`metricsThresholdList.levelThresholdList.onsetValue`	The load level for the indicator mentioned in `metricsThresholdList.metricsName` is set as per the `metricsThresholdList.levelThresholdList.level`, if the overload condition is breached this onset value.	**DataType**: integer **Mandatory(M)/Optional(O)/Conditional(C)**: M

5.2 Egress Gateway Configuration

This section explains REST API configurations required at Egress Gateway for various features.

5.2.1 Peer Configuration

This resource is used in `SbiRouting` feature. This URI can be used to add or update the list of peers wherein each peer consists of ID, host, port or virtualHost, and apiPrefix. The ID of each

peer is mapped to `peerIdentifier` in "`peersetconfiguration`" resource. The default value is null.

Peer Configuration

URI: *{apiRoot}/nrf/nf-common-component/v1/egw/peerconfiguration*

Method: PUT, PATCH, GET

- **PUT**: Updates peer configuration.
- **GET**: Retrieves peer configuration.
- **PATCH**: Updates specific peer configuration.

Resource: array (PeerConfiguration)

Body:

For static host configuration:

```
curl -v -X PUT "http://10.75.226.126:30747/nrf/nf-common-component/v1/egw/
peerconfiguration" -H  "Content-Type: application/json"  -d @peer.json

peer.json sample:-
[
  {
    "id": "peer1",
    "host": "scp-stub-service01",
    "port": "8080",
    "apiPrefix": "/",
    "healthApiPath":"/{scpApiRoot}/{apiVersion}/status"
  },
  {
    "id": "peer2",
    "host": "scp-stub-service02",
    "port": "8080",
    "apiPrefix": "/",
    "healthApiPath":"/{scpApiRoot}/{apiVersion}/status"
  }
]
```

For virtual host configuration:

```
curl -v -X PUT "http://10.75.226.126:30747/nrf/nf-common-component/v1/egw/
peerconfiguration" -H  "Content-Type: application/json"  -d @peer.json

peer.json sample:-
[
  {
    "id": "peer3",
    "virtualHost": "scp-stub-service03",
    "apiPrefix": "/",
    "healthApiPath":"/{scpApiRoot}/{apiVersion}/status"
  }
]
```

> **Note**
>
> Combination of static and virtual host configuration is not supported.

Table 5-2 Peer Configuration

Attribute Name	Description	Details
`id`	Peer identifier	**DataType**: string **Constraints**: Unique value in peer configuration **Default Value**: NA
`host`	Host details of a local peer. It can be IPv4, IPv6 and FQDN details.	**DataType**: string **Constraints**: NA **Default Value**: NA
`port`	Port details of a local host peer.	**DataType**: string **Constraints**: NA **Default Value**: NA
`virtualHost`	Host details of a remote peer. This FQDN is sent to Alternate Route Service for DNS SRV resolution.	**DataType**: string **Constraints**: NA **Default Value**: NA
`apiPrefix`	API prefix details of a peer.	**DataType**: string **Constraints**: Keep the value as **/** only for NRF **Default Value**: NA
`healthApiPath`	Include the SCP API details.	**DataType**: string **Constraints**: NA **Default Value**: /{scpApiRoot}/{apiVersion}/status

5.2.2 Peer Set Configuration

This section provides details about peer set configuration at Egress Gateway. This URI is used to add or update the list of peer sets wherein each peer set consists of `id` and list of http/https instances. Each instance consists of priority and peer identifier that is mapped to id in peerconfiguration resource. The `id` of each peer set is mapped to `peerSetIdentifier` in `routesconfiguration` resource. The default value is null.

URI: *{apiRoot}/nrf/nf-common-component/v1/egw/peersetconfiguration*

Method: PUT, PATCH, GET

- **PUT**: Updates peer set configuration.
- **GET**: Retrieves peer set configuration.
- **PATCH**: Updates specific peer set configuration.

Resource: array (PeerSetConfiguration)

Body:

```
curl -v -X PUT "http://10.75.226.126:32247/nrf/nf-common-component/v1/egw/
peersetconfiguration" -H  "Content-Type: application/json"  -d @peerset.json

sample peerset.json
[
```

```
    {
        "id":"set0",
        "httpConfiguration":[
        {
        "priority": 1,
        "peerIdentifier": "peer1"
        }],
        "httpsConfiguration":[
        {
        "priority": 1,
        "peerIdentifier": "peer1"
        }]
    }
]
```

Table 5-3 Peer Set Configuration

Attribute	Description	Details
id	Identifier for Peer Set.	**DataType**: string **Constraints**: Unique value in peer set configuration. **Default Value**: NA
httpConfiguration	Configuration for HTTP based Peers. This value will be selected, if 3GPPAPIRootScheme value is http.	**DataType**: array (Table 5-4) **Constraints**: NA **Default Value**: NA
httpsConfiguration	Configuration for HTTPs based Peers. This value will be selected, if 3GPPAPIRootScheme value is https.	**DataType**: array (Table 5-4) **Constraints**: NA **Default Value**: NA

Table 5-4 Peer Identifier Configuration

Attribute	Description	Details
priority	Priority of peer to be used in a peer set.	**DataType**: integer **Constraints**: Priority must be unique. **Default Value**: NA
peerIdentifier	Peer identifier is the value of peer configured during PeerConfiguration.	**DataType**: string **Constraints**: NA **Default Value**: NA

5.2.3 Peer Monitoring Configuration

The below API is used to retrieve the peer monitoring configuration.

URI: *{apiRoot}/nrf/nf-common-component/v1/egw/peermonitoringconfiguration*

Method:

- **GET**: Fetches the details of peer monitoring configuration.
- **PUT**: Updates replace the existing values with the new value sent for peer monitoring configuration.
- **PATCH**: Changes the value of the modified parameters in peer monitoring configuration.

Content Type: application/json

Body:

```
{
      "enabled":true,
      "timeout":1000,
      "frequency":2000,
      "failureThreshold":3,
      "successThreshold":4
   }
```

Table 5-5 Peer Monitoring Configuration

Attribute	Description	Details
enabled	Indicates the attribute to enable or disable monitoring at a global level.	**DataType**: Boolean **Constraints**:true or false **Default Value**: false
timeout	Indicates the flag to configure the duration of time after which calls to the SCP health API is timed out.	**DataType**: long **Constraints**:NA **Recommended Range:** 300 milliseconds to 10000 milliseconds **Default Value**: 1000 milliseconds
frequency	Indicates the frequency or interval at which Egress Gateway microservice initiates health check calls toward SCP.	**DataType**: long **Constraints**:NA **Recommended Range:** 300 milliseconds to 10000 milliseconds **Default Value**: 2000 milliseconds
failureThreshold	Indicates the number of consecutive failure responses after which a healthy SCP can be marked as unhealthy. Health API call to given SCP fails consecutively to these many attempts before it is marked as Unavailable from Available.	**Data Type**: Integer **Constraints**: NA **Recommended Range**: 1 to 5 **Default Value**: 3
successThreshold	Indicates the number of successful responses after which an unhealthy SCP can be marked as healthy. Health API call to given SCP shall succeed consecutively to these many attempts before it is marked as Available from Unavailable.	**Data Type**: Integer **Constraints**: NA **Recommended Range**: 1 to 5 **Default Value**: 4

5.2.4 Error Criteria Sets

This section provides details about error criteria set configuration at Egress Gateway.

URI: *{apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerrorcriteriasets*

Method: PUT, PATCH, GET

- **PUT**: Updates sbiroutingerrorcriteria configuration.
- **GET**: Retrieves sbiroutingerrorcriteria configuration.
- **PATCH**: Updates specific sbiroutingerrorcriteria configuration.

Body:

```
[
   {
      "id":"criteria_0",
      "method":[
         "GET",
         "POST",
         "PUT",
         "DELETE",
         "PATCH"
      ],
      "exceptions":[
         "java.util.concurrent.TimeoutException",
         "java.net.UnknownHostException"
      ]
   },
   {
      "id":"criteria_1",
      "method":[
         "GET",
         "POST",
         "PUT",
         "DELETE",
         "PATCH"
      ],
      "response":{
         "statuses":[
            {
               "statusSeries":"4xx",
               "status":[
                  400,
                  404
               ]
            },
            {
               "statusSeries":"5xx",
               "status":[
                  500,
                  503
               ]
            }
         ]}
   }
]
```

Table 5-6 sbiroutingerrorcriteriasets

Attribute	Description	Details
sbiroutingerrorcriteriasets.id	Unique id for a sbiRoutingErrorCriteriaSet	**DataType**: string **Constraints**: Unique value of route **Default Value**: NA
sbiroutingerrorcriteriasets.method	Methods for which reroute or retry is triggered.	**DataType**: string **Constraints**: GET, POST, PUT, PATCH, DELETE **Default Value**: NA
sbiroutingerrorcriteriasets.response.exceptions	Specific exceptions for which reroute or retry is triggered. **Note**: exceptions and response cannot be configured under same criteria Id.	**DataType**: string **Constraints**: NA **Default Value**: java.util.concurrent.TimeoutException
sbiroutingerrorcriteriasets.response.statuses.statusSeries	Http Status Series for which reroute or retry is triggered, when the error response is received from downstream.	**DataType**: string **Constraints**: 4xx, 5xx **Default Value**: NA
sbiroutingerrorcriteriasets.response.statuses.status	Specific HTTP Statuses that belongs to above mentioned status series for which reroute or retry is triggered. To enable retry or reroute for all the HTTP status belonging to a status series, configure this as **-1**.	**DataType**: string **Constraints**: 401, 404 or -1 **Default Value**: NA
sbiroutingerrorcriteriasets.response.exceptions	Specific exceptions for which reroute or retry is triggered. **Note**: exceptions and response cannot be configured under same criteria Id.	**DataType**: string **Constraints**: NA **Default Value**: java.util.concurrent.TimeoutException

5.2.5 Error Action Sets

This section provides details about error action set configuration at Egress Gateway.

URI: *{apiRoot}/nrf/nf-common-component/v1/egw/sbiroutingerroractionsets*

Method: PUT, PATCH, GET

- **PUT**: Updates sbiroutingerroraction configuration.
- **GET**: Retrieves sbiroutingerroraction configuration.
- **PATCH**: Updates specific sbiroutingerroraction configuration.

Body:

```
[
   {
      "id":"action_0",
      "action":"reroute",
      "attempts":2,
      "blacklist":{
         "enabled":false,
         "duration":60000
      }
   },
```

```
    {
       "id":"action_1",
       "action":"reroute",
       "attempts":2,
       "blacklist":{
          "enabled":false,
          "duration":60000
       }
    }
]
```

Table 5-7 sbiroutingerroractionsets

Attribute	Description	Details
sbiroutingerroractionsets.id	Unique Id for sbiRoutingErrorActionSet	**DataType**: string **Constraints**: **Default Value**: NA
sbiroutingerroractionsets.action	Action that needs to be taken when specific criteria set is matched.	**DataType**: string **Constraints**: reroute, retry **Default Value**: reroute
sbiroutingerroractionsets.attempts	Maximum no of retries to either same or different peer in case of error or failures from backend.	**DataType**: string **Constraints**: NA **Default Value**: 3
sbiroutingerroractionsets.blackList.enabled	This flag enables the peer blacklist feature using the server headers received in the response.	**DataType**: boolean **Constraints**: true, false **Default Value**: false
sbiroutingerroractionsets.blackList.duration	The duration for which the peer is blacklisted and no traffic is routed to that peer for this period.	**DataType**: integer **Constraints**: NA **Default Value**: 60000

5.2.6 Routes Configuration

This URI can be used to fetch and update the list of routes configuration.

Routes Configuration

URI: *{apiRoot}/nrf/nf-common-component/v1/egw/routesconfiguration*

Method: PUT, PATCH, GET

- PUT: Updates route configuration.
- GET: Retrieves route configuration.
- PATCH: Updates specific route configuration.

Resource: array (RoutesConfiguration)

Body

```
{
      "id":"egress_scp_proxy2",
      "uri":"http://localhost:32069/",
      "order":3,
```

```
        "metadata":{
           "httpsTargetOnly":false,
           "httpRuriOnly":false,
           "sbiRoutingEnabled":false
        },
        "predicates":[
           {
              "args":{
                 "pattern":"/nef"
              },
              "name":"Path"
           }
        ],
        "filters":[
           {
              "name":"SbiRouting",
              "args":{
                 "peerSetIdentifier":"set0",
                 "customPeerSelectorEnabled":false
              }
           }
        ]
     }
```

Table 5-8 Routes Configuration

Attribute	Description	Details
`id`	Route configuration identifier **CAUTION**:- Default Route with id '`default_route`' is configured automatically. Include this route in the message body while adding new routes using the PUT operation. Otherwise, it will impact the traffic.	**DataType**: string **Constraints**: Unique value of route **Default Value**: NA
`uri`	Provide any dummy URL, or leave the existing URL with the existing value.	**DataType**: string **Constraints**: NA **Default Value**: NA
`order`	Provide the order of the execution of this route. **Note:** The value of the order attribute must be unique for each routing configuration.	**DataType**: integer **Constraints**: NA **Default Value**: NA
`httpRuriOnly`	This flag indicates the scheme of the outgoing request from OCNRF. If the value is set to true, the scheme of RURI is changed to http. If the value is set to false, no change occurs to the scheme. **Note**: In case of non-ASM configuration and `3GPPAPIRootScheme` in Roaming Options is set to https, set the value as false.	**DataType**: boolean (true,false) **Constraints**: NA **Default Value**: NA
`httpsTargetOnly`	For NRF, the value of this flag must always be set to true. **Note**: This is a read-only attribute.	**DataType**: boolean (true, false) **Constraints**: NA **Default Value**: NA

Table 5-8 (Cont.) Routes Configuration

Attribute	Description	Details
predicates	Header predicate details for matching target PLMN mapped to this SBIRoute rule. **Note**: The predicates can be combined in a single configuration as shown in the Body, or only the required configuration can be retained for processing the message. Sample value:- "predicates": [{ "args": { "header": "OC-MCCMNC", "regexp": "310014" }, "name": "Header" }] **Note**: "header": "OC-MCCMNC" must not be changed. Only "regexp": "310014" can be modified. regexp consists of MCC and MNC values. In this example, value of MCC and MNC is 310 and 014. Multiple values can be provided for regexp as shown below: "predicates": [{ "args": { "header": "OC-MCCMNC", "regexp": "310014" }, "name": "Header" }, { "args": { "header": "OC-MCCMNC", "regexp": "315012" }, "name": "Header" }]	**DataType**: Predicate structure. See description for more details. **Constraints**: NA **Default Value**: NA
filters	Filters can be created for various purposes. Use all of the filters as mentioned in the example without any updates.	**DataType**: Filter structure. See the description for more details. **Constraints**: NA **Default Value**: NA

5.3 Ingress Gateway Configuration

This section explains REST API configurations required at Ingress Gateway for various features.

5.3.1 Policy Mapping Configuration in Ingress Gateway

The following URI can be used to update service names and corresponding policy names for the service that is mapped to "ocDiscardPolicies" based on "policyName" and enable or disable the Overload Control feature and the sampling period in overload control. The Overload Control feature is disabled by default, and the sampling period is 200. To enable the feature, invoke REST API and update the enabled flag to true.

URI: {apiRoot}/nrf/nf-common-component/v1/igw/ocpolicymapping

Method:

- GET: Get Policy mapping value of the required service.
- PUT: Update the Policy mapping value of the required service.

Content Type: application/json

Body;

```
{
    "enabled": true,
    "mappings": [
        {
            "svcName": "ocnrf-nfdiscovery",
            "policyName": "nfdiscoveryPolicy"
        },
        {
            "svcName": "ocnrf-nfaccesstoken",
            "policyName": "nfaccesstokenPolicy"
        },
        {
            "svcName": "ocnrf-nfregistration",
            "policyName": "nfregistrationPolicy"
        }
    ],
    "samplingPeriod": 200
}
```

Table 5-9 Policy Mapping Configuration

Attribute Name	Description	Details
`enabled`	To enable or disable discard policy at Ingress Gateway.	**DataType**: boolean **Mandatory(M)/Optional(O)/Conditional(C)**: M
`mappings.svcName`	The service entry to determine a mapping between service and discard policy name per service. svcName must be added in the following format:<deployment-name>-<servicename> **Note**: servicename is fixed and cannot be changed.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
`mappings.policyName`	The discard policy entry to determine a mapping between service and discard policy name per service.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
`samplingPeriod`	Time frame for each cycle of Overload Control per service. Its value is in milliseconds.	**DataType**: integer **Mandatory(M)/Optional(O)/Conditional(C)**: M

5.3.2 Discard Policy Configuration in Ingress Gateway

The following URI can be used to update discard policies that are used in overload control to select the appropriate policy from the configured list based on the load level of a particular service. By default, `ocdiscardpolicies` is null.

URI: {apiRoot}/nrf/nf-common-component/v1/igw/ocdiscardpolicies

Method:

- GET: Get discard policy configuration of the required service.
- PUT: Update discard policy configuration of the required service.

Content Type: application/json

Body:

```
[
    {
        "name": "nfdiscoveryPolicy",
        "scheme": "PercentageBased",
        "policies": [
            {
                "level": "L1",
                "value": 0,
                "action": "RejectWithErrorCode",
                "errorCodeProfile": "error429"
            },
            {
                "level": "L2",
                "value": 10,
                "action": "RejectWithErrorCode",
                "errorCodeProfile": "error429"
            },
            {
                "level": "L3",
                "value": 25,
                "action": "RejectWithErrorCode",
                "errorCodeProfile": "error429"
            },
            {
                "level": "L4",
                "value": 50,
                "action": "RejectWithErrorCode",
                "errorCodeProfile": "error503"
            }
        ]
    },
    {
        "name": "nfaccesstokenPolicy",
        "scheme": "PercentageBased",
        "policies": [
            {
                "level": "L1",
                "value": 0,
                "action": "RejectWithErrorCode",
                "errorCodeProfile": "error429"
            },
            {
                "level": "L2",
                "value": 10,
                "action": "RejectWithErrorCode",
                "errorCodeProfile": "error429"
            },
            {
                "level": "L3",
                "value": 25,
                "action": "RejectWithErrorCode",
                "errorCodeProfile": "error429"
            },
            {
```

```
                    "level": "L4",
                    "value": 50,
                    "action": "RejectWithErrorCode",
                    "errorCodeProfile": "error503"
                }
            ]
        },
        {
            "name": "nfregistrationPolicy",
            "scheme": "PercentageBased",
            "policies": [
                {
                    "level": "L1",
                    "value": 0,
                    "action": "RejectWithErrorCode",
                    "errorCodeProfile": "error429"
                },
                {
                    "level": "L2",
                    "value": 10,
                    "action": "RejectWithErrorCode",
                    "errorCodeProfile": "error429"
                },
                {
                    "level": "L3",
                    "value": 15,
                    "action": "RejectWithErrorCode",
                    "errorCodeProfile": "error429"
                },
                {
                    "level": "L4",
                    "value": 25,
                    "action": "RejectWithErrorCode",
                    "errorCodeProfile": "error503"
                }
            ]
        }
    ]
```

Table 5-10 Discard Policy Configuration

Attribute Name	Description	Details
name	Name of the discarded policy. **Note**: name must be the value configured in policyName under ocpolicymapping.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
scheme	Discarded policy scheme based on percentage.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
policies.value	Value of priority above which requests are considered as potential candidates for drop. Percentage of requests to drop in the current sampling period over the calculated rate in the previous sampling period.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M

Table 5-10 (Cont.) Discard Policy Configuration

Attribute Name	Description	Details
`policies.action`	Defines the action to be taken on selected requests rejection based on error code.	**DataType**: string The value can be: RejectWithErrorCode **Mandatory(M)/Optional(O)/Conditional(C)**: M
`policies.level`	Defines the overload level.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
`policies.errorCode Profile`	The error code profiles.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M

5.3.3 Error Code Profile Configuration in Ingress Gateway

The following URI can be used to update the errorcodeprofiles that is used in Overload Control, Controlled Shutdown feature for populating details in error responses when a request is discarded.

> **Note**
>
> Below is the sample error code profile that can be linked for controlled shutdown. The errorcodeprofiles configuration is used for other features as well. Hence, the configuration must be performed with caution.

By default, the errorcodeprofiles remains null.

URI: {apiRoot}/nrf/nf-common-component/v1/igw/errorcodeprofiles

Method:

- GET: Get Error Code configuration of the required service.
- PUT: Update Error Code configuration of the required service.

Content Type: application/json

Body:

```
[
    {
        "name": "error429",
        "errorCode": 429,
        "errorCause": "Too many requests",
        "errorTitle": "Too many requests",
        "redirectURL": "",
        "retry-after": "",
        "errorDescription": "Too many requests"
    },
    {
        "name": "error503",
        "errorCode": 503,
        "errorCause": "Backend not able to handle traffic",
        "errorTitle": "Backend not able to handle traffic",
        "redirectURL": "",
```

```
            "retry-after": "",
            "errorDescription": "Backend not able to handle traffic"
        },
        {
            "name": "shutdownerror503",
            "errorCode": 503,
            "errorCause": "UNSPECIFIED_NF_FAILURE",
            "errorTitle": "NRF is in COMPLETE_SHUTDOWN state. Service temporarily
unavailable",
            "redirectURL": "",
            "retry-after": "",
            "errorDescription": "NRF is in COMPLETE_SHUTDOWN state. Service
temporarily unavailable"
        }

    ]
```

Table 5-11 Error Code Profile Configuration

Attribute Name	Description	Details
name	Error name.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
errorCode	errorCode field in an errorScenario determines the HttpStatusCode that needs to be populated in ProblemDetails (HttpStatus field) response from Ingress Gateway when the exception occurred at Ingress Gateway matches the configured errorScenario's exceptionType field.	**DataType**: integer **Mandatory(M)/Optional(O)/Conditional(C)**: M
errorCause	errorCause field in an errorScenario determines the error cause that needs to be populated in ProblemDetails (Cause field) response from Ingress Gateway when the exception occurred at Ingress Gateway matches the configured errorScenario's exceptionType parameter.	**DataType**: integer **Mandatory(M)/Optional(O)/Conditional(C)**: O
errorTitle	errorTitle field in an errorScenario determines the title that needs to be populated in ProblemDetails (Title field) response from Ingress Gateway when the exception occurred at Ingress Gateway matches the configured errorScenario's exceptionType parameter.	**DataType**: integer **Mandatory(M)/Optional(O)/Conditional(C)**: O
redirectURL	redirectURL field in an errorScenario determines the redirection URL, this value is populated in LOCATION header while sending response from Ingress Gateway. The header is populated only when the exception occurred at Ingress Gateway matches the configured errorScenario's exceptionType parameter, the errorCode configured for the particular errorScenario lies in 3xx error series and the redirectUrl field for the particular errorScenario is configured appropriately.	**DataType**: integer **Mandatory(M)/Optional(O)/Conditional(C)**: O

Table 5-11 (Cont.) Error Code Profile Configuration

Attribute Name	Description	Details
retry-after	retry-after field in an errorScenario determines the value in seconds or particular date after which the service should be retried, this value is populated in Retry-After header while sending response from Ingress Gateway. The header is populated only when the exception occurred at Ingress Gateway matches the configured errorScenario's exceptionType parameter, the errorCode configured for the particular errorScenario lies in 3xx error series and the retry-after field for the particular errorScenario is configured appropriately in seconds.	**DataType**: integer **Mandatory(M)/Optional(O)/Conditional(C)**: O
errorDescription	errorDescription field in an errorScenario determines the description that needs to be populated in ProblemDetails (Detail field) response from Ingress Gateway when the exception occurred at Ingress Gateway matches the configured errorScenario's exceptionType field.	**DataType**: integer **Mandatory(M)/Optional(O)/Conditional(C)**: O

5.3.4 Error Code Series Configuration in Ingress Gateway

The following URI can be used to update the errorcodeserieslist used in Overload Control feature to list the configurable exception or error for an error scenario in Ingress Gateway.

URI: {apiRoot}/nrf/nf-common-component/v1/igw/errorcodeserieslist

Method:

- GET: Get Error Code Series configuration of the required service.
- PUT: Update Error Code Series configuration of the required service.

Content Type: application/json

Body:

```
[
    {
        "id": "E1",
        "exceptionList": [
            "RequestTimeout",
            "ConnectionTimeout",
            "UnknownHostException",
            "NotFoundException"

        ],
        "errorCodeSeries":
        [
            {
                "errorSet": "4xx",
                "errorCodes": [400, 408]
            },
            {
```

```
                    "errorSet": "5xx",
                    "errorCodes": [500, 503]
                }
            ]
        }
    ]
```

Table 5-12 Error Code Series Configuration

Attribute Name	Description	Details
id	Indicates the error code identifier.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
exceptionList	Lists the configurable exception or error for an error scenario in Ingress Gateway. The only supported values are: ConnectionTimeout, RequestTimeout, UnknownHostException, ConnectException, RejectedExecutionException, InternalError and NotFoundException, ClosedChannelException, BlackListIpException	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
errorCodeSeries	Lists the error codes for a specific service. **Note**: "ErrorCodeSeries" is configured only if a set of error responses with specific error codes is expected in server header. If it is not configured then all the error responses will have server header.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
errorSet	Possible values for "errorSet" attribute: 5xx, 4xx, 3xx, 2xx, 1xx	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
errorCodes	Possible values include all error codes in the respective HttpSeries value assigned for "errorSet". **Note**: Use single value of "-1" if all error codes in that HttpSeries are to be considered.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M

5.3.5 Routes Configuration in Ingress Gateway

The following URI can be used to configure the route ID and configuration in Ingress Gateway.

URI: {apiRoot}/nrf/nf-common-component/v1/igw/routesconfiguration

Method: PUT

Content Type: application/json

Body:

```
[
  {
    "id": "disc_mapping",
     "failureReqCountErrorCodeSeriesId": "E1"
  },
  {
    "id": "registration_mapping",
    "failureReqCountErrorCodeSeriesId": "E1"
```

```
  },
  {
    "id": "accesstoken_mapping",
     "failureReqCountErrorCodeSeriesId": "E1"
  }
]
```

Table 5-13 Routes Configuration

Attribute Name	Description	Details
id	Value of "id" attribute defines a specific service for route configuration.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M
failureReqCountErrorCodeSeriesId	Indicates the ID that is used to map the service with the error code ID defined in errorcodeserieslist.	**DataType**: string **Mandatory(M)/Optional(O)/Conditional(C)**: M

5.3.6 Controlled Shutdown Error Mapping Configuration

The following URI can be used for mapping between the routes and the error code profile that is used when the Ingress Gateway rejects incoming requests.

URI: {apiRoot}/nrf/nf-common-component/v1/igw/controlledshutdownerrormapping

Method:

- GET: Gets the mapping between the routes and the error code profile.
- PUT: Updates the mapping between the routes and the error code profile.

Content Type: application/json

Body:

```
{
   "routeErrorProfileList": [
      {
         "routeId":"disc_mapping",
         "errorProfileName":"shutdownerror503"
      },
      {
         "routeId":"registration_mapping",
         "errorProfileName":"shutdownerror503"
      },
      {
         "routeId":"accesstoken_mapping",
         "errorProfileName":"shutdownerror503"
      },
      {
         "routeId":"subscription_mapping",
         "errorProfileName":"shutdownerror503"
      }
   ]
}
```

Table 5-14 Controlled Shutdown Error Mapping

Attribute Name	Description	Details
routeErrorProfileList.routeId	The route id that is configured in routes configuration.	**DataType**: string **Constraints**: NA **Possible values:** <subscription_mapping,accesstoken_mapping,registration_mapping,disc_mapping> The possible values should match with the routeId of the routes configuration. **Default Value:** NA
routeErrorProfileList.errorProfileName	The error profile name that is used to fetch the error from errorcodeprofiles.	**DataType**: string **Constraints**: NA **Default Value:** shutdownerror503

5.3.7 CCA Header Validation

The following URI can be used for configuring the CCA header validation requests.

URI: `{apiRoot}/{nfType}/nf-common-component/v1/igw/ccaheader`

Method:

- GET: Gets the details stored in DB for property ccaheader.
- PUT: Replaces the existing values with the new value sent.
- PATCH: Updates the value of the parameters modified.

Content Type: application/json

Body:

```
{
 "enabled": false,
 "minExpiryTime": 2000,
 "maxTokenAge": 20,
 "role": "NRF",
 "subKey": "subjectAltName",
 "validationRule": "strict",
 "k8SecretName": "ocingress-secret",
 "k8NameSpace": "ocingress-ns",
 "fileName": "caroot.cer"
}
```

Table 5-15 CCA Header Validation Configuration

Attribute Name	Description	Details
`enabled`	Indicates if the CCA validation feature flag is enabled or disabled. **Note**: This is a read-only parameter. If you want to enable this feature, use `metadata.ccaHeaderValidation.enabled` parameter in routesConfig for `accesstoken_mapping` id in custom values file.	**Data Type**: boolean **Constraints**: true, false **Default Value:** false **Presence:** M
`minExpiryTime`	Indicates the buffer or additional time that is added to the current time to check it with the certificate expiry time.	**Data Type**: integer **Constraints**:NA **Default Value:** 0 **Presence:** M
`maxTokenAge`	Indicates the maximum token age allowed. **Note:** It skips the check if the value is 0.	**Data Type**: integer **Constraints**: 0 - 86400 seconds **Default Value:** 0 **Recommended Value:** 60 for NRF. **Presence:** M
`role`	Indicates the CCA token validator role to match the audience claim in the header. **Note:** This is a read-only parameter.	**Data Type**: string **Constraints**: NA **Default Value:** NRF **Presence:** M
`subKey`	Indicates if the certificate extension name to be read in the public key certificate received in CCA request for consumer NFs Instance Id. For NRF this value should be NF Instance Id. **Note**: This is a read-only parameter.	**Data Type**: string **Constraints**: NA **Default Value:** subjectAltName **Presence:** M
`validationRule`	Indicates the CCA validation rule. • strict: If "Issued At"+"Maximum Token Age Allowed "> current time, then the NRF validation rules are applied. • relaxed: "Issued At" validation is not mandatory. Validation rules are applied to Producer NF. For NRF, this value is always strict.	**Data Type**: string **Constraints**: strict, relaxed **Default Value:** strict **Presence:** M
`k8SecretName`	Indicates the name of the Kubernetes secret in which the CA bundle is present. **Note**: After fresh installation, these attributes must be configured with appropriate values as per the system requirement using the Rest API as these values are sample ones. Example `k8SecretName: ocingress-secret`	**Data Type**: string **Constraints**: NA **Default Value:** NA **Presence:** M

Table 5-15 (Cont.) CCA Header Validation Configuration

Attribute Name	Description	Details
k8SecretNameSpace	Indicates the Kubernetes namespace in which CA bundle is present. **Note**: changens value is updated with NRF secret for CCA header. **Note**: After fresh installation, these attributes must be configured with appropriate values as per the system requirement using the Rest API as these values are sample ones. Example `k8NameSpace: ocingress-ns`	**Data Type**: string **Constraints**: NA **Default Value:** NA **Presence:** M
fileName	Indicates the name of the CA bundle file used for CCA. This is the file generated by certificate and key generation steps. **Note**: After fresh installation, these attributes must be configured with appropriate values as per the system requirement using the Rest API as these values are sample ones. Example: `fileName:caroot.cer`	**Data Type**: string **Constraints**: NA **Default Value:** NA **Presence:** M

5.3.8 Ingress Gateway Pod Protection Options

URI: *{apiRoot}/nf-common-component/v1/igw/podprotection*

Method: PUT and GET

- GET: Retrieves Ingress Gateway pod protection options configuration.
- PUT: Enables or Disables Ingress Gateway pod protection feature.

Content Type: application/json

Body:

```
{
        "enabled": true,
        "monitoringInterval": 100,
        "congestionControl": {
            "enabled": true,
            "stateChangeSampleCount": 10,
            "actionSamplingPeriod": 3,
            "states": [
                {
                    "name": "Normal",
                    "weight": 0,
                    "entryAction": [
                        {
                            "action": "MaxConcurrentStreamsUpdate",
                            "arguments": {
                                "incrementBy": 30,
                                "incrementByActionSamplingPeriod": 3,
                                "maxConcurrentStreamsPerCon": 100
                            }
                        },
```

```
            {
                "action": "AcceptIncomingConnections",
                "arguments": {
                    "accept": true
                }
            }
        ]
    },
    {
        "name": "DoC",
        "weight": 1,
        "resourceThreshold": {
            "pendingMessage": 1500,
            "CPU": 75
        },
        "entryAction": [
            {
                "action": "AcceptIncomingConnections",
                "arguments": {
                    "accept": false
                }
            },
            {
                "action": "MaxConcurrentStreamsUpdate",
                "arguments": {
                    "incrementBy": 30,
                    "incrementByActionSamplingPeriod": 3,
                    "decrementBy": 30,
                    "decrementByActionSamplingPeriod": 1,
                    "maxConcurrentStreamsPerCon": 10
                }
            }
        ]
    },
    {
        "name": "Congested",
        "weight": 2,
        "resourceThreshold": {
           "pendingMessage": 2000,
           "CPU": 85
        },
        "entryAction": [
            {
                "action": "AcceptIncomingConnections",
                "arguments": {
                    "accept": false
                }
            },
            {
                "action": "MaxConcurrentStreamsUpdate",
                "arguments": {
                    "decrementBy": 30,
                    "decrementByActionSamplingPeriod": 1,
                    "maxConcurrentStreamsPerCon": 1
                }
            }
```

```
                        ]
                    }
                ]
            }
        }
```

Table 5-16 Pod Protection Options

Attribute	Description	Details
`enabled`	This attribute indicates if the Pod Protection feature is enabled or disabled.	**DataType**: Boolean **Constraints**: true, false **Default Value**: false
`monitoringInterval`	This attribute indicates the periodicity at which the overload state is monitored. **Unit**: Milliseconds **Note:** This is a read-only attribute. The proposed value for this attribute is 100.	**DataType**: Integer **Constraints**: NA **Default Value**: NA
`congestionControl`	This attribute specifies the congestion control configuration. For more information about congestion control parameters, see Table 2-72.	**DataType**: Object **Constraints**: NA **Default Value**: NA

Table 5-17 CongestionControlConfig

Attribute	Description	Details
`enabled`	This attribute allows the configuration of pod protection attributes for the Ingress Gateway pods. **Note**: This must be set to true for Pod Protection feature.	**DataType**: Boolean **Constraints**: true, false **Default Value**: false
`actionSamplingPeriod`	This attribute indicates the interval at which the configured action must be considered. The actions are configured under `entryAction.action` attribute. The interval is calculated as (`actionSamplingPeriod` * `monitoringInterval`). **Note:** This is a read-only attribute. The proposed value for this attribute is 3.	**DataType**: Integer **Constraints**: NA **Default Value**: NA
`stateChangeSampleCount`	This attribute indicates the number of times the pod must be in the particular congestion state before transitioning to another state. For example, if the current state is normal, and the new state is DoC, then NRF moves the pod to DoC only if the state is reported for 10 times in 1 second (`stateChangeSampleCount` * `monitoringInterval`). **Note:** This is a read-only attribute. The proposed value for this attribute is 10.	**DataType**: Integer **Constraints**: NA **Default Value**: NA

Table 5-17 (Cont.) CongestionControlConfig

Attribute	Description	Details
states	This attribute indicates the congestion states, the thresholds, and corresponding actions. For more information about congestion states, see Table 2-73.	**DataType**: Object **Constraints**: NA **Default Value**: NA

Table 5-18 CongestionStates

Attribute	Description	Details
name	The name of the congestion state. • Normal: The pod is not in overload state. • Danger of Congestion (Doc): The pod is about to go into the congested state. Actions configured in the entryAction is performed. • Congested state: The pod is in congested state. Actions configured in the entryAction is performed. **Note:** This is a read-only attribute.	**DataType**: String **Constraints**: Normal, Doc, Congested **Default Value**: NA
weight	The weight of the congestion state. The weight indicates the critical of the congestion state. The lower the value, the lower the criticality. **Note:** This is a read-only attribute. The proposed value for this attribute is for Normal is 0, DoC is 1, and Congested is 2.	**DataType**: Integer **Constraints**: NA **Default Value**: NA
entryAction	This attribute indicates the actions for the congestion state. For more information about the entry action configuration, see Table 2-74. **Note:** This is a read-only attribute.	**DataType**: List **Constraints**: NA **Default Value**: NA
resourceThreshold	This attribute indicates the resource thresholds for the given congestion state. This configuration is mandatory for the 'DoC' and 'Congested' states. For more information about the threshold for each resources, see Table 2-75. **Note:** This is a read-only attribute.	**DataType**: Object **Constraints**: NA **Default Value**: NA

Table 5-19 EntryActionConfig

Attribute	Description	Details
`action`	This attribute indicates the action for the congestion state. • `AcceptIncomingConnections`: The action indicates whether the incoming new connection is accepted or rejected based on the overload state. • `MaxConcurrentStreamsUpdate`: The action indicates whether to increase or decrease the max concurrent stream for all incoming connections till the maxConcurrentStreamsPerCon is reached.	**DataType**: String **Constraints**: MaxConcurrentStreamsUpdate, AcceptIncomingConnections **Default Value**: NA
`arguments`	This attribute indicates the actions for the congestion state. For more information about the arguments, see Table 2-76.	**DataType**: Map <String, Object> **Constraints**: NA **Default Value**: NA

Table 5-20 ResourceThreshold

Attribute	Description	Details
`cpu`	The CPU threshold is expressed in percentage. The recommended value for this attribute is for DoC is 75 and for Congested is 85. **Note:** The default value is recommended and must not be changed.	**DataType**: Integer **Constraints**: NA **Default Value**: NA
`pendingMessage`	The number of pending messages to be processed, expressed in absolute count. The recommended value for this attribute is for DoC is 1500 and for Congested is 2000. **Note:** The default value is recommended and must not be changed.	**DataType**: Integer **Constraints**: NA **Default Value**: NA

Table 5-21 Possible Arguments

Attribute	Description	Details
`accept`	The attribute indicates if the incoming connection should be accepted or not. Applicable when the action is `AcceptIncomingConnections`. true: The incoming connection is accepted. false: The incoming connection is rejected. The proposed value for Normal state is true, and for DoC and Congested state is false.	**DataType**: Boolean **Constraints**: true, false **Default Value**: NA
`incrementBy`	The attribute indicates the factor by which the current concurrent streams value will be incremented till it reaches `maxConcurrentStreamsPerCon`. **Note**: This is preconfigured for Normal and DoC state. The proposed value for this attribute is 30.	**DataType**: Integer **Constraints**: NA **Default Value**: NA

Table 5-21 (Cont.) Possible Arguments

Attribute	Description	Details
`decrementBy`	The attribute indicates the factor by which the current concurrent streams value will be decremented till it reaches `maxConcurrentStreamsPerCon`. **Note**: This is preconfigured for DoC and Congested state. The proposed value for this attribute is 30.	**DataType**: Integer **Constraints**: NA **Default Value**: NA
`maxConcurrentStreamsPerCon`	The attribute indicates the maximum number of concurrent streams per connection allowed. The proposed value for this attribute is for Normal is 100, DoC is 10, and Congested is 1.	**DataType**: Integer **Constraints**: NA **Default Value**: NA
`decrementByActionSamplingPeriod`	The attribute indicates the time interval at which the decrementBy is applied to reach `maxConcurrentStreamsPerCon`. If not provided, the `actionSamplingPeriod` is used. **Note**: This is preconfigured for DoC and Congested state. The proposed value for this attribute is 1. Unit is seconds.	**DataType**: Integer **Constraints**: NA **Default Value**: NA
`incrementByActionSamplingPeriod`	The attribute indicates the time interval at which the incrementBy is applied to reach `maxConcurrentStreamsPerCon`. If not provided, the `actionSamplingPeriod` is used. **Note**: This is preconfigured for Normal and DoC state. The proposed value for this attribute is 3. Unit is seconds.	**DataType**: Integer **Constraints**: NA **Default Value**: NA