

Oracle® Communications

Cloud Native Core, Network Exposure Function REST Specification Guide



Release 24.2.2
F98164-03
September 2025

ORACLE®

Copyright © 2022, 2025, Oracle and/or its affiliates.

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

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

If this is software, software documentation, data (as defined in the Federal Acquisition Regulation), or related documentation that is delivered to the U.S. Government or anyone licensing it on behalf of the U.S. Government, then the following notice is applicable:

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

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

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

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

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

Contents

1	Introduction	
1.1	References	1
2	NEF Configuration Using REST APIs	
2.1	Supported REST API Interfaces	1
2.1.1	Discovery Groups Management	3
2.1.1.1	Get All Registrations	3
2.1.1.2	Get Service APIs	5
2.1.1.3	Create Discovery Group	8
2.1.1.4	Delete Discovery Group	10
2.1.2	API Invoker Provisioning	11
2.1.2.1	Pre-provision API Invoker	11
2.1.2.2	Delete Pre-provisioned API Invoker	13
2.1.2.3	Update Pre-provisioned API Invoker	13
2.1.2.4	Fetch Pre-provisioned API Invoker	15
2.1.3	API Invoker Management	16
2.1.3.1	Onboard API Invoker	16
2.1.3.2	Offboard API Invoker	20
2.1.4	NEF Security Management	21
2.1.4.1	Obtain Security Method	21
2.1.4.2	Delete Security Method	24
2.1.4.3	Obtain Authorization	25
2.1.5	CAPIF Event Management	27
2.1.5.1	Create CAPIF Event Subscription	27
2.1.5.2	Delete CAPIF Event Subscription	30
2.1.5.3	Publish CAPIF Event	31
2.1.6	Monitoring Event Subscription	33
2.1.6.1	Create ME Subscription	33
2.1.6.2	Get ME Subscription	44
2.1.6.3	Delete an ME Subscription	47
2.1.6.4	Update ME Subscription	48
2.1.6.5	ME Notification	53
2.1.7	ConfigMapper API	56

2.1.7.1	Update Configuration Mapping	57
2.1.7.2	GET Configuration Mapping	58
2.1.7.3	GET All Pre-configured Mapping	58
2.1.7.4	Delete Configuration Mapping	59
2.1.8	AF Session with QoS	60
2.1.8.1	Create AF Session with QoS Subscription	60
2.1.8.2	Get QoS Session Details	65
2.1.8.3	Update (PUT) AF Session with QoS Subscription	68
2.1.8.4	Update (Patch) AF Session with QoS Subscription	72
2.1.8.5	Delete a QoS Subscription	76
2.1.8.6	QoS Notification	77
2.1.9	Traffic Influence Subscription	80
2.1.9.1	Create Traffic Influence Subscription	80
2.1.9.2	Get Traffic Influence Subscription	84
2.1.9.3	Update Traffic Influence Subscription	86
2.1.9.4	Update (Patch) Traffic Influence Subscription	89
2.1.9.5	Delete Traffic Influence Subscription	92
2.1.9.6	Traffic Influence Notification	93
2.1.10	Device Trigger Transaction	95
2.1.10.1	Create Device Trigger Transaction	95
2.1.10.2	Get Individual Device Trigger Transaction	97
2.1.10.3	Delete Individual Device Trigger Transaction	99
2.1.11	CNC Console Integration	100
2.1.11.1	Configuring NEF Features	100
2.1.11.2	Configuring CAPIF Features	164

What's New in This Guide

This section introduces the documentation updates for Release 24.2.x.

Release 24.2.2 - F98164-03, September 2025

Updated the `monitorExpireTime` parameter description in the [Create ME Subscription](#) and [Update ME Subscription](#) sections.

Release 24.2.1 - F98164-02, October 2024

There are no changes in this release.

Release 24.2.0 - F98164-01, July 2024

The following changes are made in this document:

- Added 504 error code to Supported Response Codes in the [Create ME Subscription](#) section.
- Updated description for `locationType` in the [Create ME Subscription](#) section.
- Added a note on AF request containing `LocationType` or `LdrType` in the [Create ME Subscription](#) section.
- Updated the following sections as part of Integration with CNC Console feature:
 - [Fetching Logging Data](#)
 - [Fetch Timeout Data](#)
 - [Fetch GMLC Configuration](#)
 - [Fetch AF Service ID List](#)
 - [Fetch Global Configurations](#)
 - [Add Service ID List](#)
 - [Delete Service ID List](#)
 - [Fetch SCS Short Message Entity](#)
 - [Update SCS Short Message Entity](#)
 - [Fetch QoS Profiles](#)
- Added the following sections as part of Integration with CNC Console feature:
 - [Update GMLC Configuration](#)
 - [Edit Service ID List](#)
 - [Update QoS Profiles](#)
 - [Add QoS Profiles](#)
 - [Delete QoS Profiles](#)
 - [Fetch PLMN ID Map](#)
 - [Add PLMN ID Map](#)
 - [Edit PLMN ID Map](#)
 - [Delete PLMN ID Map](#)

- [Fetch TAI Map](#)
- [Add to TAI MAP](#)
- [Edit TAI MAP](#)
- [Delete from TAI Map](#)
- [Fetch ECGI Map](#)
- [Add to ECGI Map](#)
- [Edit ECGI Map](#)
- [Delete from ECGI Map](#)
- [Fetch NCGI Map](#)
- [Add to NCGI Map](#)
- [Edit NCGI Map](#)
- [Delete from NCGI Map](#)
- [Fetch GNBID Map](#)
- [Add to GNBID Map](#)
- [Edit GNBID Map](#)
- [Delete from GNBID Map](#)
- [Fetch globalRanNodeId Map](#)
- [Add to globalRanNodeId Map](#)
- [Edit globalRanNodeId Map](#)
- [Delete from globalRanNodeId Map](#)
- [Fetch geoZoneIdToSpatialValidity Map](#)
- [Add to geoZoneIdToSpatialValidity Map](#)
- [Edit geoZoneIdToSpatialValidity Map](#)
- [Delete from geoZoneIdToSpatialValidity Map](#)
- [Fetching Logging Data](#)
- [Updating Logging Data](#)
- [Fetching Published Services from CAPIF](#)
- [Create Discovery Group](#)
- [Get Discovery Group](#)
- [Get All Discovery Group](#)
- [Delete Discovery Group](#)
- [Pre-provision the Invoker](#)
- [Get Invoker Details](#)
- [Get All Invoker Details](#)
- [Delete Invoker](#)
- [Get Access Token](#)
- [Update Access Token Expiry Time](#)

- Added a note on updating configurations only through Rest API in [CNC Console Integration](#) section.

1

Introduction

This document provides information about how to configure the services and manageable objects in Oracle Communications Cloud Native Core, Network Exposure Function (NEF) using Representational State Transfer Application Programming Interfaces (REST APIs).

The core network in 5G follows a service-based architecture, where network functions provide services that can be utilized using REST APIs by other functions. This allows the adoption of web-scale technologies and software that are used by different organizations in the telecommunications network.

For more information about NEF functionalities and features, see *Oracle Communications Cloud Native Core, Network Exposure Function (NEF) User Guide*.

Note

The performance and capacity of the NEF system may vary based on the call model, feature or interface configuration, network conditions, and underlying CNE and hardware environment.

1.1 References

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

2

NEF Configuration Using REST APIs

This chapter provides information about how to configure Oracle Communications Cloud Native Core, Network Exposure Function (NEF) using REST APIs.

NEF can be configured using Helm and REST configurations. Some configurations are performed during NEF installation using Helm and a few configurations are modified using REST APIs. REST configurations can also be performed using the Oracle Communications Cloud Native Configuration Console (CNC Console).

For more information about NEF configuration types, see the following guides:

- For Helm configuration: Oracle Communications Cloud Native Core, Network Exposure Function Installation, Upgrade, and Fault Recovery Guide.
- For REST configurations using the CNC Console: Oracle Communications Cloud Native Core, Network Exposure Function User Guide.

NEF uses GET, PUT, DELETE, and so on HTTP methods to retrieve, update, and remove REST API configuration data.

Note

- The PUT method for multiple rules as a single payload through a single API is not allowed unless specified explicitly.
- If the query parameter is incorrect, NEF does not return an error response but considers an API call without any query parameter, and therefore the GET ALL output is returned in the response.
- The CAPIF or NEF configuration Rest API's do not support any semantic validation for every field. It is operator's responsibility to populate valid values in the Rest API and CNC Console.

2.1 Supported REST API Interfaces

Note

If TLS is enabled in NEF, then for the API interfaces use the curl command with https.**Command:**

```
curl --capath . --cacert caroot.cer --cert rsa_certificate.crt --key
    rsa_private_key_pkcs1.pem --cert-type PEM -v --http2-prior-
knowledge --request POST
    \ --url https://<IP>:<TLS_PORT>/<URL> \ --header 'Accept:
application/json' \ --header 'Content-Type: application/json' \ --
data '<REQUEST_BODY>'
```

where,

The --cert option is used to specify the client certificate file. The client certificate is used for authentication purposes.

The --cacert option is used to specify the CA certificate file. The CA certificate is used to verify the server's certificate.

The --capath option is used to specify the directory containing key and certificates.

The --key option is used to specify the private key file.

The --cert-type option is used to specify the type of the provided certificate.

Sample https curl command:

```
curl --capath . --cacert caroot.cer --cert rsa_certificate.crt --key
    rsa_private_key_pkcs1.pem --cert-type PEM -v --http2-prior-
knowledge --request POST \
    --url https://10.75.240.98:443/3gpp-monitoring-event/v1/4e4df956-
a7fe-484e-a595-0dea9235cd52/subscriptions \
    --header 'Accept: application/json' \
    --header 'Authorization: Bearer
eyJ0eXAiOiJKV1QiLCJpbmZva2VyTmFtZSI6ImFmNCIsImNhcnGlmSW5zdGFuY2VJZCI6ImY
5OGIwNWUwLTlYzDgtMTFlZC04NjFkLTAYNDJhYzEyMDAwMiIsImFsZyI6IjE1JTMjU2In0.eyJ
Jpc3MiOiI0ZTRkZjk1Ni1hN2ZlLTQ4NGUyTU5NS0wZGVhOTIzNWwNNTIiLCJzY29wZSI6I
jNncHAjZTE5YWQ0NmEtYTcwYy00N2IzLTgwNmMtNGQ1MzI3OWFhOWI30jNncHAtbW9uaXRv
cmLuZy1ldmVudCIsImV4cCI6MTY3NjI3NzU5MjU0LHZXEMmkmUy25y-
nATQnM0mqY23MBL17U8byzVcfyKRnj-EPg9JzW0zZ-D-
GvRFwyDq9UvQ8fef37SNDCAo_VbC-
a8zfdNqFz0_ij1AyeIc4HgA5m_6fTIjgKtbiFmCEBNNVtdQkZpgM0EJVNOMKW1Kr7hSY_Mo
-wtFH0BLWU3lU6x6aZCU5d81VnzgFd6zCXUNI2KdbGepF-Ua-
sDLEEXPX1I9Em0f8jVUgEI7gnQisudTi1IRRqCIUWnFopync67lJNka_kt-MNx-
XObFuUFDaxTpemTftFic0H0l06XmiwUjNqv-D0tLwU19-
uBk29Cy1SnI0wWcgebRw8aeMLmaA' \
    --header 'Content-Type: application/json' \
    --data '{
{
    "externalId": "5test1q895@oracle.com",
    "monitoringType": "LOCATION_REPORTING",
    "locationType": "CURRENT_LOCATION",
    "maximumNumberOfReports": 1000,
    "accuracy": "CGI_ECGI",
    "notificationDestination":
```

```
① }' "https://ocnefsim-ocstub-svc-af:1010/af/notifications"
```

2.1.1 Discovery Groups Management

NEF allows the operators to create discovery groups and assign them to API Invokers for providing administered access to service APIs . The discovery group is a framework provided by CAPIF to define a set of API Exposing Functions (AEFs) with corresponding APIs. The operators assign these groups to API invokers for defining the NEF APIs accessible to them.

The specifications of a discovery group are as follows:

- A discovery group consists of information about the API providers, AEFs, and the APIs which were published.
- Discovery groups can be created by operators using NEF discovery-group REST API.
- Operators can assign discovery groups to API invokers during pre-provisioning using the pre-provisioning REST API.
- The group information is used by NEF to expose only the APIs linked to the group. Also, on receiving a successful onboarding response for API invoker, the designated group details are used for providing a response with the details of APIs accessible to the invoker.

The supported REST APIs for discovery group creation are as follows:

1. **Get AEF Registrations:** Allows the operator to send the GET registrations request to fetch the list of AEFs registered with NEF along with the AEF IDs.
2. **Get Service APIs:** This is a 3GPP defined REST API. It is used to send the GET service-apis request to fetch the list of service APIs corresponding to the AEF IDs. The API response provides the list of APIs and the corresponding AEFs.

① Note

The Get AEF Registrations and Get Service APIs

3. **Create Discovery Group:** Allows the operator to select the list of APIs corresponding to each AEF and send a POST discovery-group request to create discovery groups.

① Note

To create a discovery group, you must know the AEF IDs and the list of APIs that are published by each AEF. Use the Get AEF Registrations and Get Service APIs operations to get these details.

4. **Delete Discovery Group:** Allows the operator to delete any discovery groups.

2.1.1.1 Get All Registrations

This service operation is used to get details of all the API provider domains registered with NEF.

Type: GET

URI: /api-provider-management/v1/registrations**Initiated By:** NEF Administrator**! Important**

The API domain ID received in response to this operation is used as AEF, APF and AMF ID.

Table 2-1 Response Body Parameters

Field Name	Data Type	Description
apiProvDomId	string	The unique identifier assigned to the NEF API management service while registering with the API provider domain.
apiProvDomInfo	string	Generic information related to the API provider domain such as details of the API provider applications.
apiProvFuncs	array	This is a domain functions details list of the individual API provider.
apiProvFuncId	string	The unique function identifier assigned to the NEF API management service while registering/ updating the API provider domain. Note: This ID is same as the apiProvDomId.
regInfo	object	It is the required information for the CAPIF core function to register the API provider domain function.
apiProvFuncInfo	string	Generic information related to the API provider domain function such as details of the API provider applications.
apiProvFuncRole	string	Role of API provider domain function.
regInfo.apiProvCert	string	Generic client certificate of the API provider domain function.
regInfo.apiProvPubKey	string	Public Key of API Provider domain function.

Examples

The following example shows how you can get details of all the available API provision domain IDs by submitting a GET request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request GET 'http://<IP
address:Port>/api-provider-management/v1/registrations' \
--header 'Content-Type: application/json'
```

Example of the Request Body

The following snippet shows an example of the request body.

```
[
  {
    "apiProvDomId" : "daf7bb19-68a9-4c12-8767-4202445d4c4f",
```

```

"apiProvDomInfo" : "apiProvDomInfo",
"apiProvFuncs" : [
  {
    "apiProvFuncId" : "daf7bb19-68a9-4c12-8767-4202445d4c4f",
    "apiProvFuncInfo" : "API--Prov--Func--Info--APF",
    "apiProvFuncRole" : "APF",
    "regInfo" : {
      "apiProvCert" : "qwerty234",
      "apiProvPubKey" : "qwerty123"
    }
  },
  {
    "apiProvFuncId" : "daf7bb19-68a9-4c12-8767-4202445d4c4f",
    "apiProvFuncInfo" : "API--Prov--Func--Info--AMF",
    "apiProvFuncRole" : "AMF",
    "regInfo" : {
      "apiProvCert" : "qwerty234",
      "apiProvPubKey" : "qwerty123"
    }
  },
  {
    "apiProvFuncId" : "daf7bb19-68a9-4c12-8767-4202445d4c4f",
    "apiProvFuncInfo" : "API--Prov--Func--Info--AEF",
    "apiProvFuncRole" : "AEF",
    "regInfo" : {
      "apiProvCert" : "qwerty234",
      "apiProvPubKey" : "qwerty123"
    }
  }
],
"failReason" : null,
"regSec" : "RegSec",
"suppFeat" : "BC"
}
]

```

2.1.1.2 Get Service APIs

This service operation is used to get the list of NEF service APIs.

Type: GET

URI: /published-apis/v1/<apiProvDomId/AFID>/service-apis

Initiated By: NEF Administrator

Table 2-2 Response Body Parameters

Field Name	Data Type	Description
apiName	string	REST API name
aefProfiles	array	The AEF profile information, which includes the exposed API details.
aefId	string	The AEF identifier
domainName	string	Domain name of the API

Table 2-2 (Cont.) Response Body Parameters

Field Name	Data Type	Description
versions	array	Version details of the API
apiVersion	string	API major version
expiry	string	Expiry date and time of the API version
resources	array	Resources supported by the API
resourceName	string	Name of the resource
custOperations	array	Custom operations without resource association.
commType	string	Communication type used by the API.
uri	string	The communication path used by the API
custOpName	string	Part of the URI structure for a custom operation associated with a resource
operations	array	Supported HTTP methods for the API resource.
description	string	Text description of the API resource.
protocol	string	The Protocol invoked by the API
dataFormat	string	The service API category to which the service API belongs to
securityMethods	array	Security methods supported by the API
interfaceDescriptions	array	Interface details
ipv4Addr	string	String identifying an IPv4 address
port	string	Port
supportedFeatures	string	The supported features of the API
shareableInfo	array	Indicates whether the service API and the service API category can be shared to the list of provider domains.
capifProvDoms	string	List of provider domains to which the service API information to be shared.
serviceAPICategory	string	The service API category to which the service API belongs to.
apiSuppFeats	string	Indicates the features supported by the service API
pubApiPath	string	Contains the published API path
ccfIds	string	A list of CCF identifiers where the service API is already published

Examples

The following example shows how can you get details of all the available APIs by submitting a GET request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request GET 'http://<IP
address:Port>/published-apis/v1/79b4641d-066d-44f1-98ec-16d97bbc6b64/service-
apis' \--header 'Content-Type: application/json' \
```

Example of the Response Body

The following snippet shows an example of the request body.

```
[
{
  "apiName": "3gpp-monitoring-event",
  "aefProfiles": [
    {
      "aefId": "79b4641d-066d-44f1-98ec-16d97bbc6b64",
      "domainName": "https://localhost:8080",
      "versions": [
        {
          "apiVersion": "v1",
          "expiry": "2000-01-23T04:56:07Z",
          "resources": [
            {
              "resourceName": "resourceName",
              "commType": "REQUEST_RESPONSE",
              "uri": "uri",
              "custOpName": "custOpName",
              "operations": [
                "GET",
                "POST",
                "PUT",
                "DELETE"
              ],
              "description": "description"
            }
          ],
          "custOperations": [
            {
              "commType": "REQUEST_RESPONSE",
              "custOpName": "custOpName",
              "operations": [
                "GET",
                "POST",
                "PUT",
                "DELETE"
              ],
              "description": "description"
            }
          ]
        }
      ],
      "protocol": "HTTP2",
      "dataFormat": "JSON",
      "securityMethods": [
        "OAUTH",
        "PKI"
      ],
      "interfaceDescriptions": [
        {
          "ipv4Addr": "ipv4Addr",
          "port": 5248,
          "securityMethods": [
            null,
            null
          ]
        }
      ]
    }
  ]
}
```

```

        ]
      },
      {
        "ipv6Addr": "ipv6Addr",
        "port": 5248,
        "securityMethods": [
          null,
          null
        ]
      }
    ]
  },
  "description": "description",
  "supportedFeatures": "af",
  "shareableInfo": {
    "isShareable": true,
    "capifProvDoms": [
      "capifProvDoms",
      "capifProvDoms"
    ]
  },
  "serviceAPICategory": "serviceAPICategory",
  "apiSuppFeats": "af",
  "pubApiPath": {
    "ccfIds": [
      "ccfIds",
      "ccfIds"
    ]
  },
  "ccfId": "ccfId"
}
]

```

2.1.1.3 Create Discovery Group

This service operation is used by operators to create discovery groups for enabling restricted access of published APIs to API invokers.

Type: POST

URI: /discovery-group/v1/apiProvidersGroup

Initiated By: NEF Administrator

Prerequisite: The AEF IDs and list of APIs that are published by each AEF must be available to create a group. To get these details use the following operations:

- [Get All Registrations:](#)
- [Get Service APIs](#)

Table 2-3 Request Body Parameters

Field Name	Data Type	Description
groupId	string	The unique discovery group identifier

Table 2-3 (Cont.) Request Body Parameters

Field Name	Data Type	Description
apiProviderInfo	string	A list of provider group details. Each group shall comprises of list of API providers , AEFs and APIs information
apiProviderId	string	provider domain id or registration id created during provider registration
aefIds	string	A list of Aef details. It will have list of AEF's and API associated with each AEF
aefId	string	provider domain id or registration id created during provider registration. AEF Id is same as registration id.
apiName	long	apiName published to capif.
nidd		
monitoring		

Table 2-4 Supported Response Codes

Code	Description
201	Successful group creation
503	Error while creating group

Examples

The following example shows how a discovery group is created by submitting a POST request on the REST resource using cURL.

cURL Command

```
curl -X POST --http2-prior-knowledge\ http://<IP Address:Port>/discovery-  
group/v1/apiProvidersGroup \ --header 'Content-Type: application/json' \ -H
```

Example of the Request Body

The following snippet shows an example of the request body.

```
{
  "groupId": "group001",
  "apiProviderInfo": [
    {
      "apiProviderId": "9724655a-c6a3-4aa1-bdbe-91250def43e3",
      "aefIds": [
        {
          "aefId": "9724655a-c6a3-4aa1-bdbe-91250def43e3",
          "apiName": [
            "3gpp-monitoring-event"
          ]
        }
      ]
    }
  ]
}
```

```
]
}
```

Example of the Response Body

The following example shows the contents of the response body in JSON format.

```
{
  "groupId": "group001",
  "apiProviderInfo": [
    {
      "apiProviderId": "9724655a-c6a3-4aa1-bdbe-91250def43e3",
      "aefIds": [
        {
          "aefId": "9724655a-c6a3-4aa1-bdbe-91250def43e3",
          "apiName": [
            "3gpp-monitoring-event"
          ]
        }
      ]
    }
  ]
}
```

2.1.1.4 Delete Discovery Group

This service operation is used by operator to delete a discovery group.

Type: DELETE

URI: /discovery-group/v1/apiProvidersGroup/<group_id>

Initiated By: NEF Administrator

Table 2-5 Supported Response Codes

Code	Description
204	Delete successfully
404	Group not found
503	Error when deleting the group

Examples

The following example shows how a discovery group is deleted by submitting a DELETE request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request DELETE 'http://<IP
Address:Port>/apiRoot/discovery-group/v1/apiProvidersGroup/group001' --
header 'Content-Type: application/json' -v
```

2.1.2 API Invoker Provisioning

NEF uses the pre-provisioning REST API to publish the API Invoker details to CAPIF. Once the pre-provisioning is complete, the API invoker is enabled to onboard itself to the CAPIF.

2.1.2.1 Pre-provision API Invoker

This service operation is used by an API invoker to provide its details to CAPIF before getting onboarded.

Type: POST

URI: /pre-provisioning/v1/apiInvokers

Initiated By: Operator or Administrator

Table 2-6 Request Body Parameters

Field Name	Data Type	Description
apiInvokerName	string	The name of the API invoker. This is a mandatory parameter.
valid_from	string	The date and time from when the API Invoker provisioning starts. This is a mandatory parameter.
valid_to	string	The date and time by when the API Invoker provisioning remains valid. This is a mandatory parameter.
group_id	string	The unique identity of the AEF group assigned to the API Invoker.
email_address	string	The authorized email address. This is an optional parameter.
phone	long	The authorized phone number. This is an optional parameter. Note: This parameter is not consumed by CAPIF for any processing and additionally there is no configuration database with CAPIF to validate its value. Thus, it is not controlled by CAPIF.

Table 2-7 Response Body Parameters

Field Name	Data Type	Description
accessToken	string	The security access token for authentication purpose. This token shall be provided in onboarding request body (onboardingSecret parameter). This is a mandatory parameter.
onboardingUri	string	The onboarding URI that must be accessed by API Invoker to onboard NEF. This is a mandatory parameter.

Table 2-8 Supported Response Codes

Code	Description
201	API invoker onboarded successfully
400	Bad Request

Examples

The following example shows how an API Invoker is pre-provisioned by submitting a POST request on the REST resource using cURL.

cURL Command

```
curl -X POST --http2-prior-knowledge\ http://<IP Address:Port>/pre-
provisioning/v1/apiInvokers \ -H 'cache-control: no-cache' \ -H 'content-
type: application/json' \ -H
```

Example of the Request Body

The following snippet shows an example of the request body.

```
{
  "apiInvokerName": "af001",
  "valid_from": "2020-11-11T17:28:32Z",
  "valid_to": "2022-11-11T17:28:32Z",
  "group_id": "ocnef-group-1"
  "email_address": "",
  "phone": 1343435
}
```

Example of the Response Body

The following example shows the contents of the response body in JSON format.

```
{
  "accessToken": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpbnZva2VyTmFtZSI6ImIudm
9rZXItcHJvZmlsZSI6ImIudm9rZXJJZCI6ImVhODc3NTJlLTNhOGQtNDUyMS1hYzEzLTQ1ZTA3
OTNjYyIsInN1YiI6IkFwaUludm9rZXJQcmVQcm92aXNpb25pbmciLCJpYXQiOiJlNzc2NDA1MTIsIm
V4cCI6MTY0MDUzOTcxMn0.Y-
PndCI8SIeoAyzkp0ELk3hVA6qoTnELMa0ux4hgGDS3vdMweg0fvhxrvnAdT2_dFtA7z2pjYxJ8GuRc
1G48byhKqa9WWLRe-7RriRzVlwRfTTfQTfSRqZTxJunD5B33caS12pDQAKS-8qp1g3d_dJo1HzFzH6
bBXGCCq8jKe3enIHSwV-
B2xCTAi4tHdAqpxBHZA93f3hp59YU813tJyzuv6oQkxJ9HtH7jn_j7dN2Qhh5Nzmr8sc6KtdMrnFX
PdXDx28pgVY3kYLlexngmgpdgWD89diIeMUjKBcCfFS7JL22_Q12HqCAX_D99As6jrwPziFTbqVrEB
ra6p_iCg",
  "onboardingUri": "http://ocnef-ingressgateway:80/api-invoker-management/v1/
onboardedInvokers"
}
```

2.1.2.2 Delete Pre-provisioned API Invoker

This service operation is used by an API invoker to delete its details from CAPIF after offboarding.

Type: DELETE

URI: /pre-provisioning/v1/apiInvokers/<invokername>

Initiated By: Operator or Administrator

Table 2-9 Supported Response Codes

Code	Description
204	Delete successfully
404	Not found on unknown apiInvoker Name

Examples

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

cURL Command

```
curl --location --request DELETE '<IP Address:Port>/pre-provisioning/v1/apiInvokers/<invokername>'
```

2.1.2.3 Update Pre-provisioned API Invoker

This service operation is used by an API invoker to update its details in CAPIF.

Note

This API allows configuring **valid_to** and **valid_from** fields. Update requests raised for any other fields are ignored.

Type: PUT

URI: /pre-provisioning/v1/apiInvokers /{apiInvokerName}

Initiated By: Operator or Administrator

Table 2-10 Request Body Parameters

Field Name	Data Type	Description
apiInvokerName	string	It is the name of the API invoker. This is a mandatory parameter.
valid_from	string	It is the date and time from when the API Invokerprovisioning starts. This is a mandatory parameter.
valid_to	string	It is the date and time till when the API Invokerprovisioning remains valid. This is a mandatory parameter.

Table 2-10 (Cont.) Request Body Parameters

Field Name	Data Type	Description
group_id	string	It is the unique identity of the AEF group assigned to the API Invoker.
email_address	string	It is the authorized email address. This is a optional parameter.
phone	long	It is the authorized phone number. This is a optional parameter. Note: This parameter is not consumed by CAPIF for any processing and additionally there is no configuration database with CAPIF to validate its value. Thus, it is not controlled by CAPIF.

Table 2-11 Response Body Parameters

Field Name	Data Type	Description
accessToken	string	The security access token for authentication purpose. This token shall be provided in onboarding request body (onboardingSecret parameter). This is a mandatory parameter.
onboardingUri	string	The onboarding URI that must be accessed by API Invoker to onboard NEF. This is a mandatory parameter.

Table 2-12 Supported Response Codes

Code	Description
200	API Invoker updated successfully
400	Bad Request
404	API Invoker does not exists

Example:

The following example shows how an API Invoker is updating the previous pre-provisioned details by submitting a PUT request on the REST resource using cURL command.

cURL command

```
curl -X PUT --http2-prior-knowledge\ http://<IP Address:Port>/pre-
provisioning/v1/apiInvokers/invoker-profile-1 \ -H 'cache-control: no-cache'
\ -H 'content-type: application/json' \
```

Example of the Request Body

The following snippet shows an example of the request body.

```
{
  "apiInvokerName": "af001",
  "valid_from": "2020-11-11T17:28:32Z",
  "valid_to": "2022-11-11T17:28:32Z",
  "group_id": "ocnef-group-1"
  "email_address": "",
}
```

```
"phone": 1343435
}
```

Example of the Response Body

The following example shows the content of the response body in JSON format.

```
{
  "accessToken": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpbnZva2VyTmFtZSI6ImIudm9rZXItchJvZmlsZSI6ImIudm9rZXJJZCI6ImVhODc3NTJlLTNhOGQtNDUyMS1hYzEzLTQ1ZTA3OTNjYyIsInN1YiI6IkFwaUludm9rZXJQcmVQcm92aXNpb25pbmciLCJpYXQiOiJlNzc2NDA1MTIsImV4cCI6MTY0MDUzOTcxMn0.YPndCI8SioAyzkp0ELk3hva6qoTnELMaOux4hgGDS3vdMweg0fvhxrvaAdT2_dFtA7z2pjYxJ8GuRc1G48byhKqa9WWLRe-7RriRzVlwRfTTfQTfSRqZTxJunD5B33caSl2pDQAKS-8qplg3d_dJolHzFzH6bBXGCcq8jKe3enIHSwVB2xCTAi4tHdAqpxBHZaS93f3hp59YU813tJyzuv6oQkxJ9HtH7jn_j7dN2Qhh5Nzmr8sc6KtdMrnFXPdXdx28pgVY3kYLlexngmgpdgWD89diIeMUjKBcCfFS7JL22_Ql2HqCAx_D99As6jrwPziFTbqVrEBra6p_iCg", "onboardingUri": "http://ocnef-ingressgateway:80/api-invokermanagement/v1/onboardedInvokers"
}
```

Note

No re-onboarding is required with this new access token. Currently, there is no use of this access token.

2.1.2.4 Fetch Pre-provisioned API Invoker

This service operation is used by an API invoker to fetch its details in CAPIF.

Type: GET

URI: /pre-provisioning/v1/apiInvokers/<invokerName>

Initiated by: Operator or Administrator

Table 2-13 Supported Response Codes

Code	Description
200	Operation completed successfully
404	Not found on unknown apiInvoker name

Example:

The following example shows how API Invoker provisioning details are fetched by submitting a GET request on the REST resource using cURL command.

cURL command

```
curl --http2-prior-knowledge --location --request GET '<IP Address: Port>/pre-provisioning/v1/apiInvokers/<invokerName>' \
--header 'Content-Type: application/json' \ -v
```

2.1.3 API Invoker Management

NEF uses the 3GPP defined `API_Invoker_Management` REST API to manage the API Invoker onboarding and offboarding requests coming to CAPIF.

2.1.3.1 Onboard API Invoker

This service operation is used by an API invoker to onboard itself as a recognized user of CAPIF.

Type: POST

URI: `/api-invoker-management/v1/onboardedInvokers`

Initiated By: API Invoker

Note

The API invoker's onboardingSecret is provided by CAPIF during [invoker pre-provisioning](#).

Table 2-14 Request/Response Body Parameters

Field Name	Data Type	Description
apiInvokerInformation	string	Generic information related to the API invoker, such as details of the device or the application. This is a mandatory parameter.
apiList	array	A list of APIs. When included by the API invoker in the HTTP request message, it lists the APIs that the API invoker intends to invoke while onboarding. When included by the CAPIF in the HTTP response message, it lists the APIs that the API invoker is allowed to invoke while onboarding. This is an optional parameter. For more information, refer to <i>Definition of type APIList</i> and <i>Definition of type ServiceAPIDescription</i> tables in 3GPP TS 29.222 .
notificationDestination	string	Notify API invoker about onboarding result. This is a mandatory parameter.
onboardingInformation	object	Onboarding information about the API invoker is necessary for the CAPIF to onboard the API invoker. This is a mandatory parameter.
apiInvokerPublicKey	string	Public Key of API Invoker. This is a mandatory parameter.
apiInvokerCertificate	string	API invoker's generic client certificate. The subject field in the certificate shall be encoded with API invoker ID as Common Name. This is a mandatory parameter.
onboardingSecret	string	API invoker's onboarding secret, provided by CAPIF during invoker pre-provisioning . This is a mandatory parameter.

Table 2-14 (Cont.) Request/Response Body Parameters

Field Name	Data Type	Description
requestTestNotification	boolean	Set to true by Subscriber to request the CAPIF to send a test notification. Set to false or omitted otherwise. This is a mandatory parameter.
supportedFeatures	string	Used to negotiate the supported optional features of the API. This attribute is provided in the HTTP POST request and in the response of successful resource creation. This is an optional parameter.
websocketNotifConfig	object	Configuration parameters to set up notification delivery over Websocket protocol. This is an optional parameter.
websocketNotifConfig.requestWebsocketUri	boolean	Set to true by Subscriber to request the Websocket URI. This is an optional parameter.
websocketNotifConfig.websocketUri	string	The Websocket URI. This is an optional parameter.

Supported Response Header**Table 2-15 Response Header**

Field Name	Data Type	Description
Location	string	Contains the URI of the newly created resource, according to the structure: /api-invoker-management/v1/onboardedInvokers/{onboardingId}

Table 2-16 Supported Response Codes

Code	Description
201	API invoker onboarded successfully
400	Bad Request
401	Unauthorized
403	Request is forbidden.

Examples

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

cURL Command

```
curl POST --http2-prior-knowledge -v <IP Address:port>/api-invoker-  
management/v1/onboardedInvokers -H 'accept: application/json' --header  
'Content-Type: application/json' -d
```

Example of the Request Body

The following snippet shows an example of the request body.

```
{
  "apiInvokerInformation": "Api provider1",
  "apiList": [

  ],
  "notificationDestination": "http://10.75.226.101:32128/notificationdest",
  "onboardingInformation": {
    "apiInvokerCertificate": "",
    "apiInvokerPublicKey": "",
    "onboardingSecret":
      "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpbnZva2VyTmFtZSI6ImIudm9rZXItchJvZmlsZSIsImIudm9rZXJJZCI6IjVjZmEyZWZhLWQ3NDAtNDQwOC05YTY5LTE4ZDkxOGJmNTg3NSIsInN1YiI6IkFwaUludm9rZXJQcmVQcm92aXNpb25pbmciLCJpYXQiOiE1Nzc2NDA1MTIsImV4cCI6MTY0MDUzOTcxMn0.PJ38ipnBSYhFr5FyKNBzmxzUlgWBXeE9RpRZAnQ3fS7FVetynUK6snRdK_mc_FQI3rvfjPru00MeUt45Rqx9K3XxK-jHld_66RvvM5VQ47Tzis02s37yvc9XwiZbS0qX8B2292f2FgZ62G1ZPaeAIYzURQ8LcejoWDspgsn0zILs3c0zqEkrFkfbtWRQmivS3iPGBNyz50AMZuF6t7_9SgCjitb9GGNAxMxdjmaFv_GZ-YvCnuB--1XHt0Wk9e_i_bxoRV9FVADAeenE5b_LX50VkdKt0idSl8r-sY8ZZCqBoskrz7_053TFnFjrP9x6BMOUUV0X7bjWA8ZFdSwZ-A"
  },
  "requestTestNotification": true,
  "supportedFeatures": "af01",
  "websocketNotifConfig": {
    "requestWebSocketUri": true,
    "websocketUri": ""
  }
}
```

Example of the Response Body

The following example shows the contents of the response body in JSON format.

```
{
  "apiInvokerId": "fadb403b-2d49-420c-a0d1-9c6cd7e3813d",
  "onboardingInformation": {
    "apiInvokerPublicKey": "",
    "apiInvokerCertificate":,

    "onboardingSecret": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpbnZva2VyTmFtZSI6ImFmMDAxIiwiaW52b2tldklkIjoiZmFkYjQwM2ItMmQ0S00MjBjLWUwZDEtOWM2Y2Q3ZTM4MTNkIiwic3ViIjoiQXBpSw52b2tldlByZVByb3Zpc2lvbmIuZyIsImIhdCI6MTYwNTEwNTExNTcxMiwiaXhwIjoxNjY4MTg3NzEyfQ.SUQPamhxmYgeQ8fmspgj7si4qI4spv1Wp00Ai0Q0tsln33NAKKCSRqNp18aFfkd7i-RhivGs_luqd0YedYwuv3HNfdcrpTpwj_ztuIZuV7AY6xo9o0KdL6K-JTaLAUkyuLwKXUFERTSsx1PkiqfMa4waiC0r62ZD-wSEXKY2dVvUv_i1yKaDEAEo3SYsYEodAm6nf_sOY_PyoFiZ0txqrwaTqDU_M_HQQ-exkxns7U_Efb0lvz5BaIKk1WcC70p3IP_ftFVdUIRwSZiV0ayN3zCUgn5w2cEYL4DKM6_j20Rk5kvcQw9dDAeYHfyKSfLxftoxFFxa1DEuXyjhk6yZg"
  },
  "notificationDestination": "http://10.75.226.101:32128/notificationdest",
  "requestTestNotification": true,
  "websocketNotifConfig":,
}
```

```

"apiList":[
  {
    "apiName":"3gpp-monitoring-event",
    "apiId":"ac454be7-d2a1-4fed-b958-f0e495d6982a",
    "aefProfiles":[
      {
        "aefId":"2fb37df9-d7ca-4e04-a7a7-dc2167b0f9b2",
        "versions":[
          {
            "expiry":"2022-01-23T04:56:07Z",
            "apiVersion":"apiVersion",
            "resources":[
              {
                "resourceName":"resourceName",
                "commType":"REQUEST_RESPONSE",
                "uri":"uri",
                "custOpName":"custOpName",
                "operations":[
                  "GET",
                  "POST",
                  "PUT",
                  "DELETE"
                ],
                "description":"description"
              }
            ],
            "custOperations":[
              {
                "commType":"REQUEST_RESPONSE",
                "custOpName":"custOpName",
                "operations":[
                  "GET",
                  "POST",
                  "PUT",
                  "DELETE"
                ],
                "description":"description"
              }
            ]
          }
        ],
        "protocol":"HTTP2",
        "dataFormat":"JSON",
        "securityMethods":[
          "OAUTH"
        ],
        "domainName":"http://ocnef-ingressgateway:80",
        "interfaceDescriptions":[
          {
            "ipv4Addr":"ipv4Addr",
            "ipv6Addr":null,
            "port":5248,
            "securityMethods":[
            ]
          }
        ],
      },
    ],
  },

```

```

        {
            "ipv4Addr":,
            "ipv6Addr":"ipv6Addr",
            "port":5248,
            "securityMethods":[
                ]
            }
        ]
    },
    "description":"description",
    "supportedFeatures":"AF",
    "shareableInfo":{
        "isShareable":true,
        "capifProvDoms":[
            "capifProvDoms",
            "capifProvDoms"
        ]
    },
    "serviceAPICategory":"serviceAPICategory",
    "apiSuppFeats":,
    "pubApiPath":{
        "ccfIds":[
            "ccfIds",
            "ccfIds"
        ]
    },
    "ccfId":"ccfId"
}
],
"apiInvokerInformation":"Api provider1",
"supportedFeatures":
}

```

2.1.3.2 Offboard API Invoker

This service operation is used by an API invoker to offboard itself as a recognized user of NEF Exposure Gateway.

Request Type: DELETE

Request URI: /api-invoker-management/v1/onboardedInvokers/{onboardingId}

Request Initiated By: API Invoker

Table 2-17 Supported Response Codes

Code	Description
204	The individual API Invoker matching onboardingId was offboarded.
404	Not Found

Examples

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

cURL Command

```
curl --http2-prior-knowledge -X DELETE 'http://<IP Address:port>/apiRoot/api-  
invoker-management/v1/onboardedInvokers/<invoker id >' --header 'Content-  
Type: application/json' -v
```

2.1.4 NEF Security Management

CAPIF uses the 3GPP defined capif-security-management service to enable API invoker to communicate with the AF Management service to negotiate security methods and obtain authorizations to access the NEF.

The capif-security-management service facilitates the following operations:

2.1.4.1 Obtain Security Method

This service operation is used by API invoker to negotiate and obtain service API security method from the AF Manager service. The API invoker uses this information for authentication with NEF while sending service requests.

Type: PUT

URI: /capif-security/v1/trustedInvokers/{apiInvokerId}

Initiated By: API Invoker

Note

The apiInvokerId is obtained during API Invoker onboarding.

Table 2-18 Request/Response Body Parameters

Field Name	Data Type	Description
notificationDestination	string	The URI where the notification should be delivered This is a mandatory parameter.
requestTestNotification	boolean	Set to true by API invoker to request the CAPIF core function to send a test notification as defined in in subclause 7.6. Set to false or omitted otherwise. This is a mandatory parameter.

Table 2-18 (Cont.) Request/Response Body Parameters

Field Name	Data Type	Description
securityInfo	array	Represents the details of the security method for each service API interface. When included by the API invoker, it indicates the preferred method of security. When included by the CAPIF core function, it indicates the security method to be used for the service API interface. For details about security parameters, see Table 2-19 . This is a mandatory parameter.
supportedFeatures	string	Used to negotiate the supported optional features of the API. This attribute is provided in the HTTP POST request and in the response of successful resource creation. This is an optional parameter.
websocketNotifConfig	string	The configuration parameters to set up notification delivery over WebSocket protocol. This is an optional parameter.

Table 2-19 SecurityInformation Parameters

Field Name	Data Type	Description
interfaceDetails	object	Details of the interface. This is an optional parameter. For more information, refer to <i>Definition of type InterfaceDescription</i> table in 3GPP TS 29.222 .
aefId	string	The aef_id is obtained from the AefList onboarding response This is a mandatory parameter.
prefSecurityMethods	array	Represents the details of the security method for each service API interface. When included by the API invoker, it indicates the preferred method of security. When included by the OCCAPIF, it indicates the security method to be used for the service API interface. This is a mandatory parameter.
selSecurityMethod	string	This value is provided by the OCCAPIF. It indicates the selected security method for the API interface. If it is not provided, it means that no common security method is supported by the API invoker and the AEF, or the selected security method is not allowed by the local policy in OCCAPIF. This is an optional parameter.
authenticationInfo	string	The Authentication related information. This is an optional parameter.
authorizationInfo	string	The Authorization related information. This is an optional parameter.

Examples

The following example shows how an API Invoker obtains security method using PUT request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --request PUT <IP Address:port>/capif-  
security/v1/trustedInvokers/5cfa2efa-d740-4408-9a69-18d918bf5875 -H  
'accept: application/json' --header 'Content-Type: application/json' -d
```

Example of the Request Body

The following snippet shows an example of the request body.

```
{  
  "notificationDestination": "",  
  "requestTestNotification": "true",  
  "securityInfo": [  
    {  
      "aefId": "2fb37df9-d7ca-4e04-a7a7-dc2167b0f9b2",  
      "authenticationInfo": "",  
      "authorizationInfo": "",  
      "interfaceDetails": {  
        "ipv4Addr": "",  
        "ipv6Addr": "",  
        "port": 0,  
        "securityMethods": [  
          "OAUTH"  
        ]  
      },  
      "prefSecurityMethods": [  
        "OAUTH"  
      ],  
      "selSecurityMethod": "OAUTH"  
    }  
  ],  
  "supportedFeatures": "af",  
  "websocketNotifConfig": {  
    "requestWebSocketUri": true,  
    "websocketUri": ""  
  }  
}
```

Example of the Response Body

The following example shows the contents of the response body in JSON format.

```
{  
  "notificationDestination": "",  
  "requestTestNotification": "true",  
  "securityInfo": [  
    {  
      "aefId": "6106b1a4-cda8-4da0-b6d4-c4e40d6f2f0f",  
      "authenticationInfo": "",  
      "authorizationInfo": "",  
      "interfaceDetails": {  
        "ipv4Addr": "",  
        "ipv6Addr": "",  
        "port": 0,  
        "securityMethods": [  
          "OAUTH"  
        ]  
      },  
      "prefSecurityMethods": [  
        "OAUTH"  
      ],  
      "selSecurityMethod": "OAUTH"  
    }  
  ],  
  "supportedFeatures": "af",  
  "websocketNotifConfig": {  
    "requestWebSocketUri": true,  
    "websocketUri": ""  
  }  
}
```

```
    "authorizationInfo": "",
    "interfaceDetails": {
      "ipv4Addr": "",
      "ipv6Addr": "",
      "port": 0,
      "securityMethods": [
        "OAUTH"
      ]
    },
    "prefSecurityMethods": [
      "OAUTH"
    ],
    "selSecurityMethod": "OAUTH"
  }
},
"supportedFeatures": "af",
"websocketNotifConfig": {
  "requestWebsocketUri": true,
  "websocketUri": ""
}
}
```

For more information about capif-security configuration using REST API, see the "CAPIF_Security_API" section of [3GPP Technical Specification 29.222, Release 16](#).

2.1.4.2 Delete Security Method

This service operation is used by an API invoker to delete the service API security method obtained from the AF Manager.

Request Type: DELETE

Request URI: /capif-security/v1/trustedInvokers/

Request Initiated By: API Invoker

Table 2-20 Supported Response Codes

Code	Description
204	Deleted Successfully
404	Not Found

Examples

The following example shows how a service API security method is deleted for API Invoker by submitting a DELETE request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --request DELETE --http2-prior-knowledge -v
'http://<IP Address:port>/apiRoot/capif-security/v1/trustedInvokers/
<invokerid>' -H 'accept: application/json' --header 'Content-Type:
application/json'
```

2.1.4.3 Obtain Authorization

This service operation is used by an API invoker to obtain authorization to access NEF service APIs.

Type: POST

URI: /capif-security/v1/securities/{invokerId}/token

Initiated By: API Invoker

Note

The following data structure is not treated as a JSON object. It is a key, value pair data structure to be encoded using x-www-urlencoded format as specified in subclause 17.13.4.1 of W3C HTML 4.01 Specification [22].

Table 2-21 Request Body Parameters

Field Name	Data Type	Description
grant_type	string	This IE shall contain the grant type as "client_credentials". This is a mandatory parameter.
client_id	string	the API invoker Identifier. This is a mandatory parameter.
client_secret	string	This IE when present shall contain the onboarding secret which is got during API invoker onboarding. This is an optional parameter.
scope	string	This IE when present shall contain a list of AEF identifiers and its associated API names for which the access_token is authorized for use. It takes the format of 3gpp#aefld1:apiName1,apiName2,... apiNameX;aefld2:apiName1,apiName2,... apiNameY;...aefldN:apiName1,apiName2,... apiNameZ Using delimiter "#" after the discriminator "3gpp", ":" after AEF identifier, "," between API names and ";" between the last API name of the previous AEF identifier and the next AEF identifier. (NOTE 2) Example: '3gpp#aef-jiangsu-nanjing:3gpp-monitoring-event,3gpp-as-session-with-qos;aef-zhejiang-hangzhou:3gpp-cp-parameter-provisioning,3gpp-pfd-management' This is an optional parameter. Note: Currently, this parameter is not supported in request.

Table 2-22 Response Body Parameters

Field Name	Data Type	Description
access_token	string	This IE shall contain JWS Compact Serialized representation of the JWS signed JSON object containing AccessTokenClaims. This is a mandatory parameter.
token_type	string	This IE shall contain the token type (i.e. "Bearer"). This is a mandatory parameter.
expires_in	DurationSec	This IE when present shall contain the number of seconds after which the access_token is considered to be expired. This is an optional parameter.
scope	string	This IE when present shall contain a list of AEF identifiers and its associated API names for which the access_token is authorized for use. It takes the format of 3gpp#aefld1:apiName1,apiName2,... apiNameX;aefld2:apiName1,apiName2,... apiNameY;...aefldN:apiName1,apiName2,... apiNameZ Using delimiter "#" after the discriminator "3gpp", ":" after AEF identifier, "," between API names and ";" between the last API name of the previous AEF identifier and the next AEF identifier. (NOTE 2) Example: '3gpp#aef-jiangsu-nanjing:3gpp-monitoring-event,3gpp-as-session-with-qos;aef-zhejiang-hangzhou:3gpp-cp-parameter-provisioning,3gpp-pfd-management' This is an optional parameter.

Table 2-23 Supported Response Codes

Code	Description
200	Successful Access Token Request
400	Error in the Access Token Request

Examples

The following example shows how an API Invoker obtains authorization to access NEF service by submitting a POST request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge -k -X POST \<IP Address:Port>/capif-security/v1/
securities/5cfa2efa-d740-4408-9a69-18d918bf5875/token \ --header 'Content-
Type: application/x-www-form-urlencoded' \ -H 'accept: application/json' \--
data-urlencode 'client_id=85108099-3ecb-447c-8961-1407f519a958' \--data-
urlencode 'grant_type=client_credentials'
```

Example of the Response Body

The following snippet shows an example of the response body.

```
{
  "access_token": "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpc3MiOiJkd0M1ZWVhMy1hNWFhLTQ5ZWYtYjhhOC05YmYyNWYyODY4YmMiLCJzY29wZSI6IjNncHAjODVhMWMzZTQtNDcwYS00MTEzLWwEWMGYtZTl1MjgxYmU4MjZmOk1FIiwiaXhwaWJjNjQwNTM5NzEyfQ.HXNiucwTTnqpnIeqsrm4unF5v_yxnVt0uur63HHbAY9md_7ExbvdaFGuLN8AbeV10TBo7If80KaZyYHivc06PfeXXBGHzyolfMPP8YaVSPckVZT7abxVwYd-azCH53pZep74ilJaUvTWrs0AFhQHdrmkS_kC1_2kQNAAGdeNepGUlLE1BUXgcqehuVlNm7s0er1lerVNUMbQYyd_R-aERzvKnsqY62ukUp4n0Cb5TsABVDAFz9xzJH0ULaYlptCTWM3ZTrwEgnC-KjTF2xBTvJxc_U5p-IeP2saVbBPAQEH_pBJwfdAK1vN1UEXUfZPaK7vk9sE0IETbUwUDQjsykw",
  "token_type": "Bearer",
  "expires_in": 6496991,
  "scope": "3gpp#85a1c3e4-470a-4113-a00f-e9e281be826f:3gpp-monitoring-event"
}
```

For more information about capif-security configuration using REST API, see the "CAPIF_Security_API" section of [3GPP Technical Specification 29.222, Release 16](#).

2.1.5 CAPIF Event Management

The Event Management resource enables AFs and the NEF services to communicate with the NEF Event Manager service through the capif-events API. This API allows them to subscribe, unsubscribe, and receive notifications for the events managed by CAPIF.

The 3GPP defined Event Management API allows the subscribing entities, such as API invokers and API Exposing Functions (AEFs) to subscribe to and unsubscribe from the EG events. It also allows EG to send event notifications to the subscribing entities when the corresponding event is detected.

2.1.5.1 Create CAPIF Event Subscription

This service operation is used to create a new event subscription resource for CAPIF event notifications.

Type: POST

URI: /capif-events/v1/{subscriberId}/subscriptions

Initiated By: AF/OCNEF Microservice

Table 2-24 Request Body Parameters

Field Name	Data Type	Description
events	string	Describes the event for which subscription is required. This is a mandatory parameter.
eventFilters	string	The filtering criteria that can be applied when the event occurs. Only the events that meet the specified filtering criteria gets notified to consumers. This is an optional parameter.

Table 2-24 (Cont.) Request Body Parameters

Field Name	Data Type	Description
eventFilters.apilds	string	The unique identifier of the subscriber API. This is an optional parameter. Note: This parameter is applicable if the value for the <code>eventS</code> parameter contains any of the following: • <code>SERVICE_API_AVAILABLE</code> • <code>SERVICE_API_UNAVAILABLE</code> • <code>SERVICE_API_INVOCATION_SUCCESS</code> • <code>SERVICE_API_INVOCATION_FAILURE</code>
eventFilters.apiInvokerIds	string	The unique identifier of the subscriber API Invoker. This is an optional parameter. Note: This parameter is applicable if the value for the <code>eventS</code> parameter contains any of the following: • <code>API_INVOKER_ONBOARDED</code> • <code>API_INVOKER_OFFBOARDED</code> • <code>SERVICE_API_INVOCATION_SUCCESS</code> • <code>SERVICE_API_INVOCATION_FAILURE</code> • <code>API_INVOKER_AUTHORIZATION_REVOKED</code>
eventFilters.aeflds	string	The unique identifier of the subscriber AEF. This is an optional parameter. Note: This parameter is applicable if the value for the <code>eventS</code> parameter contains any of the following: • <code>SERVICE_API_INVOCATION_SUCCESS</code> • <code>SERVICE_API_INVOCATION_FAILURE</code>
eventReq	integer	Describes the reporting requirements of the event subscription.
eventReq.immRep	boolean	Indicates if the immediate reporting indication must be sent. Possible values are: • <code>true</code> • <code>false</code>
eventReq.notifMethod	string	The event notification method. Possible values are: • <code>PERIODIC</code> • <code>ONE_TIME</code> • <code>ON_EVENT_DETECTION</code>
eventReq.maxReportNbr	integer	Maximum Number of reports to be generated
eventReq.monDur	string	The event monitoring duration
notificationDestination	string	The notification destination path.
requestTestNotification	boolean	Indicates if the test notification must be sent. Possible values are: • <code>true</code> • <code>false</code>

Note

The data types supported by NEF complies with the 3GPP specifications. For more information about the 3GPP data types, see [3GPP Technical Specification 29.571, Release 16, Common Data Types for Service Based Interfaces](#).

Table 2-25 Supported Response Codes

Code	Description
201	The subscription resource is created successfully. The URI of the created resource shall be returned in the Location HTTP header.
204	No Content (successful notification)
400	Bad Request
401	Unauthorized
403	The subscription resource is not allowed to be created since one or more of the received parameters are out of the range defined by operator policies.
404	Not Found

For more details about error codes and their resolutions, see "HTTP Error Codes" in *Oracle Communications Cloud Native Core, Network Exposure Function Troubleshooting Guide*.

Examples

The following example shows how an AF or an NEF microservice creates a subscription request for receiving CAPIF event notifications by submitting a POST request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request POST 'http://<IP
address:Port>/capif-events/v1/SUBSCRIBER_ID_1/subscriptions' --header
'Content-Type: application/json' \
```

Example of the Request Body for a Service Availability Event Subscription

The following snippet shows an example of the request body.

```
"events": [
  "SERVICE_API_UNAVAILABLE",
  "SERVICE_API_AVAILABLE"
],
"eventFilters": [
  {
    "apiIds": [
      "a1621fc5-b4d1-4473-bbb5-abf3a1dd98ea",
      "API_ID_2"
    ],
    "apiInvokerIds": [
      "API_INVOKER_ID_1",
      "API_INVOKER_ID_2"
    ]
  }
],
```

```

    "aefIds": [
      "f56ad84b-ea6f-4c59-8d5a-011d899dbfd1",
      "AEF_ID_2"
    ]
  },
  "eventReq": { "immRep": false, "notifMethod": "ON_EVENT_DETECTION",
    "maxReportNbr": 0, "monDur": "2022-12-30T06:00:00Z"},
  "notificationDestination": "http://localhost:8080",
  "requestTestNotification": false
}'

```

Example of the Response Body

```

"events": [
  "SERVICE_API_UNAVAILABLE",
  "SERVICE_API_AVAILABLE"
],
"eventFilters": [
  {
    "apiIds": [
      "a1621fc5-b4d1-4473-bbb5-abf3a1dd98ea",
      "API_ID_2"
    ],
    "apiInvokerIds": [
      "API_INVOKER_ID_1",
      "API_INVOKER_ID_2"
    ],
    "aefIds": [
      "f56ad84b-ea6f-4c59-8d5a-011d899dbfd1",
      "AEF_ID_2"
    ]
  }
],
"eventReq": { "immRep": false, "notifMethod": "ON_EVENT_DETECTION",
  "maxReportNbr": 0, "monDur": "2022-12-30T06:00:00Z"},
  "notificationDestination": "http://localhost:8080",
  "requestTestNotification": false
}'

```

2.1.5.2 Delete CAPIF Event Subscription

This service operation is used to unsubscribe from the CAPIF event subscription.

Type: DELETE

URI: /capif-events/v1/{subscriberId}/subscriptions/{subscriptionId}

Initiated By: AF/OCNEF Microservice

Table 2-26 Supported Response Codes

Code	Description
204	Deleted successfully

Table 2-26 (Cont.) Supported Response Codes

Code	Description
404	Not Found For more details about error codes and their resolutions, see "HTTP Error Codes" in <i>Oracle Communications Cloud Native Core, Network Exposure Function Troubleshooting Guide</i> .

Examples

The following example shows how an AF or an NEF microservice deletes a subscription for CAPIF event notifications by submitting a DELETE request on the REST resource using cURL.

cURL Command

```
curl --location --request DELETE '<IP Address:Port>/capif-events/v1/  
SUBSCRIBER_ID_1/subscriptions/1b47dfaf-6090-4973-a483-ad72e008ad91' -header  
'Content-Type: application/json' -v
```

2.1.5.3 Publish CAPIF Event

This service operation is used by CAPIF to publish the subscribed event notifications to AFs or the NEF services.

Type: POST

URI: /capif-events/v1/publish-event

Initiated By: OCNEF Exposure Gateway

Table 2-27 Request Body Parameters

Field Name	Data Type	Description
events	string	Describes the event for which subscription is required. This is a mandatory parameter.
producerId	string	This parameter describes the subscribed event filter. This is a mandatory parameter.
eventDetail.serviceAPIDescriptions.apiName	string	The name of the service API.
eventDetail.apiIds	string	The unique identifier of the subscriber API
eventDetail.apiInvokerIds	string	The unique identifier of the subscriber API Invoker.
eventDetail.aefIds	string	The unique identifier of the subscriber AEF.
eventDetail.accCtrlPolList.apiId	string	Indicates the topology hiding information for the API.

Note

The data types supported by NEF complies with the 3GPP specifications. For more information about the 3GPP data types, see [3GPP Technical Specification 29.571, Release 16, Common Data Types for Service Based Interfaces](#).

Table 2-28 Supported Response Codes

Code	Description
204	No Content (successful notification)
400	Bad Request
401	Unauthorized
404	Not Found

For more details about error codes and their resolutions, see "HTTP Error Codes" in *Oracle Communications Cloud Native Core, Network Exposure Function Troubleshooting Guide*.

Examples

The following example shows how CAPIF creates a publish request for sending event notifications to AF or NEF services by submitting a POST request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request POST 'http://<IP
address:Port>/capif-events/v1/publish-event' --header 'Content-Type:
application/json' \
```

Example of the Request Body for Location Monitoring through UDM

The following snippet shows an example of the request body.

```
{
  "events": "SERVICE_API_AVAILABLE",
  "producerId": "PRODUCER_1",
  "eventDetail": {
    "serviceAPIDescriptions": [
      {
        "apiName": "apiName"
      }
    ],
    "apiIds": [
      "a1621fc5-b4d1-4473-bbb5-abf3a1dd98eaZ"
    ],
    "apiInvokerIds": [
    ],
    "aefIds": [
    ],
    "accCtrlPolList": {
      "apiId": "string"
    }
  }
}
```

```
    }
  }
```

Example of the Response Body

204 No Content

2.1.6 Monitoring Event Subscription

The Monitoring Event (ME) resource allows API Invokers or Application Function (AFs) to read all of the active monitoring event subscriptions or create a new monitoring event subscription at NEF through the 3GPP-monitoring-event API.

The Monitoring Event API allows the AFs to subscribe to notifications about specific events in 3GPP networks. It also allows the NEF to report the event by sending notifications to the authorized users when the corresponding event is detected.

The 3GPP-monitoring-event service facilitates the following operations:

2.1.6.1 Create ME Subscription

This service operation is used to create a new subscription resource for monitoring event notifications.

Type: POST

URI: /3gpp-monitoring-event/v1/{afId}/subscriptions

Initiated By: AF

! Important

The afId in path param should be same as api_invoker_id in the [Onboard API Invoker Response](#).

Table 2-29 Request Body Parameters

Field Name	Data Type	Description
afId	string	A unique identifier of the API invoker. This is a mandatory parameter.
externalId	string	A parameter for UE identification. One of the externalId or msisdn parameters must be used for UE identification for monitoring type LOCATION_REPORTING. This is a mandatory parameter.
msisdn	string	
externalGroupId	string	Identifies a defined group of users. NEF supports the following types: • PDN_CONNECTIVITY_STATUS • UE_REACHABILITY • LOSS_OF_CONNECTIVITY One of the externalId or msisdn or externalGroupId parameters must be used for the above monitoring event types.

Table 2-29 (Cont.) Request Body Parameters

Field Name	Data Type	Description
monitoringType	string	The type of monitoring event. NEF supports the following types: • LOCATION_REPORTING • PDN_CONNECTIVITY_STATUS • UE_REACHABILITY • LOSS_OF_CONNECTIVITY This is a mandatory parameter.
maximumNumberOfReports	integer	Identifies the maximum number of event reports to be generated. This is an optional parameter. Note It is mandatory to provide one of the parameters value from <code>maximumNumberOfReports</code> or <code>monitorExpireTime</code>.

Table 2-29 (Cont.) Request Body Parameters

Field Name	Data Type	Description
monitorExpireTime	DateTime	When the request includes milliseconds, the response sent to the AF returns the monitorExpireTime without milliseconds. This is because the system converts OffsetDateTime to LocalDateTime when forwarding the request to the 5GC agent and UDM. In the following example, the value 800 is removed from the response. • Request: 2100-06-30T06:30:20.800Z • Response: 2100-06-30T06:30:20Z When the request includes zone information, the monitorExpireTime in the response returned to AF will have the time which is converted to server time zone. This conversion accounts for the zone offset and zone information. As a result, the monitorExpireTime in response may be different from the monitorExpireTime provided in the request. Sample request: 2025-09-19T06:30:00+08:15 This is an optional parameter. Note It is mandatory to provide the value of either maximumNumberOfReports or monitorExpireTime.

Table 2-29 (Cont.) Request Body Parameters

Field Name	Data Type	Description
locationType	string	Specifies the type of location. Possible values are: • CURRENT_LOCATION: The requests to be notified for the current location • LAST_KNOWN_LOCATION: The requests to be notified for the last known location • CURRENT_OR_LAST_KNOWN_LOCATION: The AF requests the current or last known location • INITIAL_LOCATION: The requests to be notified for the initial location Note The INITIAL_LOCATION is supported only for GMLC based location monitoring. This is mandatory parameter for UDM based Location Reporting requests and GMLC based immediate Location Reporting request.
maximumDetectionTime	string	The maximum duration for detecting connectivity loss. NEF supports only LOSS_OF_CONNECTIVITY.
reachabilityType	string	If "monitoringType" is "UE_REACHABILITY", this parameter identifies whether the request is for "Reachability for Data". This is applicable for Ue-reachability_notification.
maximumLatency	integer	If "monitoringType" is "UE_REACHABILITY", this parameter may be included to identify the maximum delay acceptable for downlink data transfers. This is applicable for Ue-reachability_notification.
maximumResponseTime	string	If "monitoringType" is "UE_REACHABILITY", this parameter can be considered to identify the duration for which the UE remains reachable to allow the SCS or AS to deliver the required downlink data. This is applicable for Ue-reachability_notification.
suggestedNumberOfDIPackets	integer	If "monitoringType" is "UE_REACHABILITY", this parameter can be considered to identify the number of packets that the serving gateway shall buffer in case that the UE is not reachable. This is applicable for Ue-reachability-notification.

Table 2-29 (Cont.) Request Body Parameters

Field Name	Data Type	Description
idleStatusIndication	boolean	If "monitoringType" is set to "UE_REACHABILITY", this parameter can be considered to indicate the notification of when a UE, for which PSM is enabled, transitions into idle mode. "true": indicate enabling of notification "false": indicate no need to notify Default: "false" This is applicable for Ue-reachability_notification.
locQoS		Indicates the level of the required precision for the location information. QoS requirement for an immediate MT-LR or deferred MT-LR. If the locQoS attribute specifies that the cell level location accuracy is required, then NEF interacts with the appropriate GMLC deployed within the network by invoking the 3GPP defined Ngmlc_Location_ProvideLocation service operation. Note: This is a mandatory parameter for GMLC based location monitoring.
locQoS.hAccuracy		Horizontal positioning of the subscriber location
locQoS.VAccuracy		Vertical positioning of the subscriber location
locQoS.lcsQoSClass		Defines the degree of adherence by the location service to another quality of service parameter (Accuracy), if requested.
ldrType		This parameter is available only for the deferred requests for location monitoring. Possible values are: PERIODIC: The AF requests the location details of the subscriber periodically. ENTERING_INTRO_AREA: The AF requests the location details of the subscriber when the subscriber enters a specified area. The area details are specified in locationArea5G parameter. LEAVING_FROM_AREA: The AF requests the location details of the subscriber when the subscriber leaves a specified area. The area details are specified in locationArea5G parameter. BEING_INSIDE_AREA: The AF requests the location details of the subscriber when the subscriber is in a specified area. The area details are specified in locationArea5G parameter. MOTION: The AF requests the location details of the subscriber when the subscriber moves by more than some predefined straight line distance from a previous location. For more details on the parameters, see 3GPP 23273 Specification
locationArea5G		Indicates the area within which the AF requests the area event of the target subscriber.

Table 2-29 (Cont.) Request Body Parameters

Field Name	Data Type	Description
repPeriod		Indicates the time interval between successive reporting. It shall be present when the AF requests to report delays periodically.
minimumReportInterval		Defines the minimum time interval between each location reporting notification for periodic request. Note: This parameter is not included if the maximumNumberOfReports attribute is present and set to a one-time event.
maxRptExpireIntvl		Defines the maximum time interval between each location reporting notification.
samplingInterval		Defines the maximum time interval between consecutive evaluations by a subscriber for a trigger event.
reportingLocEstInd		Indicates if the event reporting should include location estimation information. The location estimation information is included in event reporting if the value is true. By default, the value is false.
accuracy	string	The supported value are: - CGI_ECGI: This value is applicable for UDM based location monitoring. This value raises requests to be notified about cell level location accuracy. - GEO_AREA: This value is applicable for GMLC based location monitoring. This value raises requests to be notified for the location accuracy based on the geographic area. This is an optional parameter.
notificationDestination	link	The destination URI where the notification for the subscription must be sent. This is a mandatory parameter.

Note

- An AF request should either contain LocationType or LdrType and be mutually exclusive. If both are present, then NEF would throw validation failure.
- The data types supported by NEF complies with the 3GPP specifications. For more information about the 3GPP data types, see [3GPP Technical Specification 29.571, Release 16, Common Data Types for Service Based Interfaces](#).

Table 2-30 Supported Response Codes

Code	Description
201	The subscription resource is created successfully. The URI of the created resource shall be returned in the Location HTTP header.

Table 2-30 (Cont.) Supported Response Codes

Code	Description
200	The subscription resource is created successfully. The operation is successfully, and corresponding monitoring event report is included.
403	The subscription resource is not allowed to be created since one or more of the received parameters are out of the range defined by operator policies.
	The subscription resource is not allowed to be created since the Idle Status Indication is received in the request but not supported by the network.
415	Unsupported Media Type
504	Gateway Error

Examples

The following example shows how an AF creates a subscription request for notification about monitoring event by submitting a POST request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request POST 'http://<IP
address:Port>/3gpp-monitoring-event/v1/afID1000/subscriptions' --header
'Content-Type: application/json' \ -H "accept: application/json -H
'authorization:
    Bearer
eyJ0eXAiOiJKV1QiLCJpbnZva2VyTmFtZSI6ImIudm9rZXItcHJvZmIsZSI6ImFsZyI6IjJTMjU2In
0.eyJpc3MiOiI3OWE1OWYyMi1iMzMzLTlRMTItYTWE3NS1hMTgxM2NkYWZtZTciLCJzY29wZSI6IjNn
cHAjZmYwYzYk1NjctMDBkZC00YTlRMTk5MmQzMmNhMDNlNTQyZWZRM0jNncHAtdW9uaXRvcmluZy1ldm
VudCIsImV4cCI6MTY2ODE4NzcwMn0.Ns0ARB8K7ahZYX1w-kt4x0aX-
0EJW45WjnijC9Ki8lSxZlQcrNjQnQZLUN9Pzf6Bvwbxf7-
yPw42_JD7k9I3ku54MuK1nNVIMAUD_mm_AIZ1jW9tJ9hSG39NA63eB_HtBw3J60dM9B1_pHELT-8g
GTt7MCQ_V96REZsjHJ5EN9zbxVmmsb5H4J4TJMwAA9dQh_L-
gFiMX9m4dZ08PvhtYE_2Q_ptWaTsZUNj00yn6FKB6G0EYbIvqA3Wykufolt400p6pa7FJbpfvTBXBK
GQ1Czv33kB_Vj1tqswpxCKjva_mA20o0AZmw2Y6p9_2QouIC0LFrSI_HIArtjF7AUg'
```

Note

The authorization bearer value is obtained from the `access_token` parameter in the response body of the Obtain Authorization API. For more information about the parameter, see [Obtain Authorization](#).

Example of the Request Body for Location Monitoring through UDM

The following snippet shows an example of the request body.

```
{
  "externalId": "9590523908@Oracle",
  "monitoringType": "LOCATION_REPORTING",
  "locationType": "CURRENT_LOCATION",
```

```

    "maximumNumberOfReports": 100,
    "monitorExpireTime": "2022-08-19T06:40:00Z",
    "accuracy": "CGI_ECGI",
    "notificationDestination": "http://af-svc:80/notifications"
  }

```

Example of the Response Body

```

{
  "self": "http://ocnef-ingressgateway:80/3gpp-monitoring-event/v1/afID1000/subscriptions/e3253426-bca9-4c84-a7f7-2dc4b5e616bb",
  "externalId": "9590523908@Oracle",
  "monitoringType": "LOCATION_REPORTING",
  "locationType": "CURRENT_LOCATION",
  "maximumNumberOfReports": 100,
  "monitorExpireTime": "2022-08-19T06:40:00Z",
  "accuracy": "CGI_ECGI",
  "notificationDestination": "http://af-svc:80/notifications"
}

```

Note

The notificationDestination link is described as follows:

```
http://<AF Hostname>:<AF Port>/<any path supported by AF for notifications>
```

Example of the Request Body for GMLC Based Location Monitoring (Immediate Request)

The following snippet shows an example of the request body:

```

{
  "externalId": "immed5@Oracle.com",
  "monitoringType": "LOCATION_REPORTING",
  "maximumNumberOfReports": 1,
  "locationType": "CURRENT_LOCATION",
  "locQoS": {
    "hAccuracy": 11.0,
    "vAccuracy": 11.0,
    "lcsQoSClass": "ASSURED"
  },
  "notificationDestination":
    "http://af_host:af_port/af/notifications"
}

```

Example of the Response Body

```

{
  "externalId": "immed17@Oracle.com",

```

```

    "locationInfo": {
      "geographicArea": {
        "shape": "POINT",
        "point": {
          "lon": 27.2046,
          "lat": 77.4977
        }
      }
    },
    "monitoringType": "LOCATION_REPORTING"
  }
}

```

Example of the Request Body for GMLC Based Location Monitoring (Deferred Request)

- **ldrType: PERIODIC**

The following snippet shows an example of the request body for a deferred request where, **ldrType: PERIODIC**:

```

{
  "externalId": "Defe1@Oracle.com",
  "monitoringType": "LOCATION_REPORTING",
  "maximumNumberOfReports": 2,
  "monitorExpireTime": "2022-04-10T06:40:00Z",
  "locQoS": {
    "hAccuracy": 12.0,
    "vAccuracy": 13.0,
    "lcsQoSClass": "ASSURED"
  },
  "ldrType": "PERIODIC",
  "repPeriod": 3600,
  "notificationDestination":
    "http://af\_host:af\_port/af/notifications"
}

```

- **ldrType: MOTION**

The following snippet shows an example of the request body for a deferred request where, **ldrType: MOTION**:

```

{
  "externalId": "Defe1@Oracle.com",
  "monitoringType": "LOCATION_REPORTING",
  "maximumNumberOfReports": 2,
  "monitorExpireTime": "2022-04-10T06:40:00Z",
  "locQoS": {
    "hAccuracy": 12.0,
    "vAccuracy": 13.0,
    "lcsQoSClass": "ASSURED"
  },
  "ldrType": "MOTION",
  "minimumReportInterval": 3600,
  "maxRptExpireIntvl": 4800,
  "samplingInterval": 5000,
  "reportingLocEstInd": "true",
  "linearDistance": "20",
  "notificationDestination":

```

```
    "http://af_host:af_port/af/notifications"
  }
```

- `ldrType`: ENTERING_INTO_AREA/LEAVING_FROM_AREA/BEING_INSIDE_AREA
The following snippet shows an example of the request body for a deferred request where, `ldrType`: ENTERING_INTO_AREA/LEAVING_FROM_AREA/BEING_INSIDE_AREA:

```
{
  "externalId": "Defe1@Oracle.com",
  "monitoringType": "LOCATION_REPORTING",
  "maximumNumberOfReports": 2,
  "monitorExpireTime": "2022-04-10T06:40:00Z",
  "locQoS": {
    "hAccuracy": 12.0,
    "vAccuracy": 13.0,
    "lcsQoSClass": "ASSURED"
  },
  "ldrType": "ENTERING_INTO_AREA",
  "locationArea5G": {
    "geographicAreas": [
      {
        "shape": "POINT",
        "point": {
          "lon": "20.0",
          "lat": "20.0"
        }
      }
    ]
  },
  "minimumReportInterval": 3600,
  "maxRptExpireIntvl": 4800,
  "samplingInterval": 5000,
  "reportingLocEstInd": "true",
  "notificationDestination":
    "http://af_host:af_port/af/notifications"
}
```

Example of the Response Body

```
{
  "self": "http://ocnef-ext-ingress-gateway:80/3gpp-monitoring-event/v1/invoker2/subscriptions/4be33db8-62b2-4845-b0f1-992ae24e14bd",
  "externalId": "switch@Oracle.com",
  "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/notifications",
  "monitoringType": "LOCATION_REPORTING",
  "maximumNumberOfReports": 3,
  "monitorExpireTime": "2023-04-25T06:40:00Z",
  "minimumReportInterval": 3600,
  "maxRptExpireIntvl": 4800,
  "samplingInterval": 5000,
  "reportingLocEstInd": true,
  "linearDistance": 20,
```

```

    "locQoS": {
      "hAccuracy": 12.0,
      "vAccuracy": 13.0,
      "lcsQoSClass": "ASSURED"
    },
    "ldrType": "MOTION",
    "monitoringEventReport": {
      "externalId": "switch@Oracle.com",
      "locationInfo": {
        "geographicArea": {
          "shape": "POINT",
          "point": {
            "lon": 27.2046,
            "lat": 77.4977
          }
        }
      },
      "ldrType": "MOTION"
    },
    "monitoringType": "LOCATION_REPORTING"
  }
}

```

Example of the Request Body for UE Reachability

```

{
  "externalId": "pdu35@Oracle.com",
  "monitoringType": "UE_REACHABILITY",
  "maximumNumberOfReports": 10,
  "monitorExpireTime": "2024-10-22T16:59:00Z",
  "reachabilityType": "DATA",
  "maximumLatency": 100,
  "maximumResponseTime": 1,
  "suggestedNumberOfDlPackets": 300,
  "idleStatusIndication": "true",
  "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/notifications"
}

```

Example of the Request Body for Loss of Connectivity

```

{
  "externalId": "loc21@Oracle.com",
  "monitoringType": "LOSS_OF_CONNECTIVITY",
  "maximumNumberOfReports": 10,
  "monitorExpireTime": "2024-10-22T16:59:00Z",
  "maximumDetectionTime": 10,
  "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/notifications"
}

```

Example of the Request Body for PDU Connectivity Status

```

{
  "externalId": "pdu@Oracle.com",
  "monitoringType": "PDN_CONNECTIVITY_STATUS",
  "maximumNumberOfReports": 10,

```

```
"monitorExpireTime": "2024-10-22T16:59:00Z",  
"notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/notifications"  
}
```

Note

The ConfigMapper API can be used to pre-configure DNN for a given AF ID for PDU Session related subscription using MonitoringEvent. For more information, see [ConfigMapper API](#).

2.1.6.2 Get ME Subscription

This service operation is used to fetch details for a specific monitoring event subscription from NEF.

Type: GET

URI: /3gpp-monitoring-event/v1/{afId}/subscriptions/{subscriptionId}

Initiated By: AF

Important

The afId in path param should be same as apiInvokerId in the [Onboard API Invoker Response](#).

Table 2-31 Request Body Parameters

Field Name	Data Type	Description
afId	string	A unique identifier of the API invoker. This is a mandatory parameter.
subscriptionId	string	The unique identifier of the ME subscription. This is a mandatory parameter.

Table 2-32 Response Body Parameters

Field Name	Data Type	Description
externalId	string	The unique identifier of the ME subscription. This is a mandatory parameter.
notificationDestination	string	The path supported by AF to receive the notifications. This is a mandatory parameter.
monitoringType	string	The type of monitoring event.
maximumNumberOfReports	integer	The maximum number of event reports to be generated for the subscription.

Table 2-32 (Cont.) Response Body Parameters

Field Name	Data Type	Description
locationType	string	Possible values are: - CURRENT_LOCATION: The requests to be notified for the current location - LAST_KNOWN_LOCATION: The requests to be notified for the last known location - CURRENT_OR_LAST_KNOWN_LOCATION: The AF requests the current or last known location
monitorExpireTime	integer	The duration (in days) after which the subscription expires. This is an optional parameter.
maximumDetectionTime	string	The maximum duration for detecting connectivity loss. NEF supports only LOSS_OF_CONNECTIVITY.
externalGroupId	string	Identifies a defined group of users. NEF supports the following types: - PDN_CONNECTIVITY_STATUS - UE_REACHABILITY - LOSS_OF_CONNECTIVITY
reachabilityType	string	If "monitoringType" is "UE_REACHABILITY", this parameter is considered to identify whether the request is for "Reachability for Data". This is applicable for Ue-reachability_notification.
maximumLatency	integer	If "monitoringType" is "UE_REACHABILITY", this parameter may be included to identify the maximum delay acceptable for downlink data transfers. This is applicable for Ue-reachability_notification.
maximumResponseTime	string	If "monitoringType" is "UE_REACHABILITY", this parameter can be considered to identify the length of time for which the UE remains reachable to allow the SCS or AS to deliver the required downlink data. This is applicable for Ue-reachability_notification.
suggestedNumberOfDIPackets	integer	If "monitoringType" is "UE_REACHABILITY", this parameter can be considered to identify the number of packets that the serving gateway shall buffer in case that the UE is not reachable. This is applicable for Ue-reachability-notification.
idleStatusIndication	boolean	If "monitoringType" is set to "UE_REACHABILITY" or "AVAILABILITY_AFTER_DDN_FAILURE", this parameter can be considered to indicate the notification of when a UE, for which PSM is enabled, transitions into idle mode. "true": indicate enabling of notification "false": indicate no need to notify Default: "false" This is applicable for Ue-reachability_notification, Availability_after_DDN_failure_notification, and Availability_after_DDN_failure_notification_enhancement.

Table 2-32 (Cont.) Response Body Parameters

Field Name	Data Type	Description
accuracy	string	Possible values are: - CGI_ECGI: The requests to be notified at cell level location accuracy - ENODEB: The requests to be notified at eNodeB level location accuracy - TA_RA: The requests to be notified at TA/RA level location accuracy - PLMN: The requests to be notified at PLMN level location accuracy - TWAN_ID: The requests to be notified at TWAN identifier level location accuracy - GEO_AREA: The requests to be notified of the geographical area accuracy.

Note

The data types supported by NEF comply with the 3GPP specifications. For more information about the 3GPP data types, see [3GPP Technical Specification 29.571, Release 16, Common Data Types for Service Based Interfaces](#).

Examples

The following example shows how an AF fetches the ME subscription details by submitting a GET request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request GET 'http://<IP
address:Port>/3gpp-monitoring-event/v1/afID1000/subscriptions/
8834804d-1338-477b-b43a-6d40a233bb3d HTTP/2' --header 'Content-Type:
application/json' \ -H 'authorization:
Bearer
eyJ0eXAiOiJKV1QiLCJpbnZva2VyTmFtZSI6ImIudm9rZXItCHJvZmlsZSI6ImFsZyI6IlJTMjU2In
0.eyJpc3MiOiI3OWE1OWYyMi1iMzMzLTJhMTItYWE3NS1hMTgxM2NkYWZlZTciLCJzY29wZSI6IjNn
cHAjZmYwYzlkNjctMDkzMDYTRmLTk5MmQtZmNhMDNlNTQyZWZmOjNncHAtbW9uaXRvcmluZy1ldm
VudCIsImV4cCI6MTY2ODE4NzcwMn0.Ns0ARB8K7ahZYX1w-kt4x0aX-
0EJW45WjnijC9Ki8lSxZlQcrNjQnQZLUN9Pzf6Bvwbxf7-
yPw42_JD7k9I3ku54MuK1nNVIMAUD_mm_AIZ1jW9tJ9hSG39NA63eB_HtBw3J60dM9B1_pHELT-8g
GTt7MCQ_V96REZsjHJ5EN9zbxVmmsb5H4J4TJMWdAA9dQh_L-
gFiMX9m4dZ08PvhtYE_2Q_ptWaTsZUNj00yn6FKB6G0EYbIvqA3Wykufolt400p6pa7FJbpfvTBXBK
GQ1Czv33kB_Vj1tqswpxCKjva_mA20o0AZmW2Y6p9_2QouIC0LFrSI_HiArtjf7Aug'
\
```

Note

The authorization bearer value is obtained from the access_token parameter in the response body of the Obtain Authorization API. For more information about the parameter, see [Obtain Authorization](#).

Example of the Response Body

```
{
  "self": "http://ocnef-sim-ingress-gateway:8080/3gpp-monitoring-
event/v1/AF-app-1/subscriptions/8834804d-1338-477b-b43a-6d40a233bb3d",
  "externalId": "demouser1@oracle.com",
  "notificationDestination": "http://10.75.226.251:30568/af/
notifications",
  "monitoringType": "LOCATION_REPORTING",
  "maximumNumberOfReports": 100,
  "locationType": "CURRENT_LOCATION",
  "accuracy": "CGI_ECGI"
}
```

Note

The notificationDestination link is described as follows:

http://<AF Hostname>:<AF Port>/<any path supported by AF for notifications>

2.1.6.3 Delete an ME Subscription

This service operation is used to delete an ME subscription.

Type: DELETE

URI: /3gpp-monitoring-event/subscriptions/{subscriptionId}

Initiated By: AF

Examples

The following example shows how an AF deletes the ME subscription by submitting a DELETE request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request DELETE 'http://<IP
address:Port>/3gpp-monitoring-event/v1/afID1000/subscriptions/
8834804d-1338-477b-b43a-6d40a233bb3d' \ --header 'Accept: application/json'
\ --header 'Authorization: Bearer
eyJ0eXAiOiJKV1QiLCJpbnZva2VyTmFtZSI6ImZvcnRpnkzIiwiaWYwbnIjoilMyNTYifQ.eyJpc3M
iOiJmZWUzMzZkNi04YzJlLTQxMTYtODFjYi1mZGJjYmNjZGZjZjMiLCJzY29wZSI6IjNncHAjYzY1O
DgX0TktN2RjNS00MTQ5LWJhNDAtMzFmNmY4MzFmThl0jNncHAtbW9uaXRvcmluZy1ldmVudCIsImV
4cCI6MTY2ODE4NzcxMn0.Kbx1UDWrbtu7xUYehowu_b2qlp-
MmWMQ9ixVrPLbqLrPGSy6IcPuVHKcmgIOzNfY-40EIUHG6xvQFGej1TVXooNCOKe-e9YYRVtyzmo-
aZN-
INlT3JyfaCw_zAY0pQg9ttJSBshEAgap4GyrylvymKlZpDAs9Q80IegBBbjCR7xkQDnJRK9qNqzkW
DbPrXUkU0nDMQoIXiDX6QRdXr3n1wVQ6BeLVMPrrR6a0z2JcqVr-INMS7BM0xP_-
XI42o2WASTFvfrwk_T1v08GpWyj1xQmLSTCcKouBlX6i-
NnFmpZR9LUYvIO4m53C05oXvA5JYGwctRzLcxpKynewvGHlQ' \
```

Note

The authorization bearer value is obtained from the `access_token` parameter in the response body of the Obtain Authorization API. For more information about the parameter, see [Obtain Authorization](#).

2.1.6.4 Update ME Subscription

Using this service operation, you can update an existing ME subscription.

Type: PUT

URI: {apiRoot}/3gpp-monitoring-event/v1/{scsAsId}/subscriptions/{subscriptionId}

Initiated By: AF

Note

- The afId (the scsAsId in URI) in the path parameter must be same as the `api_invoker_id` during [pre provisioning](#).
- GMLC interface does not support any PUT or PATCH functionality. If there is a request on a subscription created on GMLC, it is failed stating that update is not supported on GMLC.

Table 2-33 Request Body Parameters

Parameters	Data Type	Is Parameter Modifiable through PUT request?	Description
afId	string	No	A unique identifier of the API invoker. This is a mandatory parameter.
externalId	string	No	A parameter for UE identification. One of the externalId or msisdn parameters must be used for UE identification for monitoring type LOCATION_REPORTING.
msisdn	string	No	
externalGroupId	string	No	Identifies a defined group of users. NEF supports the following types: • PDN_CONNECTIVITY_STATUS • UE_REACHABILITY • LOSS_OF_CONNECTIVITY One of the externalId or msisdn or externalGroupId parameters must be used for the above monitoring event types.
monitoringType	string	No	The type of monitoring event. NEF supports the following types: • LOCATION_REPORTING • PDN_CONNECTIVITY_STATUS • UE_REACHABILITY • LOSS_OF_CONNECTIVITY This is a mandatory parameter.

Table 2-33 (Cont.) Request Body Parameters

Parameters	Data Type	Is Parameter Modifiable through PUT request?	Description
locationType	string	Yes	Possible values are: • CURRENT_LOCATION: The requests to be notified about the current location of the subscriber • LAST_KNOWN_LOCATION: The requests to be notified about the last known location of the subscriber • CURRENT_OR_LAST_KNOWN_LOCATION: The AF requests the current or last known location of the subscriber This is a mandatory parameter, if the monitoringType selected is LOCATION_REPORTING.
maximumNumberOfReports	integer	Yes	Identifies the maximum number of event reports to be generated. This is a mandatory parameter.
monitorExpireTime	string	Yes	When the request includes milliseconds, the response sent to the AF returns the monitorExpireTime without milliseconds. This is because the system converts OffsetDateTime to LocalDateTime when forwarding the request to the 5GC agent and UDM. In the following example, the value 800 is removed from the response. • Request: 2100-06-30T06:30:20.800Z • Response: 2100-06-30T06:30:20Z When the request includes zone information, the monitorExpireTime in the response returned to AF will have the time which is converted to server time zone. This conversion accounts for the zone offset and zone information. As a result, the monitorExpireTime in response may be different from the monitorExpireTime provided in the request. Sample request: 2025-09-19T06:30:00+08:15 This is an optional parameter. Note It is mandatory to provide the value of either maximumNumberOfReports or monitorExpireTime.
maximumDetectionTime	integer	Yes	The maximum duration for detecting connectivity loss. NEF supports only LOSS_OF_CONNECTIVITY.

Table 2-33 (Cont.) Request Body Parameters

Parameters	Data Type	Is Parameter Modifiable through PUT request?	Description
reachabilityType	string	No	If "monitoringType" is "UE_REACHABILITY", this parameter is considered to identify whether the request is for "Reachability for Data". This is applicable for Ue-reachability_notification.
maximumLatency	integer	Yes	If "monitoringType" is "UE_REACHABILITY", this parameter may be included to identify the maximum delay acceptable for downlink data transfers. This is applicable for Ue-reachability_notification.
maximumResponseTime	integer	Yes	If "monitoringType" is "UE_REACHABILITY", this parameter can be considered to identify the length of time for which the UE remains reachable to allow the SCS or AS to deliver the required downlink data. This is applicable for Ue-reachability_notification.
suggestedNumberOfDIPackets	integer	Yes	If "monitoringType" is "UE_REACHABILITY", this parameter can be considered to identify the number of packets that the serving gateway shall buffer in case that the UE is not reachable. This is applicable for Ue-reachability-notification.
groupReportGuardTime	integer	Yes	Identifies the report guard time. This is applicable for the following types: • PDN_CONNECTIVITY_STATUS • UE_REACHABILITY • LOSS_OF_CONNECTIVITY
idleStatusIndication	boolean	Yes	If "monitoringType" is set to "UE_REACHABILITY", this parameter can be considered to indicate the notification of when a UE, for which PSM is enabled, transitions into idle mode. • "true": indicate enabling of notification • "false": indicate no need to notify Default: "false" This is applicable for Ue-reachability_notification.
accuracy	string	No	The supported value is: CGI_ECGI - The requests to be notified at cell level location accuracy. This is an optional parameter.
notificationDestination	link	Yes	The URI where the notification for the subscription must be sent. This is a mandatory parameter.

Table 2-34 Supported Response Codes

Code	Description
200	The subscription resource is updated. The URI of the created resource is returned in the Location HTTP header.
403	The subscription resource is not created because one or more of the received parameters are out of the range defined by operator policies.

Table 2-34 (Cont.) Supported Response Codes

Code	Description
	The subscription resource is not created because the Idle Status Indication is received in the request but not supported by the network.
415	Unsupported Media Type

Examples

The following example shows how an AF creates a subscription request for notification about monitoring event by submitting a PUT request on the REST resource using cURL:

cURL Command

```
curl --http2-prior-knowledge --location --request PUT 'http://<IP
address:Port>/3gpp-monitoring-event/v1/afID1000/subscriptions/subscriptionID'
--header 'Content-Type: application/json' \
```

Example of the Request Body for Location Reporting

```
{
  "externalId": "9590523908@Oracle",
  "monitoringType": "LOCATION_REPORTING",
  "locationType": "CURRENT_LOCATION",
  "maximumNumberOfReports": 100,
  "monitorExpireTime": "2024-10-22T16:59:00Z",
  "accuracy": "CGI_ECGI",
  "notificationDestination": "http://af-svc:80/notifications"
}
```

Example of the Response Body for Location Reporting

```
{
  "self": "http://ocnef-ingressgateway:80/3gpp-monitoring-event/v1/afID1000/
subscriptions/e3253426-bca9-4c84-a7f7-2dc4b5e616bb",
  "externalId": "9590523908@Oracle",
  "monitoringType": "LOCATION_REPORTING",
  "locationType": "CURRENT_LOCATION",
  "maximumNumberOfReports": 100,
  "monitorExpireTime": "2024-10-22T16:59:00Z",
  "accuracy": "CGI_ECGI",
  "notificationDestination": "http://af-svc:80/notifications"
}
```

Note

The format of notificationDestination parameter is as follows:

`http://<AF Hostname>:<AF Port>/<any path supported by AF for notifications>`

Where, AF Hostname is the hostname of the AF Port is the port defined for the AF.

Example of the Request Body for UE Reachability

```
{
  "externalId": "pdu35@Oracle.com",
  "monitoringType": "UE_REACHABILITY",
  "maximumNumberOfReports": 10,
  "monitorExpireTime": "2024-10-22T16:59:00Z",
  "reachabilityType": "DATA",
  "maximumLatency": 100,
  "maximumResponseTime": 1,
  "suggestedNumberOfDlPackets": 300,
  "idleStatusIndication": "true",
  "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/
notifications"
}
```

Example of the Request Body for Loss of Connectivity

```
{
  "externalId": "loc21@Oracle.com",
  "monitoringType": "LOSS_OF_CONNECTIVITY",
  "maximumNumberOfReports": 10,
  "monitorExpireTime": "2024-10-22T16:59:00Z",
  "maximumDetectionTime": 10,
  "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/
notifications"
}
```

Example of the Request Body for PDU Connectivity Status

```
{
  "externalId": "pdu@Oracle.com",
  "monitoringType": "PDN_CONNECTIVITY_STATUS",
  "maximumNumberOfReports": 10,
  "monitorExpireTime": "2024-10-22T16:59:00Z",
  "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/
notifications"
}
```

Note

The ConfigMapper API can be used to pre-configure DNN for a given AF ID for PDU Session related subscription using MonitoringEvent. For more information, see [ConfigMapper API](#).

2.1.6.5 ME Notification

This service operation is used by an NF, such as Access Mobility Function (AMF) to notify about the monitoring events corresponding to subscriptions.

Type: POST

URI: /5gc-agent/v1/<translator>/<encoded-notification-receiving-service-url>

where, the <translator> and the <encoded-notification-receiving-service-url> are internal to NEF. NEF sends these values in the notification destination URL while subscribing to receive the notification.

Initiated By: AMF

Table 2-35 Request/Response Body Parameters

Field Name	Data Type	Description
notifyCorrelationId	string	The parameter provided by NEF to uniquely correlate the notifications with the subscription. This is a mandatory parameter.
reportList	array	See the ReportList Parameter . This is a mandatory parameter.

Table 2-36 ReportList Parameters

Field Name	Data Type	Description
type	string	Describes the type of the event which triggers the report. This is a mandatory parameter.
gpsi	string	This is a mandatory parameter for notifications for externalgroupid subscription.
refId	integer	
lossOfConnectReason	integer	If "monitoringType" is "LOSS_OF_CONNECTIVITY", this parameter shall be included if available to identify the reason why loss of connectivity is reported. This is applicable for Loss_of_connectivity_notification
maxAvailabilityTime	integer	
reachability	string	

Table 2-36 (Cont.) ReportList Parameters

Field Name	Data Type	Description
idleStatusIndication	boolean	If "monitoringType" is set to "UE_REACHABILITY" or "AVAILABILITY_AFTER_DDN_FAILURE", this parameter may be included to indicate the notification of when a UE, for which PSM is enabled, transitions into idle mode. "true": indicate enabling of notification "false": indicate no need to notify Default: "false" This is applicable for Ue-reachability_notification, Availability_after_DDN_failure_notification, and Availability_after_DDN_failure_notification_enhancement
state	string	Describes the state of the event which triggered the report. This IE is set to "TRUE" when subscriptionId IE is present. This is a mandatory parameter.
timeStamp	string	This IE shall contain the time at which the event is generated. This is a mandatory parameter.
location	object	Represents the location information of the UE. The possible values are: - nrLocationnone - eutraLocation - n3gaLocation This is a mandatory parameter.

Table 2-37 Supported Response Codes

Code	Description
204	Created
404	Not Found
400	Bad Request The subscription resource is not allowed to be created since the Idle Status Indication is received in the request but not supported by the network.
503	Service Unavailable

Examples

The following example shows how the AMF notifies about monitoring event by submitting a POST request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request POST '<IP Address:Port>/5gc-agent/v1/ME/b2NuZWYtbW9uaXRvcmluZ2V2ZW50czo4MDgw' \
--header 'Content-Type: application/json'
```

Example of Request Body for Location Reporting

```
{
  "notifyCorrelationId": "8cf61446-3c28-43ad-bdb3-906031d5562c",
  "reportList": [{
    "type": "LOCATION_REPORT",
    "state": {
      "active": true
    },
    "timeStamp": "2021-10-21T12:22:54Z",
    "location": {
      "nrLocation": {
        "tai": {
          "plmnId": {
            "mnc": 314,
            "mcc": 11
          },
          "tac": "1000001"
        },
        "ncgi": {
          "plmnId": {
            "mnc": 314,
            "mcc": 11
          },
          "nrCellId": "225BD6007"
        },
        "ageOfLocationInformation": 3276,
        "ueLocationTimestamp": "2021-09-15T12:22:54Z"
      }
    }
  }]
}
```

Example of Response Body for Location Reporting

204 No Content

Example of Request Body for Loss of Connectivity

```
{
  "notifyCorrelationId": "1648eef6-d613-4fde-8c6d-bca551701f43",
  "reportList": [
    {
      "type": "LOSS_OF_CONNECTIVITY",
      "gpsi": "+919845123456",
      "refId": "1234",
      "lossOfConnectReason": "DEREGISTERED",
      "timeStamp": "2024-10-21T12:22:54Z",
      "state": {
        "active": "true",
        "remainReports": "5",
        "remainDuration": "5"
      }
    }
  ]
}
```

```
    }}
  }
```

Example of Request Body for UE Reachability

```
{
  "notifyCorrelationId": "14f59a2d-7369-4c02-90a7-073e170fc431",
  "reportList": [
    {
      "type": "REACHABILITY_REPORT",
      "gpsi": "+919845123456",
      "refId": "1234",
      "maxAvailabilityTime": "2024-10-21T12:22:54Z",
      "reachability": "REACHABLE",
      "timeStamp": "2023-10-21T12:02:00Z",
      "idleStatusIndication":
        {
          "timeStamp": "2024-10-21T12:22:54Z",
          "activeTime": "5",
          "subsRegTimer": "5",
          "edrxCycleLength": "5",
          "suggestedNumOfDlPackets": "5"
        },
      "state":
        {
          "active": "true",
          "remainReports": "5",
          "remainDuration": "5"
        }
    }
  ]
}
```

Example of Request Body for PDN Connectivity Status

```
{
  "notifId": "1bbc27c3-3928-409f-8d4b-0ff474aebd13",
  "eventNotifs": [
    {
      "event": "PDU_SES_EST"/"PDU_SES_REL",
      "timeStamp": "2024-10-22T16:59:00Z",
      "gpsi": "john.doe@example.com",
      "ipv4Addr": "10.120.19.10",
      "pduSessType": "IPV4",
      "dnn": "dnn"
    }
  ]
}
```

2.1.7 ConfigMapper API

ConfigMapper API is used to pre-configure DNN for a given AF ID for a PDU Session related subscription using Monitoring Events service.

When NEF forwards a call towards UDM for a PDU Session subscription, NEF reads the pre-configured DNN against the AF ID and sends it to UDM. If mapping is not available in NEF, then DNN is not passed to UDM. All the APIs are accessed through FiveGC ingress gateway.

① Note

- Currently, the ConfigMapper API is solely used to pre-configure the DNN for a given AF ID for PDU Session Subscription. For information ME Subscription, see [Monitoring Event Subscription](#).
- From NEF 24.1.0 release, ConfigMapper APIs are routed through Fivegcagent Ingress Gateway instead of External Ingress Gateway.

2.1.7.1 Update Configuration Mapping

Using this service operation, you can update a pre-configured DNN mapping for a given AF ID.

Type: PUT

URI: /nef/apd-manager/{apiVersion}/af/configmap/{identifier}

Initiated By: AF

Examples

The following example shows how an AF updates a pre-configured DNN mapping for a given AF ID by submitting a PUT request on the REST resource using cURL:

cURL Command

```
curl --http2-prior-knowledge --location --request PUT 'http://<IP
address:Port>/nef/apd-manager/v1/af/configmap/e7473247-
f880-4255-952baa2b8c048369' --header 'Content-Type: application/json' \ --
data-raw '
{
  "id": "e7473247-f880-4255-952b-aa2b8c048369",
  "config":
  {
    "dnn": "dnn"
  }
} ' -v
```

Example of the Request Body

```
{
  "id": "e7473247-f880-4255-952b-aa2b8c048369",
  "config":
  {
    "dnn": "dnn"
  }
}
```

Example of the Response Body

```
{
  "id": "e7473247-f880-4255-952b-aa2b8c048369",
  "config":
  {
    "dnn": "dnn1"
  }
}
```

HTTP Status Codes - Response

200, 400, 404

2.1.7.2 GET Configuration Mapping

Using this service operation, you can fetch a pre-configured DNN mapping for a given AF ID.

Type: GET

URI: /nef/apd-manager/{apiVersion}/af/configmap/{identifier}

Initiated By: AF

Examples

The following example shows how an AF fetches a pre-configured DNN mapping for a given AF ID by submitting a GET request on the REST resource using cURL:

cURL Command

```
curl --http2-prior-knowledge --location --request GET 'http://<IP
address:Port>/nef/apd-manager/v1/af/configmap/e7473247-f880-4255-952b-
aa2b8c048369' \--header 'Content-Type: application/json' \
-v
```

Example of the Response Body

```
{
  "id": "e7473247-f880-4255-952b-aa2b8c048369",
  "config":
  {
    "dnn": "dnn1"
  }
}
```

HTTP Status Codes - Response

200, 404

2.1.7.3 GET All Pre-configured Mapping

Using this service operation, you can fetch mapping of all pre-configured DNNs and their respective AF IDs.

Type: GET

URI: /nef/apd-manager/{apiVersion}/af/configmap/

Initiated By: AF

Examples

The following example shows how an AF fetches mapping of all pre-configured DNNs and their respective AF IDs by submitting a GET request on the REST resource using cURL:

cURL Command

```
curl --http2-prior-knowledge --location --request GET 'http://<IP  
address:Port>/nef/apd-manager/v1/af/configmap/' --header 'Content-Type:  
application/json' \  
-v
```

Example of the Response Body

```
[  
  {  
    "id": "e7473247-f880-4255-952b-aa2b8c048369",  
    "config":  
    {  
      "dnn": "dnn1"  
    }  
  },  
  {  
    "id": "eaa10638-d178-4583-adf7-ba72875aee31",  
    "config":  
    {  
      "dnn": "dnn"  
    }  
  }  
]
```

HTTP Status Codes - Response

200

2.1.7.4 Delete Configuration Mapping

Using this service operation, you can delete a pre-configured DNN mapping for a given AF ID.

Type: DELETE

URI: /nef/apd-manager/{apiVersion}/af/configmap/{identifier}

Initiated By: AF

Examples

The following example shows how an AF deletes a pre-configured DNN mapping for a given AF ID by submitting a DELETE request on the REST resource using cURL:

cURL Command

```
curl --http2-prior-knowledge --location --request DELETE 'http://<IP  
address:Port>/nef/apd-manager/v1/af/configmap/e7473247-f880-4255-952b-  
aa2b8c048369' \--header 'Content-Type: application/json' \  
-v
```

HTTP Status Codes - Response

204, 404

2.1.8 AF Session with QoS

The AF Session with QoS resource allows API Invokers or Application Function (AFs) to set up a session with NEF with required QoS based on the application and service requirement through the 3gpp-as-session-with-qos API.

The 3gpp-as-session-with-qos API allows the AFs to subscribe to notifications about QoS events in 3GPP networks.

NEF supports the following event subscription:

- QOS_MONITORING
- QOS_NOTIF
- USAGE_REPORT
- FAILED_RESOURCES_ALLOCATION
- SUCCESSFUL_RESOURCES_ALLOCATION

NEF does implicit subscription for SUCCESSFUL_RESOURCES_ALLOCATION and FAILED_RESOURCES_ALLOCATION events.

NEF also allows the NEF to report the event by sending notifications to the authorized users when the corresponding event is detected.

The 3gpp-as-session-with-qos service facilitates the following operations:

2.1.8.1 Create AF Session with QoS Subscription

This service operation is used to create a new subscription resource for QoS provisioning and notifications.

Type: POST

URI: /3gpp-as-session-with-qos/v1/{afId}/subscriptions

Initiated By: AF

! Important

The afId in path param should be same as apiInvokerId in the [Onboard API Invoker Response](#).

Table 2-38 Request Body Parameters

Field Name	Data Type	Description
self	string	Specifies the referenced resource provided by NEF after subscription creation.
supportedFeatures	string	List the features supported by NEF. Note: OCNEF supported value is "7C" and it corresponds to support for features 'EthAsSessionQoS_5G', 'MacAddressRange_5G', 'AlternativeQoS_5G', 'QoSMonitoring_5G' and 'DisableUENotification_5G'.
notificationDestination	link	The destination URI where the notification for the subscription must be sent. This is a mandatory parameter.
flowInfo	array	Represents flow information. This is a mandatory parameter.
flowInfo.flowId	integer	Identifies the IP flows that is sent during event subscription
flowInfo.flowDescriptions	string	Indicates the packet filters of the IP flow
ethFlowInfo	array	The Ethernet data flow.
ethFlowInfo.EthFlowDescription	object	Defines a packet filter for an Ethernet flow.
qosReference	string	The predefined QoS information.
altQoSReferences	array	Represent that the subscription request is for the QOS_NOTIF event. This field must be included when the AF needs to subscribe for the QOS_NOTIF event. When this value is available, NEF subscribes for the QOS_NOTIF event in the Npcf_PolicyAuthorization request. For more information on this request, see <i>Oracle Communications Cloud Native Core, Policy User Guide</i> . This is an optional parameter.
disUeNotif	boolean	Specifies if QoS flow parameter signalling to the UE should be disabled when the SMF is notified by the NG-RAN of changes in the fulfilled QoS situation.
uelpv4Addr	string	The IPv4 address of the UE.
uelpv6Addr	string	The IPv6 address of the UE.

Table 2-38 (Cont.) Request Body Parameters

Field Name	Data Type	Description
macAddr	string	The Media Access Control (MAC) address of the UE.
ipDomain	string	The IPv4 domain identifier. This field is applicable only if ueIpv4Addr is available.
usageThreshold	object	Represent that the subscription request for the USAGE_REPORT event. This field must be included when the AF needs to subscribe for the USAGE_REPORT event. When this value is available, NEF subscribes for the USAGE_REPORT event in the Npcf_PolicyAuthorization request.
usageThreshold.duration	integer	Indicates the duration in seconds after which the usage threshold is considered to be reached.
usageThreshold.totalVolume	integer	Indicates the total data octets for both downlink and uplink volume after which the usage threshold is considered to be reached.
usageThreshold.downlinkVolume	integer	Downlink data octets
usageThreshold.uplinkVolume	integer	Uplink data octets
sponsorInfo	object	The information about the sponsor who is sponsoring the traffic
sponsorInfo.sponsorId	string	The sponsor ID.
sponsorInfo.aspld	string	The Application Service Provider ID.
qosMonInfo	string	Requested Qos Monitoring parameters. This field must be included when the AF needs to subscribe for the QOS_MONITORING event. When this value is available, NEF subscribes for the QOS_MONITORING event in the Npcf_PolicyAuthorization request. This is an optional parameter.
qosMonInfo.reqQosMonParams	array	Specifies the parameters to be monitored for a service data flow. The possible values are: • DOWNLINK • UPLINK • ROUND_TRIP
qosMonInfo.repFreqs	array	Indicates the frequency for the reporting, such as event triggered, periodic, when the PDU Session is released, or any other combination.

Table 2-38 (Cont.) Request Body Parameters

Field Name	Data Type	Description
qosMonInfo.repThreshDI	integer	Contains an unsigned integer.
qosMonInfo.repThreshUI	integer	Contains an unsigned integer.
qosMonInfo.repThreshRp	integer	Contains an unsigned integer.
qosMonInfo.waitTime	integer	The wait duration in seconds.
qosMonInfo.repPeriod	integer	The wait duration in seconds.

Note

UE address (IP address or MAC address), Flow description(s), QoS Reference are all mandatory parameter for a AF Session with QoS subscription. The data types supported by NEF complies with the 3GPP specifications. For more information about the 3GPP data types, see [3GPP Technical Specification 29.122, Release 16, T8 reference point for Northbound APIs](#).

Table 2-39 Supported Response Codes

Code	Description
201	The subscription resource is created successfully.
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
411	Content length required
413	Payload large
415	Unsupported Media Type
500	Internal Server Error
503	Service Unavailable

Examples

The following example shows how an AF creates a subscription request for notification about QoS monitoring event by submitting a POST request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request POST 'http://<IP
address:Port>/3gpp-as-session-with-qos/v1/af1/subscriptions' --header
'Content-Type: application/json' \ -H "accept: application/json -H
'authorization:
    Bearer
eyJ0eXAiOiJKV1QiLCJpbnZva2VyTmFtZSI6ImIudm9rZXItcHJvZm1sZSIsImFsZyI6I1JTMjU2In
0.eyJpc3MiOiI3OWE1OWYyMi1iMzMzLTJiMTItYWE3NS1hMTgxM2NkYWZzZTciLCJzY29wZSI6IjNn
cHAjZmYwYzYk1NjctMDkzZC00YTJkLTk5MmQtZmNhMDNlNTQyZWZm0jNncHAtbW9uaXRvcmluZy1ldm
VudCIsImV4cCI6MTY2ODE4NzcxMn0.NsOARB8K7ahZYX1w-kt4x0aX-
0EJW45WjnijC9Ki8lSxZlQcrNjQnQZLUN9Pzf6Bvwbf7-
yPw42_JD7k9I3ku54MuK1nNVIMAUD_mm_AIz1jw9tJ9hSG39NA63eB_HtBw3J60dM9B1_pHELT-8g
```

```
GTt7MCQ_V96REZsjHJ5EN9zbXVmmsb5H4J4TJMwAA9dQh_L-
gFiMX9m4dZ08PvhtYE_2Q_ptWaTsZUNj00yn6FKB6G0EYbIvqA3Wykufo1t400p6pa7FJbpfvTBXBK
GQ1Czv33kB_Vj1tqswpxCKjva_mA20o0AZmW2Y6p9_2QouIC0LFrsI_HIaRtjF7AUg'
```

Note

The authorization bearer value is obtained from the `access_token` parameter in the response body of the Obtain Authorization API. For more information about the parameter, see [Obtain Authorization](#).

Example of the Request Body

The following snippet shows an example of the request body for the QoS Monitoring and Usage Report events subscriptions:

```
{
  "notificationDestination": "http://ocnefsim-ocstub-svc-pcf:1010/af/
notification",
  "flowInfo": [
    {
      "flowId": 1234,
      "flowDescriptions": ["DL"]
    }
  ],
  "qosMonInfo": {
    "reqQosMonParams": ["DOWNLINK", "UPLINK", "ROUND_TRIP"],
    "repFreqs": ["EVENT_TRIGGERED"],
    "repThreshDl": 50,
    "repThreshUl": 55,
    "repThreshRp": 60,
    "waitTime": 66,
    "repPeriod": 50
  },
  "usageThreshold": {
    "duration": 40,
    "totalVolume": 30,
    "downlinkVolume": 99,
    "uplinkVolume": 89
  },
  "qosReference": "5QI",
  "ueIpv4Addr": "192.168.1.21"
}
```

Example of the Response Body

```
{
  "notificationDestination": "http://ocnefsim-ocstub-svc-pcf:1010/af/
notification",
  "flowInfo": [
    {
      "flowId": 1234,
      "flowDescriptions": ["DL"]
    }
  ]
}
```

```

    }
  ],
  "qosMonInfo":{
    "reqQosMonParams": ["DOWNLINK", "UPLINK", "ROUND_TRIP"],
    "repFreqs": ["EVENT_TRIGGERED"],
    "repThreshDl": 50,
    "repThreshUl": 55,
    "repThreshRp": 60,
    "waitTime": 66,
    "repPeriod": 50
  },
  "usageThreshold":{
    "duration": 40,
    "totalVolume": 30,
    "downlinkVolume" : 99,
    "uplinkVolume" : 89
  },
  "qosReference": "5QI",
  "ueIpv4Addr": "192.168.1.21"
}

```

Note

The notificationDestination link is described as follows:

http://<AF Hostname>:<AF Port>/<any path supported by AF for notifications>

2.1.8.2 Get QoS Session Details

This service operation is used to fetch details for a specific QoS session subscription from NEF.

Type: GET

URI: /3gpp-as-session-with-qos/v1/{afId}/subscriptions/<subscription Id>

Initiated By: AF

Important

The afId in path param should be same as apiInvokerId in the [Onboard API Invoker Response](#).

Table 2-40 Request Body Parameters

Field Name	Data Type	Description
afId	string	A unique identifier of the API invoker. This is a mandatory parameter.

Table 2-40 (Cont.) Request Body Parameters

Field Name	Data Type	Description
subscriptionId	string	The unique identifier of the QoS subscription. This is a mandatory parameter.

Table 2-41 Response Body Parameters

Field Name	Data Type	Description
notificationDestination	link	The destination URI where the notification for the subscription must be sent. This is a mandatory parameter.
flowInfo	array	Represents flow information. This is a mandatory parameter.
flowInfo.flowId	integer	Identifies the IP flows that is sent during event subscription
flowInfo.flowDescriptions	string	Indicates the packet filters of the IP flow
qosMonInfo	array	Qos Monitoring information. It can be present when the event "QOS_MONITORING" is subscribed
reqQosMonParams	array	Requested Qos Monitoring Parameters
repFreqs	array	Indicates the frequency for the reporting, such as event triggered, periodic, when the PDU Session is released, and/or any combination.
usageThreshold	object	Represents the usage threshold.
usageThreshold.duration	integer	Indicates the duration in seconds after which the usage threshold is considered to be reached.
usageThreshold.totalVolume	integer	Indicates the total data octets for both downlink and uplink volume after which the usage threshold is considered to be reached.
usageThreshold.downlinkVolume	integer	Downlink data octets
usageThreshold.uplinkVolume	integer	Uplink data octets
qosReference	string	Identifies a pre-defined QoS information
ueIpv4Addr	string	The Ipv4 address of the UE.

Note

The data types supported by NEF comply with the 3GPP specifications. For more information about the 3GPP data types, see [3GPP Technical Specification 29.571, Release 16, Common Data Types for Service Based Interfaces](#).

Examples

The following example shows how an AF fetches the ME subscription details by submitting a GET request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request GET 'http://<IP
address:Port>/3gpp-as-session-with-qos/v1/af1/subscriptions/
f13f507c-8414-4608-9e6e-34d6d0f0f530' HTTP/2' \--header 'Content-Type:
application/json' \ -H 'authorization:
    Bearer
eyJ0eXAiOiJKV1QiLCJpb2Vya2VyTmFtZSI6ImIudm9rZXItcHJvZmIsZSIsImFsZyI6IiJTMjU2In
0.eyJpc3MiOiI3OWE1OWYyMi1iMzMzLTJhMTItYWE3NS1hMTgxM2NkYWZlZTciLCJzY29wZSI6IjNn
cHAjZmYwYzkyNjctMDk5OTY0YTRmLTk5MmQzMmNhMDNlNTQyZWZmOjNncHAAtbW9uaXRvcmluZy1ldm
VudCIsImV4cCI6MTY2ODE4NzcxMn0.NsOARB8K7ahZYX1w-kt4x0aX-
0EJW45WjnjC9Ki8lSxZlQcrNjQnQZLUN9Pzf6Bvwbxf7-
yPw42_JD7k9I3ku54MuK1nNVIMAUD_mm_AIZ1jW9tJ9hSG39NA63eB_HtBw3J60dM9B1_pHELT-8g
GTt7MCQ_V96REZsjHJ5EN9zbxVmmsb5H4J4TJMwAA9dQh_L-
gFiMX9m4dZ08PvhtYE_2Q_ptWaTsZUNj00yn6FKB6G0EYbIvqA3Wykufolt400p6pa7FJbpfvTBXBK
GQ1Czv33kB_Vj1tqswpxCKjva_mA20o0AZmw2Y6p9_2QouIC0LFrsI_HiArtjF7AUg'
\
```

Note

The authorization bearer value is obtained from the `access_token` parameter in the response body of the Obtain Authorization API. For more information about the parameter, see [Obtain Authorization](#).

Example of the Response Body

```
{
  "self": "http://ocnef-ext-nef-ingress-gateway:80/3gpp-as-session-with-
qos/v1/af1/subscriptions/f13f507c-8414-4608-9e6e-34d6d0f0f530",
  "notificationDestination": "http://ocnefsim-ocstub-svc-pcf:1010/af/
notification",
  "flowInfo": [
    {
      "flowId": 1234,
      "flowDescriptions": [
        "DL"
      ]
    }
  ],
  "qosReference": "5QI",
  "ueIpv4Addr": "192.168.1.21",
  "usageThreshold": {
    "duration": 40,
    "totalVolume": 30,
    "downlinkVolume": 99,
    "uplinkVolume": 89
  },
  "qosMonInfo": {
```

```
    "reqQosMonParams": [  
      "DOWNLINK",  
      "UPLINK",  
      "ROUND_TRIP"  
    ],  
    "repFreqs": [  
      "EVENT_TRIGGERED"  
    ],  
    "repThreshDl": 50,  
    "repThreshUl": 55,  
    "repThreshRp": 60,  
    "waitTime": 66,  
    "repPeriod": 50  
  }  
}
```

Note

The notificationDestination link is described as follows:

http://<AF Hostname>:<AF Port>/<any path supported by AF for notifications>

2.1.8.3 Update (PUT) AF Session with QoS Subscription

This service operation is used to update an existing subscription resource for QoS provisioning and notifications.

Note

the Update QoS Subscription operation does not support the removal of any existing parameter. It only supports the addition of new parameters or updating the value of an existing parameter.

Type: PUT

URI: /3gpp-as-session-with-qos/v1/{afId}/subscriptions/{subscription Id}

Initiated By: AF

Important

The afId in path param should be same as apiInvokerId in the [Onboard API Invoker Response](#).

Table 2-42 Request Body Parameters

Field Name	Data Type	Description
self	string	Specifies the referenced resource provided by NEF after subscription creation.
supportedFeatures	string	List the features supported by NEF.
notificationDestination	link	The destination URI where the notification for the subscription must be sent. This is a mandatory parameter.
flowInfo	array	Represents flow information. This is a mandatory parameter.
flowInfo.flowId	integer	Identifies the IP flows that is sent during event subscription
flowInfo.flowDescriptions	string	Indicates the packet filters of the IP flow
ethFlowInfo	array	The Ethernet data flow.
ethFlowInfo.EthFlowDescription	object	Defines a packet filter for an Ethernet flow.
qosReference	string	The predefined QoS information.
altQoSReferences	array	Represent that the subscription request is for the QOS_NOTIF event. This field must be included when the AF needs to subscribe for the QOS_NOTIF event. When this value is available, NEF subscribes for the QOS_NOTIF event in the Npcf_PolicyAuthorization request. This is an optional parameter.
disUeNotif	boolean	Specifies if QoS flow parameter signalling to the UE should be disabled when the SMF is notified by the NG-RAN of changes in the fulfilled QoS situation.
uelpv4Addr	string	The IPv4 address of the UE.
uelpv6Addr	string	The IPv6 address of the UE.
macAddr	string	The Media Access Control (MAC) address of the UE.
ipDomain	string	The IPv4 domain identifier. This field is applicable only if uelpv4Addr is available.

Table 2-42 (Cont.) Request Body Parameters

Field Name	Data Type	Description
usageThreshold	object	Represent that the subscription request for the USAGE_REPORT event. This field must be included when the AF needs to subscribe for the USAGE_REPORT event. When this value is available, NEF subscribes for the USAGE_REPORT event in the Npcf_PolicyAuthorization request.
usageThreshold.duration	integer	Indicates the duration in seconds after which the usage threshold is considered to be reached.
usageThreshold.totalVolume	integer	Indicates the total data octets for both downlink and uplink volume after which the usage threshold is considered to be reached.
usageThreshold.downlinkVolume	integer	Downlink data octets
usageThreshold.uplinkVolume	integer	Uplink data octets
sponsorInfo	object	The information about the sponsor who is sponsoring the traffic
sponsorInfo.sponsorId	string	The sponsor ID.
sponsorInfo.aspid	string	The Application Service Provider ID.
qosMonInfo	string	Requested Qos Monitoring parameters. This field must be included when the AF needs to subscribe for the QOS_MONITORING event. When this value is available, NEF subscribes for the QOS_MONITORING event in the Npcf_PolicyAuthorization request. This is an optional parameter.
qosMonInfo.reqQosMonParams	array	Specifies the parameters to be monitored for a service data flow. The possible values are: • DOWNLINK • UPLINK • ROUND_TRIP
qosMonInfo.repFreqs	array	Indicates the frequency for the reporting, such as event triggered, periodic, when the PDU Session is released, or any other combination.
qosMonInfo.repThreshDL	integer	Contains an unsigned integer.
qosMonInfo.repThreshUL	integer	Contains an unsigned integer.
qosMonInfo.repThreshRp	integer	Contains an unsigned integer.
qosMonInfo.waitTime	integer	The wait duration in seconds.

Table 2-42 (Cont.) Request Body Parameters

Field Name	Data Type	Description
qosMonInfo.repPeriod	integer	The wait duration in seconds.

Note

The data types supported by NEF complies with the 3GPP specifications. For more information about the 3GPP data types, see [3GPP Technical Specification 29.122, Release 16, T8 reference point for Northbound APIs](#).

Table 2-43 Supported Response Codes

Code	Description
200	OK (Successful update of the subscription)
415	Unsupported Media Type

Examples

The following example shows how an AF modifies a subscription for QoS monitoring event by submitting a PUT request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request PUT 'http://<IP
address:Port>/3gpp-as-session-with-qos/v1/af1/subscriptions/b10e96ec-
cc15-4753-af87-50bf646d31e3' \--header 'Content-Type: application/json' \ -H
'authorization:
    Bearer
eyJ0eXAiOiJKV1QiLCJpbnZva2VyTmFtZSI6ImIudm9rZXItcHJvZmIsZSIsImFsZyI6ImlJTMjU2In
0.eyJpc3MiOiI3OWE1OWYyMi1iMzMzLTlRMTItYWE3NS1hMTgxM2NkYWZlZTciLCJzY29wZSI6IjNn
cHAjZmYwYzkyNjctMDEkZC00YTRmLTk5MmQzMmNhMDNlNTQyZWZmOjNncHAtbW9uaXRvcmluZy1ldm
VudCIsImV4cCI6MTY2ODE4NzcxMn0.NsOARB8K7ahZYX1w-kt4x0aX-
0EJW45WjnijC9Ki8lSxZlQcrNjQnQZLUN9Pzf6Bvwbxf7-
yPw42_JD7k9I3ku54MuK1nNVIMAUD_mm_AIZ1jW9tJ9hSG39NA63eB_HtBw3J60dM9B1_pHELT-8g
GTt7MCQ_V96REZsjHJ5EN9zbxVmmsb5H4J4TJMwDA9dQh_L-
gFiMx9m4dZ08PVhtYE_2Q_ptWaTsZUNj00yn6FKB6G0EYbIvqA3Wykufolt400p6pa7FJbpfvTBXBK
GQ1Czv33kB_Vj1tqswpxCKjva_mA20o0AZmW2Y6p9_2QouIC0LFrSI_HiArtjF7AUg'
```

Note

The authorization bearer value is obtained from the access_token parameter in the response body of the Obtain Authorization API. For more information about the parameter, see [Obtain Authorization](#).

Example of the Request Body

The following snippet shows an example of the request body for the QoS Monitoring and Usage Report events subscriptions:

```
{
  "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/
notification",
  "flowInfo": [
    {
      "flowId": 1234,
      "flowDescriptions": ["DL"]
    }
  ],
  "qosMonInfo": {
    "reqQosMonParams": ["DOWNLINK", "UPLINK", "ROUND_TRIP"],
    "repFreqs": ["EVENT_TRIGGERED"],
    "repThreshDl": 150,
    "repThreshUl": 155,
    "repThreshRp": 160,
    "waitTime": 166,
    "repPeriod": 150
  },
  "usageThreshold": {
    "duration": 140,
    "totalVolume": 130,
    "downlinkVolume": 199,
    "uplinkVolume": 189
  },
  "qosReference": "5QI",
  "altQoSReferences": ["BQ"],
  "ueIpv6Addr": "2301:0db8:1001:0000:0000:0ab9:C0A8:0113"
}
```

Example of the Response Body

HTTP/2 201

Note

The notificationDestination link is described as follows:

```
http://<AF Hostname>:<AF Port>/<any path supported by AF for
notifications>
```

2.1.8.4 Update (Patch) AF Session with QoS Subscription

This service operation is used to partially update or replace an existing resource of a QoS subscription.

Note

the Update QoS Subscription operation does not support the removal of any existing parameter. It only supports the addition of new parameters or updating the value of an existing parameter.

Type: PATCH

URI: /3gpp-as-session-with-qos/v1/{afId}/subscriptions/{subscription Id}

Initiated By: AF

Important

The afId in path param should be same as apiInvokerId in the [Onboard API Invoker Response](#).

Table 2-44 Request Body Parameters

Field Name	Data Type	Description
flowInfo	array	Represents flow information. This is a mandatory parameter.
flowInfo.flowId	integer	Identifies the IP flows that is sent during event subscription
flowInfo.flowDescriptions	string	Indicates the packet filters of the IP flow
ethFlowInfo	array	The Ethernet data flow.
ethFlowInfo.EthFlowDescription	object	Defines a packet filter for an Ethernet flow.
qosReference	string	The predefined QoS information.
altQoSReferences	array	Represent that the subscription request is for the QOS_NOTIF event. This field must be included when the AF needs to subscribe for the QOS_NOTIF event. When this value is available, NEF subscribes for the QOS_NOTIF event in the Npcf_PolicyAuthorization request. This is an optional parameter.
disUeNotif	boolean	Specifies if QoS flow parameter signalling to the UE should be disabled when the SMF is notified by the NG-RAN of changes in the fulfilled QoS situation.

Table 2-44 (Cont.) Request Body Parameters

Field Name	Data Type	Description
usageThreshold	object	Represent that the subscription request for the USAGE_REPORT event. This field must be included when the AF needs to subscribe for the USAGE_REPORT event. When this value is available, NEF subscribes for the USAGE_REPORT event in the Npcf_PolicyAuthorization request.
usageThreshold.duration	integer	Indicates the duration in seconds after which the usage threshold is considered to be reached.
usageThreshold.totalVolume	integer	Indicates the total data octets for both downlink and uplink volume after which the usage threshold is considered to be reached.
usageThreshold.downlinkVolume	integer	Downlink data octets
usageThreshold.uplinkVolume	integer	Uplink data octets
qosMonInfo	string	Requested Qos Monitoring parameters. This field must be included when the AF needs to subscribe for the QOS_MONITORING event. When this value is available, NEF subscribes for the QOS_MONITORING event in the Npcf_PolicyAuthorization request. This is an optional parameter.
qosMonInfo.reqQosMonParams	array	Specifies the parameters to be monitored for a service data flow. The possible values are: • DOWNLINK • UPLINK • ROUND_TRIP
qosMonInfo.repFreqs	array	Indicates the frequency for the reporting, such as event triggered, periodic, when the PDU Session is released, or any other combination.
qosMonInfo.repThreshDI	integer	Contains an unsigned integer.
qosMonInfo.repThreshUI	integer	Contains an unsigned integer.
qosMonInfo.repThreshRp	integer	Contains an unsigned integer.
qosMonInfo.waitTime	integer	The wait duration in seconds.
qosMonInfo.repPeriod	integer	The wait duration in seconds.

Note

The data types supported by NEF complies with the 3GPP specifications. For more information about the 3GPP data types, see [3GPP Technical Specification 29.122, Release 16, T8 reference point for Northbound APIs](#).

Table 2-45 Supported Response Codes

Code	Description
200	OK (Successful update of the subscription)
400	Bad Request
401	Unauthorized
404	Not Found
411	Content length required
413	Payload large
415	Unsupported Media Type
500	Internal Server Error
503	Service Unavailable

Examples

The following example shows how an AF partially modifies or replace a subscription resource for QoS monitoring event by submitting a PATCH request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request PATCH 'http://<IP
address:Port>/3gpp-as-session-with-qos/v1/af1/subscriptions/b10e96ec-
cc15-4753-af87-50bf646d31e3' \--header 'Content-Type: application/json' \ -H
'authorization:
    Bearer
eyJ0eXAiOiJKV1QiLCJpbnZva2VyTmFtZSI6ImIudm9rZXItcHJvZmIsZSIsImFsZyI6I1JTMjU2In
0.eyJpc3MiOiI3OWE1OWYyMi1iMzMzLTJMTiYWE3NS1hMTgxM2NkYWZlZTciLCJzY29wZSI6IjNn
cHAjZmYwYzYk1NjctMDBkZC00YTRmLTk5MmQtZmNhMDNlNTQyZWZRM0jNncHAtbW9uaXRvcmluZy1ldm
VudCIsImV4cCI6MTY2ODE4NzcxMn0.NsOARB8K7ahZYX1w-kt4x0aX-
0EJW45WjnijC9Ki8lSxZlQcrNjQnQZLUN9Pzf6Bvwbxf7-
yPw42_JD7k9I3ku54MuK1nNVIMAUD_mm_AIZ1jW9tJ9hSG39NA63eB_HtBw3J60dM9B1_pHELT-8g
GTt7MCQ_V96REZsjHJ5EN9zbxVmmsb5H4J4TJMwAA9dQh_L-
gFiMX9m4dZ08PvhtYE_2Q_ptWaTsZUNj00yn6FKB6G0EYbIvqA3Wykufolt400p6pa7FJbpfvTBXBK
GQ1Czv33kB_Vj1tqswpxCKjva_mA20o0AZmW2Y6p9_2QouIC0LFrsI_HiArTjF7AUg'
```

Note

The authorization bearer value is obtained from the `access_token` parameter in the response body of the Obtain Authorization API. For more information about the parameter, see [Obtain Authorization](#).

Example of the Request Body

The following snippet shows an example of the request body for the QoS Monitoring and Usage Report events subscriptions:

```
{
  "flowInfo": [
    {
      "flowId": 1234,
      "flowDescriptions": ["DL"]
    }
  ],
  "qosMonInfo": {
    "reqQosMonParams": ["DOWNLINK", "UPLINK", "ROUND_TRIP"],
    "repFreqs": ["EVENT_TRIGGERED"],
    "repThreshDl": 450,
    "repThreshUl": 455,
    "repThreshRp": 460,
    "waitTime": 466,
    "repPeriod": 450
  },
  "usageThreshold": {
    "duration": 40,
    "totalVolume": 30,
    "downlinkVolume": 99,
    "uplinkVolume": 89
  },
  "qosReference": "5QI"
}
```

Example of the Response Body

HTTP/2 201

2.1.8.5 Delete a QoS Subscription

This service operation is used to delete an QoS subscription.

Type: DELETE

URI: /3gpp-as-session-with-qos/v1/{afID}/subscriptions/{subscriptionId}

Initiated By: AF

Examples

The following example shows how an AF deletes a QoS subscription by submitting a DELETE request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request DELETE 'http://<IP
address:Port>/3gpp-as-session-with-qos/v1/af1/subscriptions/
f13f507c-8414-4608-9e6e-34d6d0f0f530' \ --header 'Accept: application/json'
\ --header
'Authorization: Bearer
eyJ0eXAiOiJKV1QiLCJpbnZva2VyTmFtZSI6ImZvcnRpZnkzIiwiaWwiYXNjaWoiOiUyNTYifQ.eyJpc3M
iOiJmZWUzMzk0Ni04YzJlLTQxMTYtODFjYi1mZGJjYmNjZGZjZjMiLCJzY29wZSI6IjNnCHAjYzY10
```

```
Dgx0TktN2RjNS00MTQ5LWJhNDAtMzFmNmY4MzFlMThlOjNncHAtbW9uaXRvcmluZy1ldmVudCIsImV4cCI6MTY2ODE4NzcxMn0.Kbx1UDWrbtu7xUYehowu_b2q1p-
MmWMQ9ixVrPLbqLrPGSy6IcPuVHKcmgIOzNfY-40EIUHG6xvQFGej1TVXooNCOKe-e9YYRVtyzmo-
aZN-
INlT3JyfaCw_zAY0pQg9ttJSBshEAgaop4GyrylvymKlZpDAs9Q80IegBBbjCR7xkQDnJRK9qNqzkw
DbPrXUkU0nDMQoIXiDX6QRdXr3n1wVQ6BeLVMPPrR6a0z2JcqVr-INMs7BM0xP_-
XI42o2WASTFvfrwk_T1v08GpWyj1xQmLSTCcKouBlX6i-
NnFmpZR9LUYvIO4m53C05oXvA5JYGwctRzLcxpKynewVvGHlQ' \
```

Note

The authorization bearer value is obtained from the `access_token` parameter in the response body of the Obtain Authorization API. For more information about the parameter, see [Obtain Authorization](#).

2.1.8.6 QoS Notification

The following service operations are used by PCF for QoS notification:

- Notify**
 This service operation is used by PCF to notify about the QoS events corresponding to subscriptions:
Type: POST
URI: /5gc-agent/v1/<translator>/<encoded-notification-receiving-service-url>/notify
Initiated By: PCF
- Terminate Notification**
 This service operation is used by PCF to terminate a notifications about the QoS events corresponding to subscriptions:
Type: POST
URI: /5gc-agent/v1/<translator>/<encoded-notification-receiving-service-url>/terminate
Initiated By: PCF

Table 2-46 Request/Response Body Parameters

Field Name	Data Type	Description
notifyURI	string	The parameter provided by NEF to uniquely correlate the notifications with the subscription This is a mandatory parameter.

Table 2-47 Supported Response Codes

Code	Description
204	No Content (successful notification)
400	Bad Request
401	Unauthorized
404	Not Found

Table 2-47 (Cont.) Supported Response Codes

Code	Description
411	Content length required
413	Payload large
500	Internal Server Error
503	Service Unavailable

Examples

The following example shows how the PCF notifies about QoS event by submitting a POST request on the REST resource using cURL:

cURL Command for Notify

```
curl --http2-prior-knowledge --location --request POST '<IP
Address:Port>/5gc-agent/v1/QOS/b2NuZWYtbW9uaXRvcmluZ2V2ZW50czo4MDgw/
notify' --header
'Content-Type: application/json'
```

cURL Command for Terminate Notification

```
curl --http2-prior-knowledge --location --request POST '<IP
Address:Port>/5gc-agent/v1/QOS/b2NuZWYtbW9uaXRvcmluZ2V2ZW50czo4MDgw/
terminate' --header
'Content-Type: application/json'
```

Example of Request body

- Termination Notification**

```
{
  "transaction": "http://ocnef-ext-nef-ingress-gateway:80/3gpp-as-session-
with-qos/v1/af2/subscriptions/8cfd071-fa3c-45eb-961b-73c3892f7a4d",
  "eventReports": [
    {
      "event": "SESSION_TERMINATION"
    }
  ]
}
```

- Qos Monitoring Report Notification**

```
{
  "transaction": "http://ocnef-ext-nef-ingress-gateway:80/3gpp-as-session-
with-qos/v1/af2/subscriptions/8cfd071-fa3c-45eb-961b-73c3892f7a4d",
  "eventReports": [
    {
      "event": "QOS_MONITORING",
      "flowIds": [1, 2],
      "qosMonReports": [
        {
          "ulDelays": [50, 20],

```

```

        "dlDelays": [10, 30],
        "rtDelays": [5, 7]
      }
    ]
  }
}

```

- **Qos Usage Report Notification**

```

{
  "transaction": "http://ocnef-ext-nef-ingress-gateway:80/3gpp-as-session-
with-qos/v1/af2/subscriptions/8cfd071-fa3c-45eb-961b-73c3892f7a4d",
  "eventReports": [
    {
      "event": "USAGE_REPORT",
      "accumulatedUsage": {
        "duration": 40,
        "totalVolume": 30,
        "downlinkVolume": 99,
        "uplinkVolume": 89
      },
      "flowIds": [1, 2]
    }
  ]
}

```

- **Qos Notification Control**

```

{
  "transaction": "http://ocnef-ext-nef-ingress-gateway:80/3gpp-as-session-
with-qos/v1/af2/subscriptions/8cfd071-fa3c-45eb-961b-73c3892f7a4d",
  "eventReports": [
    {
      "event": "QOS_GUARANTEED",
      "flowIds": [1, 2, 1, 2]
    }
  ]
}

```

Example of Response body

- **Termination Notification**

"204 No Content"

- **Qos Monitoring Report Notification**

"204 No Content"

- **Qos Usage Report Notification**

"204 No Content"

- **Qos Notification Control**

"204 No Content"

2.1.9 Traffic Influence Subscription

The Traffic Influence resource allows Application Function (AFs) to read all of the active Traffic Influence subscriptions or create a new Traffic Influence subscription at NEF through the 3gpp-traffic-influence API.

The Traffic Influence API allows the AFs to subscribe to notifications about specific subscriptions. It also allows the NEF to report the event by sending notifications when an event is detected.

The 3gpp-traffic-influence service facilitates the following operations:

2.1.9.1 Create Traffic Influence Subscription

This service operation is used to create a new subscription resource for Traffic Influence.

Type: POST

URI: /3gpp-traffic-influence/v1/{afId}/subscriptions

Initiated By: AF

Table 2-48 Path Parameters

Field Name	Data Type	Description
afId	string	A unique identifier of the AF.
afServiceId	string	Identifies a service on behalf of which the AF is issuing the request.
afAppId	string	Identifies an application.
afTransId	string	Identifies an NEF Northbound interface transaction, generated by the AF.
appReloInd	boolean	Identifies whether an application can be relocated once a location of the application has been selected.
dnn	string	Identifies the Data Network Name.
snssai	string	Identifies Single Network Slice Selection Assistance Information.
externalGroupId	string	Group of UEs identified by External Group Identifier.
anyUeInd	boolean	Identifies whether the AF request applies to any UE. This attribute shall set to "true" if applicable for any UE, otherwise, set to "false".
subscribedEvents	string	Identifies the requirement to be notified of the event(s). Only 'UP_PATH_CHANGE' is supported.
gpsi	string	Identifies a GPSI.
ipv4Addr	string	String identifying an IPv4 address.
ipDomain	string	The IPv4 domain identifier.
ipv6Addr	string	String identifying an IPv6 address.

Table 2-48 (Cont.) Path Parameters

Field Name	Data Type	Description
macAddr	string	The Media Access Control (MAC) address of the UE.
dnaiChgType	string	Describes the types of DNAI change.
notificationDestination	string	The destination URI where the notification for the subscription must be sent. This is a mandatory parameter.
requestTestNotification	boolean	Set to true by the SCS/AS to request the NEF to send a test notification. Set to false or do not use this.
websockNotifConfig	string	Configuration parameters to set up notification delivery over Websocket protocol.
self	string	Specifies the referenced resource provided by NEF after subscription creation.
trafficFilters	string	Identifies IP packet filters.
ethTrafficFilters	string	Identifies Ethernet packet filters.
trafficRoutes	string	Identifies the N6 traffic routing requirement.
tfcCorrInd	boolean	Indication of traffic correlation.
tempValidities	string	Indicates the time interval(s) during which the AF request is to be applied.
validGeoZoneIds	string	Identifies a geographic zone that the AF request applies only to the traffic of UE(s) located in this specific zone.
afAckInd	boolean	Identifies whether the AF acknowledgement of UP path event notification is expected.
addrPreserInd	boolean	Indicates whether UE IP address should be preserved.
suppFeat	string	Indicates the list of supported features.

Note

- All the path parameters in the above table are mandatory.
- The data types supported by NEF complies with the 3GPP specifications. For more information about the 3GPP data types, see [3GPP Technical Specification 29.571, Release 16, Common Data Types for Service Based Interfaces](#).

Table 2-49 Supported Response Codes

Code	Description
201	Created (Successful creation of subscription)
400	Bad Request
401	Unauthorized
403	Request is forbidden.
404	Not Found
411	
413	

Table 2-49 (Cont.) Supported Response Codes

Code	Description
415	Unsupported Media Type
429	
500	
503	Service Unavailable
default	

Examples

The following example shows how an AF creates a subscription request for notification about traffic influence by submitting a POST request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request POST 'http://<IP
address:Port>/3gpp-traffic-influence/v1/{afId}/subscriptions' \
--header 'Content-Type: application/json' \
```

Example of the Request Body

The following snippet shows an example of the request body.

```
{
  "afServiceId": "afServiceId1",
  "afTransId": "1524",
  "subscribedEvents": [
    "UP_PATH_CHANGE"
  ],
  "dnn": "s1",
  "ipv4Addr": "66.122.33.88",
  "dnaiChgType": "EARLY_LATE",
  "notificationDestination": "http://af_fqdn:1010/af/notification",
  "trafficFilters": [
    {
      "flowId": 1234,
      "flowDescriptions": [
        "UL"
      ]
    }
  ],
  "trafficRoutes": [
    {
      "dnai": "asd",
      "routeInfo": {
        "ipv4Addr": "10.10.10.10",
        "portNumber": 5050
      },
      "routeProfId": "profile1"
    }
  ]
}
```

```
    ]
  }
}
```

Example of the Response Body

```
{
  "afServiceId": "afServiceId1",
  "afAppId": null,
  "afTransId": "1524",
  "appReloInd": null,
  "dnn": "s1",
  "snssai": null,
  "externalGroupId": null,
  "anyUeInd": null,
  "subscribedEvents": [
    "UP_PATH_CHANGE"
  ],
  "gpsi": null,
  "ipv4Addr": "66.122.33.88",
  "ipDomain": null,
  "ipv6Addr": null,
  "macAddr": null,
  "dnaiChgType": "EARLY_LATE",
  "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/
notification",
  "requestTestNotification": null,
  "websocketNotifConfig": null,
  "self": "http://ocnef-ext-nef-ingress-gateway:80/3gpp-traffic-
influence/v1/d2ad29d1-8c8e-4fb2-872e-937cc34bda53/subscriptions/
4606ec28-5a0a-4451-acaf-341733201857",
  "trafficFilters": [
    {
      "flowId": 1234,
      "flowDescriptions": [
        "UL"
      ]
    }
  ],
  "ethTrafficFilters": null,
  "trafficRoutes": [
    {
      "dnai": "asd",
      "routeInfo": {
        "ipv4Addr": "10.10.10.10",
        "ipv6Addr": null,
        "portNumber": 5050
      },
      "routeProfId": ""
    }
  ],
  "tfcCorrInd": null,
  "tempValidities": null,
  "validGeoZoneIds": null,
  "afAckInd": null,
  "addrPreserInd": null,
}
```

```
"suppFeat": null  
}
```

Note

The notificationDestination link is described as follows:

`http://<AF Hostname>:<AF Port>/<any path supported by AF for notifications>`

2.1.9.2 Get Traffic Influence Subscription

This service operation is used to fetch details for a specific Traffic Influence from NEF.

Type: GET

URI: /3gpp-traffic-influence/v1/{afId}/subscriptions/{subscriptionId}

Initiated By: AF

Table 2-50 Path Parameters

Field Name	Data Type	Description
afId	string	A unique identifier of the AF.
subscriptionId	string	A unique identifier of the subscription resource.

Note

- All the path parameters in the above table are mandatory.
- The data types supported by NEF complies with the 3GPP specifications. For more information about the 3GPP data types, see [3GPP Technical Specification 29.571, Release 16, Common Data Types for Service Based Interfaces](#).

Table 2-51 Supported Response Codes

Code	Description
200	OK (Successful get the active subscription)
307	
308	
400	Bad Request
401	Unauthorized
403	Request is forbidden.
404	Not Found
406	
429	
500	

Table 2-51 (Cont.) Supported Response Codes

Code	Description
503	Service Unavailable
default	

Examples

The following example shows how an AF fetches details for a specific Traffic Influence from NEF by submitting a GET request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request GET 'http://<IP
address:Port>/3gpp-traffic-influence/v1/{afId}/subscriptions/
{subscriptionId}' \--header 'Content-
Type: application/json' \
```

Example of the Response Body

```
{
  "afServiceId": "afServiceId1",
  "afAppId": "af_ti_app1",
  "afTransId": "1524",
  "appReloInd": null,
  "dnn": null,
  "snssai": null,
  "externalGroupId": null,
  "anyUeInd": null,
  "subscribedEvents": [
    "UP_PATH_CHANGE"
  ],
  "gpsi": "adfsdfuserc123om",
  "ipv4Addr": null,
  "ipDomain": null,
  "ipv6Addr": null,
  "macAddr": null,
  "dnaiChgType": "EARLY_LATE",
  "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/
notification",
  "requestTestNotification": null,
  "websockNotifConfig": null,
  "self": "http://ocnef-ext-nef-ingress-gateway:80/3gpp-traffic-
influence/v1/d2ad29d1-8c8e-4fb2-872e-937cc34bda53/subscriptions/
f7271524-2af2-4154-b3e6-78d686655187",
  "trafficFilters": null,
  "ethTrafficFilters": null,
  "trafficRoutes": [
    {
      "dnai": "asd",
      "routeInfo": {
        "ipv4Addr": "10.10.10.10",
        "ipv6Addr": null,
        "portNumber": 5050
      }
    }
  ]
}
```

```

        },
        "routeProfId": ""
    }
],
"tfcCorrInd": null,
"tempValidities": null,
"validGeoZoneIds": null,
"afAckInd": null,
"addrPreserInd": null,
"suppFeat": null
}

```

① Note

The notificationDestination link is described as follows:

`http://<AF Hostname>:<AF Port>/<any path supported by AF for notifications>`

2.1.9.3 Update Traffic Influence Subscription

This service operation is used to update a subscription resource for Traffic Influence.

Type: PUT

URI: /3gpp-traffic-influence/v1/{afId}/subscriptions/{subscriptionId}

Initiated By: AF

Table 2-52 Path Parameters

Field Name	Data Type	Description
afId	string	A unique identifier of the AF.
subscriptionId	string	Identifier of the subscription resource.

① Note

- All the path parameters in the above table are mandatory.
- The data types supported by NEF complies with the 3GPP specifications. For more information about the 3GPP data types, see [3GPP Technical Specification 29.571, Release 16, Common Data Types for Service Based Interfaces](#).

Table 2-53 Supported Response Codes

Code	Description
200	OK (Successful update of the subscription)
307	
308	

Table 2-53 (Cont.) Supported Response Codes

Code	Description
400	Bad Request
401	Unauthorized
403	Request is forbidden.
404	Not Found
411	
413	
415	Unsupported Media Type
429	
500	
503	Service Unavailable
default	

Examples

The following example shows how an AF updates details for a specific Traffic Influence from NEF by submitting a PUT request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request PUT 'http://<IP
address:Port>/3gpp-traffic-influence/v1/{afId}/subscriptions/
{subscriptionId}' \--header 'Content-
Type: application/json' \
```

Example of the Request Body

The following snippet shows an example of the request body.

```
{
  "afServiceId": "afServiceId1",
  "afTransId": "1524",
  "subscribedEvents": [
    "UP_PATH_CHANGE"
  ],
  "dnn": "s1",
  "ipv4Addr": "66.122.33.88",
  "dnaiChgType": "EARLY_LATE",
  "notificationDestination": "http://af_fqdn:1010/af/notification",
  "trafficFilters": [
    {
      "flowId": 4321,
      "flowDescriptions": [
        "UL"
      ]
    }
  ],
  "trafficRoutes": [
    {
      "dnai": "",
```

```

        "routeInfo": {
            "ipv4Addr": "10.10.10.10",
            "portNumber": 5050
        },
        "routeProfId": "profile2"
    }
]
}

```

Example of the Response Body

```

{
    "afServiceId": "afServiceId1",
    "afAppId": null,
    "afTransId": "1524",
    "appReloInd": null,
    "snssai": null,
    "externalGroupId": null,
    "anyUeInd": null,
    "subscribedEvents": [
        "UP_PATH_CHANGE"
    ],
    "gpsi": null,
    "ipv4Addr": null,
    "ipDomain": null,
    "dnn": "s1",
    "ipv4Addr": "66.122.33.88",
    "macAddr": null,
    "dnaiChgType": "EARLY_LATE",
    "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/
notification",
    "requestTestNotification": null,
    "websocketNotifConfig": null,
    "self": "http://ocnef-ext-nef-ingress-gateway:80/3gpp-traffic-
influence/v1/d2ad29d1-8c8e-4fb2-872e-937cc34bda53/subscriptions/
58fc8fa5-2868-4623-9e58-6aae8763fb22",
    "ethTrafficFilters": null,
    "trafficFilters": [
        {
            "flowId": 4321,
            "flowDescriptions": [
                "UL"
            ]
        }
    ],
    "trafficRoutes": [
        {
            "dnai": "",
            "routeInfo": {
                "ipv4Addr": "10.10.10.10",
                "portNumber": 5050
            },
            "routeProfId": "profile2"
        }
    ]
}

```

```
"tfcCorrInd": null,  
"tempValidities": null,  
"validGeoZoneIds": null,  
"afAckInd": null,  
"addrPreserInd": null,  
"suppFeat": null  
}
```

Note

The notificationDestination link is described as follows:

`http://<AF Hostname>:<AF Port>/<any path supported by AF for notifications>`

2.1.9.4 Update (Patch) Traffic Influence Subscription

This service operation is used to partially update or replace an existing resource of a Traffic Influence subscription.

Type: PATCH

URI: /3gpp-traffic-influence/v1/{afId}/subscriptions/{subscriptionId}

Initiated By: AF

Table 2-54 Path Parameters

Field Name	Data Type	Description
afId	string	A unique identifier of the AF.
subscriptionId	string	Identifier of the subscription resource.

Note

- All the path parameters in the above table are mandatory."
- The data types supported by NEF complies with the 3GPP specifications. For more information about the 3GPP data types, see [3GPP Technical Specification 29.571, Release 16, Common Data Types for Service Based Interfaces](#).

Table 2-55 Supported Response Codes

Code	Description
200	OK. The subscription was modified successfully.
307	
308	
400	Bad Request
401	Unauthorized

Table 2-55 (Cont.) Supported Response Codes

Code	Description
403	Request is forbidden.
404	Not Found
411	
413	
415	Unsupported Media Type
429	
500	
503	Service Unavailable
default	

Examples

The following example shows how an AF updates details for a specific Traffic Influence from NEF by submitting a PUT request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request PATCH 'http://<IP
address:Port>/3gpp-traffic-influence/v1/{afId}/subscriptions/
{subscriptionId}' --header 'Content-
Type: application/merge-patch+json' \
```

Example of the Request Body

The following snippet shows an example of the request body.

```
{
  "tfcCorrInd": false,
  "trafficRoutes": [
    {
      "dnai": "",
      "routeInfo": {
        "ipv4Addr": "170.122.123.45",
        "portNumber": 6060
      },
      "routeProfId": "PAT1"
    }
  ]
}
```

Example of the Response Body

```
{
  "afServiceId": "afServiceId1",
  "afAppId": null,
  "afTransId": "1524",
  "appReloInd": null,
  "dnn": null,
  "snssai": null,
```

```

    "externalGroupId": null,
    "anyUeInd": null,
    "subscribedEvents": [
        "UP_PATH_CHANGE"
    ],
    "gpsi": null,
    "ipv4Addr": null,
    "ipDomain": null,
    "ipv6Addr": "2001:1234:3333:4444:5555:6666:7777:8888",
    "macAddr": null,
    "dnaiChgType": "EARLY_LATE",
    "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/
notification",
    "requestTestNotification": null,
    "websockNotifConfig": null,
    "self": "http://ocnef-ext-nef-ingress-gateway:80/3gpp-traffic-
influence/v1/d2ad29d1-8c8e-4fb2-872e-937cc34bda53/subscriptions/
58fc8fa5-2868-4623-9e58-6aae8763fb22",
    "trafficFilters": [
        {
            "flowId": 4321,
            "flowDescriptions": [
                "UL"
            ]
        }
    ],
    "ethTrafficFilters": null,
    "trafficRoutes": [
        {
            "dnai": "",
            "routeInfo": {
                "ipv4Addr": "170.122.123.45",
                "ipv6Addr": null,
                "portNumber": 6060
            },
            "routeProfId": "PAT1"
        }
    ],
    "tfcCorrInd": false,
    "tempValidities": null,
    "validGeoZoneIds": null,
    "afAckInd": null,
    "addrPreserInd": null,
    "suppFeat": null
}

```

① Note

The notificationDestination link is described as follows:

http://<AF Hostname>:<AF Port>/<any path supported by AF for notifications>

2.1.9.5 Delete Traffic Influence Subscription

This service operation is used to delete an existing Traffic Influence subscription.

Type: DELETE

URI: /3gpp-traffic-influence/v1/{afId}/subscriptions/{subscriptionId}

Initiated By: AF

Table 2-56 Path Parameters

Field Name	Data Type	Description
afId	string	A unique identifier of the AF.
subscriptionId	string	Identifier of the subscription resource.

Note

- All the path parameters in the above table are mandatory.
- The data types supported by NEF complies with the 3GPP specifications. For more information about the 3GPP data types, see [3GPP Technical Specification 29.571, Release 16, Common Data Types for Service Based Interfaces](#).

Table 2-57 Supported Response Codes

Code	Description
204	No Content (Successful deletion of the existing subscription)
307	
308	
400	Bad Request
401	Unauthorized
403	Request is forbidden.
404	Not Found
429	
500	
503	Service Unavailable
default	

Examples

The following example shows how an AF deletes a specific Traffic Influence subscription from NEF by submitting a DELETE request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request DELETE 'http://<IP  
address:Port>/3gpp-traffic-influence/v1/{afId}/subscriptions/  
{subscriptionId}' --header 'Content-  
Type: application/json' \
```

2.1.9.6 Traffic Influence Notification

This service operation is used to notify about the Traffic Influence subscriptions.

Type: POST

URI: /5gc-agent/v1/TI/<encoded-notification-receivingservice-url>

Initiated By: AF

Table 2-58 Supported Response Codes

Code	Description
204	No Content (successful notification)
307	
308	
400	Bad Request
401	Unauthorized
403	Request is forbidden.
404	Not Found
411	
413	
415	Unsupported Media Type
429	
500	
503	Service Unavailable
default	

Examples

The following example shows how the AF notifies about Traffic Influence by submitting a POST request on the REST resource using cURL.

cURL Command

```
curl --http2-prior-knowledge --location --request POST '/5gc-agent/v1/
<translator>/<encoded-notification-receivingservice-
url>' --header 'Content-
Type: application/json'
```

Example:

```
curl --http2-prior-knowledge --location --request POST '<IP Address:Port>/
5gc-agent/v1/TI/b2NuZWYtdHJhZmZpY2luZmx1ZW5jZTo4MDgw' --header 'Content-
Type: application/json'
```

```
{
  "notifId": "82b5f3f9-bb5d-4f61-adfd-58c1e0b3f486",
  "eventNotifs": [
    {
      "event": "UP_PATH_CH",
```

```

"timeStamp": "2022-12-30T06:30:00Z",
"supi": "string",
"gsi": "string",
"sourceDnai": "string",
"targetDnai": "string",
"dnaiChgType": "EARLY_LATE",
"sourceUeIpv4Addr": "0.0.0.0",
"sourceUeIpv6Prefix": "string",
"targetUeIpv4Addr": "0.0.0.0",
"targetUeIpv6Prefix": "string",
"sourceTraRouting": {
  "dnai": "",
  "routeInfo": {
    "ipv4Addr": "0.0.0.0",
    "ipv6Addr": "",
    "portNumber": 1
  },
  "routeProfId": ""
},
"targetTraRouting": {
  "dnai": "",
  "routeInfo": {
    "ipv4Addr": "0.0.0.0",
    "ipv6Addr": "",
    "portNumber": 1
  },
  "routeProfId": "profile1"
},
"ueMac": "00-0C-29-6C-F3-E5",
"adIpv4Addr": "0.0.0.0",
"adIpv6Prefix": "string",
"reIpv4Addr": "0.0.0.0",
"reIpv6Prefix": "string",
"plmnId": {
  "mcc": 333,
  "mnc": 22
},
"accType": "3GPP_ACCESS",
"pduSeId": 2,
"dddStatus": "BUFFERED",
"dddTraDescriptor": {
  "ipv4Addr": "0.0.0.0",
  "ipv6Addr": "",
  "portNumber": 10,
  "macAddr": "00-0C-29-6C-F3-E5"
},
"maxWaitTime": "2022-12-30T06:30:00Z",
"ipv4Addr": "0.0.0.0",
"ipv6Prefixes": [
  "string"
],
"ipv6Addrs": [
  "string"
],
"pduSessType": "IPV4",

```

```

    "qfi": 1,
    "appId": "string",
    "ethfDescs": [
      {
        "ethType": ""
      }
    ],
    "fDescs": [
      "string"
    ],
    "dnn": "string",
    "snssai": {
      "sst": 1,
      "sd": "A12345"
    },
    "ulDelays": [
      1,2
    ],
    "dlDelays": [
      1,2
    ],
    "rtDelays": [
      1,2
    ]
  }
],
"ackUri": ""
}

```

2.1.10 Device Trigger Transaction

Using Device Triggering, applications notify User Equipment (UE) to perform application-specific tasks such as establishing communication with applications, changing device settings, and so on. Device triggering is required when an IP address for the UE is unavailable or unreachable by an application.

This mechanism enhances NEF functionality by introducing new APIs that allow AFs to remotely trigger specific actions on devices within a 5G network.

2.1.10.1 Create Device Trigger Transaction

This service operation is used to create a new Device Trigger transaction.

Type: POST

URI: /3gpp-device-triggering/v1/{afId}/transactions

Initiated By: AF

Table 2-59 Request Body Parameters

Field Name	Data Type	Description
self	string	This is the link to the Device Trigger transaction resource. This parameter is supplied by the NEF in HTTP responses.

Table 2-59 (Cont.) Request Body Parameters

Field Name	Data Type	Description
externalId	string	Unique identifier of a user as defined. It is a string containing a local identifier followed by "@" and a domain identifier.
msisdn	string	This allows to identify the MSISDN number of a UE.
supportedFeatures	string	This attribute is provided in the POST request and in the response of creating a resource.
validityPeriod	integer	The validity time for a specific task.
priority	string	Identifies the priority of the device trigger.
applicationPortId	integer	This is used to uniquely identify the triggering application addressed in the device for destination port. It is an unsigned integer with valid values between 0 and 65535.
appSrcPortId	integer	This is used to uniquely identify the triggering application addressed in the device for originator port. It is an unsigned integer with valid values between 0 and 65535.
triggerPayload	string	The device triggering payload.
notificationDestination	string	A URI indicating the notification destination for T8 notifications.
deliveryResult	string	The delivery result shall be included in the HTTP responses that indicate the delivery status of the device triggering.

Example:

The following example shows how an AF creates a Device Trigger request by submitting a POST request using cURL.

cURL Command:

```
curl --http2-prior-knowledge --location --request POST 'http://
<IPaddress:Port>/3gpp-device-triggering/v1/afID1000/transactions' \-
header 'Content-Type: application/json' \ -H "accept: application/json
-H 'authorization:
Bearer
eyJ0eXAiOiJKV1QiLCJpbnZva2VyTmFtZSI6IjE1NjUtZmZmLWFSbDQtYWZmMiIsImNhcnGlmSW5zdG
FuY2VJZCI6ImY5ODg4YmZkMS1jZTg1LTRmNTQtODUwYS0wZTk5MTQ2OWM5NmMiLCJzY29wZSI6Ij
In0.eyJpc3MiOiIyODg4YmZkMS1jZTg1LTRmNTQtODUwYS0wZTk5MTQ2OWM5NmMiLCJzY29wZSI6Ij
NncHAjNDJhZTNhYzAtYTlhNy00MjQ3LTlkOGUyY2YyMGQzZDUzMGU2OjNncHAtYXN0c2VzZ2lubi13
aXRoLXFcycwzZ3BwLWRldm1jZS10cm1nZ2VyaW5nLDNncHAtbW9uaXRvcmluZy1ldmVudCwzZ3BwLX
RyYWZmaW5mbHVlbnNlIiwiaXhwIjoxNjkyMjc5NDcyfQ.oisNp8fbVucsfVecEhVWXXALGxJen
NjhFuYJ-0CPYmW0XF9LWkvDhUknTdjxgIwnDxwntm_4-
p9kdhDKX7R1NHe2y2R4EtystLp-8Z8NsgMat_TuCMnx_sBn1NFxQYmGxVABiWJzKZMAWAInH1AGJqX
0LEKoAottV2dq8aVYh-Ji55LKX0ysfL001BjTCIf2mRXTLDDPbI-zPv6c2TzbI-
swrNMLRsmAv6piSeRPsGwtzR8QdIz5P5HJq_QwEpn0AKh6qZWUKBUmvodpVEw8uZEK0SJValHInB3q
5o7jFhjvFx69uk-kCzKM3D0GdfYA_mxN948F1F76mepkGI-DTA'
```

Table 2-60 Supported Response Codes

Code	Description
201	The subscription resource is created successfully.
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
411	Content length required
413	Payload large
415	Unsupported Media Type
500	Internal Server Error
503	Service Unavailable

Example of the Request Body for Device Trigger Transaction

The following snippet shows an example of the request body.

```
{
  "msisdn": "998796775",
  "validityPeriod": 60,
  "priority": "PRIORITY",
  "applicationPortId": 4248,
  "appSrcPortId": 33238,
  "triggerPayload": "START_APP1_SESSION",
  "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/
notification"
}
```

Example of the Response Body

```
{
  "self": "http://ocnef-ingressgateway:80/3gpp-device-triggering/v1/
afID1000/transactions/e3253426-bca9-4c84-a7f7-2dc4b5e616bb",
  "msisdn": "998796775",
  "validityPeriod": 60,
  "priority": "PRIORITY",
  "applicationPortId": 4248,
  "appSrcPortId": 33238,
  "triggerPayload": "START_APP1_SESSION",
  "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/
notification"
}
```

2.1.10.2 Get Individual Device Trigger Transaction

This service operation is used to get a existing Device Trigger transaction.

Type: GET

URI: /3gpp-device-triggering/v1/{afId}/transactions/{transactionId}

Initiated By: AF

Example:

The following example shows how an AF retrieves an existing device trigger by submitting a GET request using cURL.

cURL Command:

```
curl --http2-prior-knowledge --location --request GET 'http://
<IPaddress:Port>/3gpp-device-triggering/v1/afID1000/transactions/
8834804d-1338-477b-b43a-6d40a233bb3d HTTP/2' \--header 'Content-Type:
application/json' \ -H 'authorization:
Bearer
eyJ0eXAiOiJKV1QiLCJpbnZva2VyTmFtZSI6IjE1NjUtZmZmLWFSbDQtYWZmMiIsImNhcGlmSW5zdG
FuY2VJZCI6ImY5OGIwNWUwLTlYzdGtMTFlZC04NjFkLTAYNDJhYzEyMDAwMiIsImFsZyI6IjE1JTMjU2
In0.eyJpc3MiOiIyODg4YmZkMS1jZTg1LTRmNTQtODUwYS0wZTk5MTQ2OWM5NmMiLCJzY29wZSI6Ij
NncHAjNDJhZTNhYzAtYTlhNy00MjQ3LTlkOGEtY2QyMGQzZDUzMGE20jNncHAtYXMtc2Vzc2lubi13
aXRoLXFvcywzZ3BwLWRldm1jZS10cm1nZ2VyaW5nLDNncHAtbW9uaXRvcmluZy1ldmVudCwzZ3BwLX
RyYWZmawMtaw5mbHVlbmNlIiwizXhwIjoxNjkyMjcyNDcyfQ.oisNp8fbVucsfVecEhVWXXALGxJeN
NjhFuYJ-0CPYmWOXF9LWkvDhUknTdjxgIwnDxwntm_4-
p9kdhDKX7R1NHe2y2R4EtystLp-8Z8NsgMat_TuCMnx_sBn1NFxQYmGxVABiWJzKZMAWAInHlAGJqX
0lEKoAottV2dq8aVYh-Ji55LKX0ysfL001BjTCIf2mRXtLDDPbI-zPv6c2TzbI-
swrNMLRsmAv6piSeRPsGwtzR8QdIz5P5HJq_QwEpn0AKh6qZWUKBUmvodpVEw8uZEK0SJValHInB3q
507jFhjvFx69uk-kCzKM3D0GDFYA_mxN948F1F76mepkGI-DTA'
```

Table 2-61 Supported Response Codes

Code	Description
200	OK (Successful get the active subscription)
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "self": "http://ocnef-ingressgateway:80/3gpp-device-triggering/v1/
afID1000/transactions/e3253426-bca9-4c84-a7f7-2dc4b5e616bb",
  "msisdn": "998796775",
  "validityPeriod": 60,
  "priority": "PRIORITY",
  "applicationPortId": 4248,
  "appSrcPortId": 33238,
  "triggerPayload": "START_APP1_SESSION",
  "notificationDestination": "http://ocnefsim-ocstub-svc-af:1010/af/
notification"
}
```

2.1.10.3 Delete Individual Device Trigger Transaction

This service operation is used to delete an existing Device Trigger transaction.

Type: DELETE

URI: /3gpp-device-triggering/v1/{afId}/transactions/{transactionId}

Initiated By: AF

Example:

The following example shows how an AF deletes an existing Device Trigger by submitting a DELETE request using cURL.

cURL Command:

```
curl --http2-prior-knowledge --location --request DELETE 'http://
<IPaddress:Port>/3gpp-device-triggering/v1/afID1000/transactions/
8834804d-1338-477b-b43a-6d40a233bb3d HTTP/2' \--header 'Content-Type:
application/json' \ -H 'authorization:
Bearer
eyJ0eXAiOiJKV1QiLCJpbnZva2VyTmFtZSI6IjE1NjUtZmZmLWFSbDQtYWZmMiIsImNhcGlmSW5zdG
FuY2VJZCI6ImY5OGIwNWUwLTlYzDgtMTFlZC04NjFkLTAYNDJhYzEyMDAwMiIsImFsZyI6IjTMjU2
In0.eyJpc3MiOiIyODg4YmZkMS1jZTg1LTRmNTQtODUwYS0wZTk5MTQ2OWM5NmMiLCJzY29wZSI6Ij
NncHAjNDJhZTNhYzAtYTlhNy00MjQ3LTlkOGEtY2QyMGQzZDUzMGZ20jNncHAtYXNtc2Vzc2lubi13
aXR0LXFvcyZ3BwLWRldm1jZS10cm1nZ2VyaW5nLDNncHAtbW9uaXRvcmluZy1ldmVudCwzZ3BwLX
RyYWZmaW5mbHVlbnNlIiwiaXhwIjoxNjkyMjc5NDcyfQ.oisNp8fbVucsfVecEhVWXXALGxJeN
NjhFuYJ-0CPYmW0XF9LWkvDhUknTdjxgIwnDxwntm_4-
p9kdhdKX7R1NHe2y2R4EtystLp-8Z8NsgMat_TuCMnx_sBn1NFxQYmGxVABiWJzKZMAWAInH1AGJqX
0LEKoAottV2dq8aVYh-Ji55LKX0ysfL001BjTCIf2mRXTLDDPbI-zPv6c2TzbI-
swrNMLRsmAv6piSeRPsGwtzR8QdIz5P5HJq_QwEpn0AKh6qZWUKBUmvodpVEw8uZEK0SJValHInB3q
507jFhjvFx69uk-kCzKM3D0GDFYA_mxN948F1F76mepkGI-DTA'
```

Table 2-62 Supported Response Codes

Code	Description
200	OK
204	No Content
400	Bad Request
401	Unauthorized
403	Forbidden
404	Not Found
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Delivery Report Notification

SMS-SC reports the success or failure of delivering the device trigger to the UE. Report notification (Delivery Report Request - diameter message) on the T4 interface to NEF.

Example notification sent by NEF incase of a successful delivery report notification received from SMS-SC .

```
{
  transaction: "http://ocnef-ingressgateway:80/3gpp-device-triggering/v1/
afID1000/transactions/e3253426-bca9-4c84-a7f7-2dc4b5e616bb", "
  result: "SUCCESS"
}
```

2.1.11 CNC Console Integration

The integration of NEF or CAPIF with the CNC Console enables you to configure and modify different services and features using the CNC Console.

Note

During installation, all the configurations would be read from Helm. Any configurations that support **Update** operation through REST API can only be updated using REST API or Console and any further updates to these configurations using Helm would be ignored. For further information, refer to *Oracle Communications Cloud Native Core, Network Exposure Function REST Specification Guide* and *Configuring Network Exposure Function using the CNC Console* chapter in *Oracle Communications Cloud Native Core, Network Exposure Function User Guide*.

2.1.11.1 Configuring NEF Features

This section provides information about enabling NEF features.

Note

apiroot should point to fivegc-ingress-gateway service.

2.1.11.1.1 Fetching Logging Data

This service operation is used to fetch logging data.

Type: GET

URI: {apiRoot}/nef/nf-common-component/v1/{serviceName}/logging

Initiated By: CNCC

Table 2-63 Response Body Parameters

Field Name	Data Type	Description
appLogLevel	string	This parameter sets the application log level.
packageLogLevel	JSON	This parameter sets the package log level.

Table 2-64 packageLogLevel Parameters

Field Name	Data Type	Description
packageName	string	Name of the package for which log level has to be changed.
logLevelForPackage	string	This is the value of log level.

Example:

The following example shows how logging data is fetched by a GET request.

```
{apiRoot}/nef/nf-common-component/v1/monitoringevents/logging
```

Table 2-65 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

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

2.1.11.1.2 Updating Logging Data

This service operation is used to update logging data.

Type: PUT

URI: {apiRoot}/nef/nf-common-component/v1/{serviceName}/logging

Initiated By: CNCC

Example:

The following example shows how logging data is updated by a PUT request.

```
{apiRoot}/nef/nf-common-component/v1/monitoringevents/logging
```

Example of the Request Body

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

Example of the Response Body

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

2.1.11.1.3 Fetch Timeout Data

This service operation is used to fetch timeout data.

Type: GET

URI: {apiRoot}/nef/nf-common-component/v1/{serviceName}/timeout

Initiated By: CNCC

Table 2-66 Response Body Parameters

Field Name	Data Type	Description
serviceType	string	Name of the NEF service.
requesttimeout	string	Actual value of request timeout.

Example:

The following example shows how timeout data is fetched by a GET request.

```
{apiRoot}/nef/nf-common-component/v1/monitoringevents/timeout
```

Table 2-67 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden

Table 2-67 (Cont.) Supported Response Codes

Code	Description
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "serviceType": "monitoringevents",
  "requesttimeout": "12000"
}
```

2.1.11.1.4 Fetch GMLC Configuration

This service operation is used to fetch GMLC configuration.

Type: GET

URI: {apiRoot}/nef/nf-common-component/v1/monitoringevents/gmlcconfigs

Initiated By: CNCC

Table 2-68 Response Body Parameters

Field Name	Data Type	Description
gmlcHaccuracy	string	Specifies the minimum horizontal accuracy.
gmlcVaccuracy	string	Specifies the minimum vertical accuracy.
reportingInterval	integer	The location reporting interval in seconds.
destIfLocQosAbsent	enum	Indicates the location provider when LocQos is not present in the subscription request. Possible values: • udm • gmlc Default: udm
switchOnErrorCodes.codeCause	string	Contains a list of error codes and causes for which the failover must be performed. Value to be supplied should be concatenation of errorcode and errorcause, separated by "--". Example: 503 -- Connection Refused Where, errorCode: 503 errorCause: Connection Refused

Table 2-68 (Cont.) Response Body Parameters

Field Name	Data Type	Description
explicitCancellation	boolean	Indicates whether an explicit cancellation request should be sent to GMLC. It is sent: - when the maximumNumberOfReports are received from GMLC. - when Delete Subscription request is received from AF based on the GMLC initiated Cancel Location request.
switchToUdmOnFailure	enum	Enables the failover to UDM, when GMLC subscription fails for a specific reason. The location reporting event request is sent to UDM when NEF receives failure from GMLC. Possible values are: - ALL: Enables the failover for subscription of both immediate and deferred requests. - IMMEDIATE: Enables the failover for subscription of immediate location requests. - DEFERRED: Enables the failover for subscription of deferred location requests. - NONE: Disables the failover. Default: NONE

Example:

The following example shows how GMLC configuration is fetched by a GET request.

```
{apiRoot}/nef/nf-common-component/v1/monitoringevents/gmlcconfigs
```

Table 2-69 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "gmlcEnabled": true,
  "gmlcHaccuracy": 10.0,
  "gmlcVaccuracy": 10.0,
  "reportingInterval": 3600,
  "destIfLocQosAbsent": "gmlc",
  "switchOnErrorCodes": [
    {
```

```

        "codeCause": " 503 ~~ Connection Refused "
      },
      {
        "codeCause": " 503 ~~ Unknown Host Exception "
      }
    ],
    "explicitCancellation": false,
    "switchToUdmOnFailure": "ALL"
  }
}

```

2.1.11.1.5 Update GMLC Configuration

This service operation is used to update GMLC configuration.

Type: PUT

URI: {apiRoot}/nef/nf-common-component/v1/monitoringevents/gmlcconfigs

Initiated By: CNCC

Example:

The following example shows how GMLC configuration is fetched by a PUT request.

{apiRoot}/nef/nf-common-component/v1/monitoringevents/gmlcconfigs

Example of the Request Body

```

{
  "gmlcEnabled": true,
  "gmlcHaccuracy": 12.0,
  "gmlcVaccuracy": 12.0,
  "reportingInterval": 3600,
  "destIfLocQosAbsent": "gmlc",
  "switchOnErrorCodes": [
    {
      "codeCause": " 503 ~~ Connection Refused "
    },
    {
      "codeCause": " 503 ~~ Unknown Host Exception "
    }
  ],
  "explicitCancellation": false,
  "switchToUdmOnFailure": "ALL"
}

```

Note

The gmlcEnabled parameter cannot be modified, it can be enabled or disabled using only helm upgrades.

Example of the Response Body

```
{
  "gmlcEnabled": true,
  "gmlcHaccuracy": 12.0,
  "gmlcVaccuracy": 12.0,
  "reportingInterval": 3600,
  "destIfLocQosAbsent": "gmlc",
  "switchOnErrorCodes": [
    {
      "codeCause": " 503 ~~ Connection Refused "
    },
    {
      "codeCause": " 503 ~~ Unknown Host Exception "
    }
  ],
  "explicitCancellation": false,
  "switchToUdmOnFailure": "ALL"
}
```

2.1.11.1.6 Fetch ShortCodeNotification Configuration

This service operation is used to fetch ShortCodeNotification configuration.

Type: GET

URI: {apiRoot}/nef/common-component/v1/msisdlessmosms/
shortcodenotificationconfig

Initiated By: CNCC

Example:

The following example shows how ShortCodeNotification configuration is fetched by a GET request.

{apiRoot}/nef/common-component/v1/msisdlessmosms/shortcodenotificationconfig

Table 2-70 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "shortcode": "31628870634",
```

```

    "notificationUrl": "http://ocnefsim-ocstub-svc-af:1010/af/notification"
  },
  {
    "shortcode": "9876543210",
    "notificationUrl": "http://ocnefsim-ocstub-svc-af:1010/af/notification"
  }
}

```

2.1.11.1.7 Update ShortCodeNotification Configuration

This service operation is used to update ShortCodeNotification configuration.

Type: PATCH

URI: {apiRoot}/nef/common-component/v1/msisdnllessmosms/
shortcodenotificationconfig/{shortcode}

Initiated By: CNCC

Table 2-71 Response Body Parameters

Field Name	Data Type	Description
shortcode	string	Value of Short Code
notificationUrl	string	Notification URL associated

Example:

The following example shows how ShortCodeNotification configuration is update by a PATCH request.

```
{apiRoot}/nef/common-component/v1/msisdnllessmosms/shortcodenotificationconfig/
31628870634
```

Table 2-72 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```

{
  "shortcode": "9876543210",
  "notificationUrl": "http://ocnefsim-ocstub-svc-af:1010/af/notification"
}

```

Example of the Response Body

```
{
  "shortcode": "31628870634",
  "notificationUrl": "http://ocnefsim-ocstub-svc-af:1010/af/notification"
}
```

2.1.11.1.8 Fetch QoS Configuration

This service operation is used to fetch QoS configuration.

Type: GET

URI: {apiRoot}/nef/nf-common-component/v1/qualityofservice/qosconfig

Initiated By: CNCC

Table 2-73 Request Body Parameters

Field Name	Data Type	Description
switchOnErrorCodes	Array	Error code configuration
convergedPeCharging	Boolean	Flag to enable converged PE charging
directPCRFFlowEnabled	Boolean	Flag to enable PCRF flow
maxQosReferencesAllowed	Integer	Maximum value of allowed QOS reference
switchToPCRFOnAuthFailure	Boolean	Flag to switch towards PCRF on Auth Failure

Table 2-74 switchOnErrorCodes

Field Name	Data Type	Description
code	String	HTTP code value
cause	String	Cause of error in problem details

Example:

The following example shows how QoS configuration is fetched by a GET request.

{apiRoot}/nef/nf-common-component/v1/qualityofservice/qosconfig

Table 2-75 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "switchOnErrorCodes": [
    {
      "code": "404",
      "cause": "Not Found"
    }
  ],
  "convergedPeCharging": false,
  "directPCRFFlowEnabled": false,
  "maxQosReferencesAllowed": 2,
  "switchToPCRFOnAuthFailure": true
}
```

2.1.11.1.9 Update QoS Configuration

This service operation is used to update QoS configuration.

Type: PUT

URI: {apiRoot}/nef/nf-common-component/v1/qualityofservice/qosconfig

Initiated By: CNCC

Example:

The following example shows how QoS configuration is updated by a PUT request.

{apiRoot}/nef/nf-common-component/v1/qualityofservice/qosconfig

Table 2-76 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "switchToPCRFOnAuthFailure": true,
  "convergedPeCharging": false,
  "directPCRFFlowEnabled": false,
  "maxQosReferencesAllowed": 2,
  "switchOnErrorCodes": [
    {
      "code": "404",
      "cause": "Not Found"
    }
  ]
}
```

```

    }
  ]
}

```

Example of the Response Body

```

{
  "switchToPCRFOnAuthFailure": true,
  "convergedPeCharging": false,
  "directPCRFFlowEnabled": false,
  "maxQosReferencesAllowed": 2,
  "switchOnErrorCodes": [
    {
      "code": "404",
      "cause": "Not Found"
    }
  ]
}

```

2.1.11.1.10 Fetch Global Configurations

This service operation is used to fetch global configurations.

Type: GET

URI: {apiRoot}/nef/nf-common-component/v1/globaloptions

Initiated By: CNCC

Table 2-77 Response Body Parameters

Field Name	Data Type	Description
nefSiteId	String	UUID of NEF Site
nfInstanceId	String	NF Instance ID of NEF
geoRedundancy	JSON	Geo-redundancy related configuration of NEF

Table 2-78 afServiceIdList Parameters

Field Name	Data Type	Description
afServiceId	String	AF Service ID
dnn	String	Data Network Name
snssai	JSON	Single Network Slice selection assistance information
routeToLoc	Array	Route to Location Configuration

Table 2-79 snssai Parameters

Field Name	Data Type	Description
sd	String	Service Differentiator
sst	Integer	Slice Service Type

Table 2-80 routeToLoc Parameters

Field Name	Data Type	Description
dnai	String	Data Network Access identifier
routeInfo	JSON	Route Information
routeProfileId	String	Route Profile ID

Table 2-81 routeInfo Parameters

Field Name	Data Type	Description
ipv4Addr	String	IPv4 Address
ipv6Addr	String	IPv6 Address
portNumber	Integer	Port Number

Table 2-82 scsShortMessageEntity Parameters

Field Name	Data Type	Description
address	String	Address
typeOfNumber	String	Type of number
numberingPlan	String	Numbering Plan

Table 2-83 qosProfiles Parameters

Field Name	Data Type	Description
mediaType	String	Media Type
maxReqBwDI	String	
maxReqBwUI	String	
minReqBwDI	String	
minReqBwUI	String	
qosReferenceId	String	QOS reference ID

Table 2-84 plmnIdMap Parameters

Field Name	Data Type	Description
plmnId	String	Public Land Mobile Network ID
mnc	String	Mobile Network Code
mcc	String	Mobile Country Code

Table 2-85 taiMap Parameters

Field Name	Data Type	Description
taId	String	Tracking Area ID
plmnId	String	Public Land Mobile Network ID. Note: This value should be pre-configured in plmnIdMap.

Table 2-85 (Cont.) taiMap Parameters

Field Name	Data Type	Description
tac	String	Tracking Area Code
nid	String	Network ID

Table 2-86 ecgiMap Parameters

Field Name	Data Type	Description
ecgild	String	EUTRA Cell Global Identifier
plmnId	String	Public Land Mobile Network ID. Note: This value should be pre-configured in plmnIdMap.
eutraCellId	String	E-UTRAN Cell Identifier
nid	String	Network ID

Table 2-87 ncgiMap Parameters

Field Name	Data Type	Description
ncgild	String	New Radio Cell Global Identifier
plmnId	String	Public Land Mobile Network ID. Note: This value should be pre-configured in plmnIdMap.
nrCellId	String	NR Cell Identity
nid	String	Network ID

Table 2-88 gNbIdMap Parameters

Field Name	Data Type	Description
gNbId	String	Next Generation Node B ID
gNBValue	String	GNB Value
bitLength	String	Bit length

Table 2-89 globalRanNodeIdMap Parameters

Field Name	Data Type	Description
nid	String	Network ID
eNbId	String	evolved NodeB ID
gNbId	String	Next Generation Node B ID Note: This value should be preconfigured in gNbIdMap.
plmnId	String	Public Land Mobile Network ID. Note: This value should be pre-configured in plmnIdMap.
tngfId	String	Trusted non-3GPP Gateway Function ID
wagfId	String	W-AGF identifier
n3IwfId	String	N3 IWF identifier

Table 2-89 (Cont.) globalRanNodeIdMap Parameters

Field Name	Data Type	Description
ngeNbId	String	NG-eNodeB identifier
ranNodeId	String	RAN Node ID

Table 2-90 geoZoneIdToSpatialValidityMap Parameters

Field Name	Data Type	Description
zone	String	Zone ID
prald	String	PRA ID
ecgiList	String	EUTRA Cell Global Identifier Note: This value should be pre-configured in ecgiMap.
ncgiList	String	New Radio Cell Global Identifier Note: This value should be pre-configured in ncgiMap.
presenceState	String	Allowed values are IN_AREA, OUT_OF_AREA, UNKNOWN, and INACTIVE
additionalPrald	String	Additional PRA ID
globaleNbIdList	String	Next Generation Node B ID Note: This value should be pre-configured in gNbIdMap.
trackingAreaList	String	Tracking Area ID Note: This value should be pre-configured in taiMap.
globalRanNodeIdList	String	Next Generation Node B ID Note: This value should be pre-configured in gNbIdMap.

Table 2-91 geoRedundancy Parameters

Field Name	Data Type	Description
featureStatus	String	Flag to enable or disable geo-redundancy
peerGRSiteList	Array	Configuration of NEF peer sites
handleNotification	JSON	Notification handler

Table 2-92 peerGRSiteList Parameters

Field Name	Data Type	Description
siteName	String	Site Name peer NEF
siteInstanceId	String	Site Instance ID of peer NEF

Table 2-93 handleNotification Parameters

Field Name	Data Type	Description
checkSiteStatus	Boolean	Flag to check peer site status

Table 2-93 (Cont.) handleNotification Parameters

Field Name	Data Type	Description
checkDBReplicationStatus	Boolean	Flag to check DB replication status

Example:

The following example shows how global configurations is fetched by a GET request.

```
{apiRoot}/nef/nf-common-component/v1/globaloptions
```

Table 2-94 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "taiMap": [
    {
      "nid": "",
      "tac": "121",
      "taiId": "taiId1",
      "plmnId": "plmnId1"
    }
  ],
  "ecgiMap": [
    {
      "nid": "",
      "ecgiId": "ecgiId1",
      "plmnId": "plmnId1",
      "eutraCellId": "23"
    }
  ],
  "ncgiMap": [
    {
      "nid": "",
      "ncgiId": "ncgiId1",
      "plmnId": "plmnId1",
      "nrCellId": "3"
    }
  ],
  "gNbIdMap": [
```

```

{
  "gNbId": "gNbId1",
  "gNBValue": "1345",
  "bitLength": "23"
},
{
  "nefSiteId": "9faf1bbc-6e4a-4454-a507-aef01a101a27",
  "plmnIdMap": [
    {
      "mcc": "234",
      "mnc": "123",
      "plmnId": "plmnId1"
    }
  ],
  "qosProfiles": [
    {
      "mediaType": "AUDIO",
      "maxReqBwDl": "",
      "maxReqBwUl": "",
      "minReqBwDl": "",
      "minReqBwUl": "",
      "qosReferenceId": "qosReferenceId1"
    },
    {
      "mediaType": "VIDEO",
      "maxReqBwDl": "",
      "maxReqBwUl": "",
      "minReqBwDl": "",
      "minReqBwUl": "",
      "qosReferenceId": "qosReferenceId2"
    }
  ],
  "nfInstanceId": "9faf1bbc-6e4a-4454-a507-aef01a101a27",
  "geoRedundancy": {
    "featureStatus": "ENABLED",
    "peerGRSiteList": [
      {
        "siteName": "Site-2",
        "siteInstanceId": "6faf1bbc-6e4a-4454-a507-a14ef8e1bc5d"
      }
    ],
    "handleNotification": {
      "checkSiteStatus": true,
      "checkDBReplicationStatus": true
    }
  },
  "afServiceIdList": [
    {
      "dnn": "dnnForServiceId1",
      "snssai": {
        "sd": "123456",
        "sst": 123
      }
    }
  ],

```

```

        "routeToLoc": [
        {
            "dnai": "dnnaiForServiceId1",
            "routeInfo":
            {
                "ipv4Addr": "127.0.0.1",
                "ipv6Addr": "0:0:0:0:0:0:0:1",
                "portNumber": 8080
            },
            "routeProfId": "rpiForServiceId1"
        }
        ],
        "afServiceId": "afServiceId1"
    },
    {
        "dnn": "dnnForServiceId2",
        "snssai":
        {
            "sd": "123456",
            "sst": 123
        },
        "routeToLoc": [
        {
            "dnai": "dnnaiForServiceId2",
            "routeInfo":
            {
                "ipv4Addr": "127.0.0.1",
                "ipv6Addr": "0:0:0:0:0:0:0:1",
                "portNumber": 8080
            },
            "routeProfId": "rpiForServiceId2"
        }
        ],
        "afServiceId": "afServiceId2"
    }
    ],
    "enabledFeatures":
    {
        "gmlc": false,
        "deviceTrigger": false,
        "msisdnllessmosms": false,
        "convergedScefNef": true,
        "monitoringEvents": true,
        "qualityOfService": true,
        "trafficInfluence": true
    },
    "globalRanNodeIdMap": [
    {
        "nid": "",
        "eNbId": "",
        "gNbId": "gNbId1",
        "plmnId": "plmnId1",
        "tngfId": "",
        "wagfId": "",
        "n3IwfId": "",
        "ngeNbId": "",
    }
    ]
}

```

```

        "ranNodeId": "ranNodeId1"
    },
    ],
    "scsShortMessageEntity":
    {
        "address": 123499789,
        "typeOfNumber": "NETWORK_SPECIFIC_NUMBER",
        "numberingPlan": "PRIVATE_NUMBERING_PLAN"
    },
    "geoZoneIdToSpatialValidityMap": [
    {
        "zone": "zone1",
        "praId": "",
        "ecgiList": [
            "ecgiId1"
        ],
        "ncgiList": [
            "ncgiId1"
        ],
        "presenceState": "",
        "additionalPraId": "",
        "globaleNbIdList": [],
        "trackingAreaList": [
            "taiId1"
        ],
        "globalRanNodeIdList": [
            "ranNodeId1"
        ]
    }
    ]
}

```

2.1.11.1.11 Fetch AF Service ID List

This service operation is used to fetch AF service ID list.

Type: GET

URI: {apiRoot}/nef/common-component/v1/global/afServiceIdList

Initiated By: CNCC

Example:

The following example shows how AF service ID list is fetched by a GET request.

{apiRoot}/nef/common-component/v1/global/afServiceIdList

Table 2-95 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found

Table 2-95 (Cont.) Supported Response Codes

Code	Description
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "dnn": "dnnForServiceId1",
  "snssai":
  {
    "sd": "123456",
    "sst": 123
  },
  "routeToLoc": [
    {
      "dnai": "dnaiForServiceId1",
      "routeInfo":
      {
        "ipv4Addr": "127.0.0.1",
        "ipv6Addr": "0:0:0:0:0:0:0:1",
        "portNumber": 8080
      },
      "routeProfId": "rpiForServiceId1"
    }
  ],
  "afServiceId": "afServiceId1"
},
{
  "dnn": "dnnForServiceId2",
  "snssai":
  {
    "sd": "123456",
    "sst": 123
  },
  "routeToLoc": [
    {
      "dnai": "dnaiForServiceId2",
      "routeInfo":
      {
        "ipv4Addr": "127.0.0.1",
        "ipv6Addr": "0:0:0:0:0:0:0:1",
        "portNumber": 8080
      },
      "routeProfId": "rpiForServiceId2"
    }
  ],
  "afServiceId": "afServiceId2"
}
```

2.1.11.1.12 Add Service ID List

This service operation is used to add Service ID List.

Type: POST

URI: {apiRoot}/nef/common-component/v1/global/afServiceIdList/{afServiceId}

Initiated By: CNCC

Table 2-96 Response Body Parameters

Field Name	Data Type	Description
afServiceId	String	AF Service ID
dnn	String	Data Network Name
snssai	JSON	Single Network Slice selection assistance Information
routeToLoc	Array	Route to Location Configuration

Table 2-97 snssai Parameters

Field Name	Data Type	Description
sd	String	Service Differentiator
sst	Integer	Slice Service Type

Table 2-98 routeToLoc Parameters

Field Name	Data Type	Description
dnai	String	Data Network Access identifier
routeInfo	JSON	Route Information
routeProfileId	String	Route Profile ID

Table 2-99 routeInfo Parameters

Field Name	Data Type	Description
ipv4Addr	String	IPv4 address
ipv6Addr	String	IPv6 address
portNumber	Integer	Port Number

Example:

The following example shows how AF service ID list is added by a PUT request.

```
{apiRoot}/nef/common-component/v1/global/afServiceIdList/334
```

Table 2-100 Supported Response Codes

Code	Description
200	OK

Table 2-100 (Cont.) Supported Response Codes

Code	Description
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "afServiceId": "334",
  "dnn": "44",
  "snssai":
  {
    "sst": 4,
    "sd": "123456"
  },
  "routeToLoc": []
}
```

Example of the Response Body

```
{
  "afServiceId": "334",
  "dnn": "44",
  "snssai":
  {
    "sst": 4,
    "sd": "123456"
  },
  "routeToLoc": []
}
```

2.1.11.1.13 Edit Service ID List

This service operation is used to edit Service ID List.

Type: PATCH

URI: {apiRoot}/nef/common-component/v1/global/afServiceIdList/{afServiceId}

Initiated By: CNCC

Table 2-101 Response Body Parameters

Field Name	Data Type	Description
afServiceId	String	AF Service ID

Table 2-101 (Cont.) Response Body Parameters

Field Name	Data Type	Description
dnn	String	Data Network Name
snssai	JSON	Single Network Slice selection assistance Information
routeToLoc	Array	Route to Location Configuration

Table 2-102 snssai Parameters

Field Name	Data Type	Description
sd	String	Service Differentiator
sst	Integer	Slice Service Type

Table 2-103 routeToLoc Parameters

Field Name	Data Type	Description
dnai	String	Data Network Access identifier
routeInfo	JSON	Route Information
routeProfileId	String	Route Profile ID

Table 2-104 routeInfo Parameters

Field Name	Data Type	Description
ipv4Addr	String	IPv4 address
ipv6Addr	String	IPv6 address
portNumber	Integer	Port Number

Example:

The following example shows how AF service ID list is edited by a PATCH request.

```
{apiRoot}/nef/common-component/v1/global/afServiceIdList/afServiceId1
```

Table 2-105 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "afServiceId": "afServiceId1",
  "dnn": "dnnForServiceId1",
  "snssai":
  {
    "sst": 123,
    "sd": "133456"
  },
  "routeToLoc": [
    {
      "dnai": "dnnaiForServiceId1",
      "routeInfo":
      {
        "ipv4Addr": "127.0.0.1",
        "ipv6Addr": "0:0:0:0:0:0:0:1",
        "portNumber": 8080
      },
      "routeProfId": "rpiForServiceId1"
    }
  ]
}
```

Example of the Response Body

```
{
  "afServiceId": "afServiceId1",
  "dnn": "dnnForServiceId1",
  "snssai":
  {
    "sst": 123,
    "sd": "133456"
  },
  "routeToLoc": [
    {
      "dnai": "dnnaiForServiceId1",
      "routeInfo":
      {
        "ipv4Addr": "127.0.0.1",
        "ipv6Addr": "0:0:0:0:0:0:0:1",
        "portNumber": 8080
      },
      "routeProfId": "rpiForServiceId1"
    }
  ]
}
```

2.1.11.1.14 Delete Service ID List

This service operation is used to delete AF service ID list.

Type: DELETE

URI: {apiRoot}/nef/common-component/v1/global/afServiceIdList/{afServiceId}

Initiated By: CNCC

Example:

The following example shows how AF service ID list is deleted by a DELETE request.

```
{apiRoot}/nef/common-component/v1/global/afServiceIdList/334
```

Table 2-106 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

2.1.11.1.15 Fetch SCS Short Message Entity

This service operation is used to fetch SCS short message entity.

Type: GET

URI: {apiRoot}/nef/nf-common-component/v1/global/scsShortMessageEntity

Initiated By: CNCC

Example:

The following example shows how SCS short message entity is fetched by a GET request.

```
{apiRoot}/nef/nf-common-component/v1/global/scsShortMessageEntity
```

Table 2-107 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{  
  "address": "1234997899",
```

```
"typeOfNumber": "INTERNATIONAL_NUMBER",  
"numberingPlan": "PRIVATE_NUMBERING_PLAN"  
}
```

2.1.11.1.16 Update SCS Short Message Entity

This service operation is used to update SCS short message entity.

Type: PUT

URI: {apiRoot}/nef/nf-common-component/v1/global/scsShortMessageEntity

Initiated By: CNCC

Table 2-108 Response Body Parameters

Field Name	Data Type	Description
address	String	Address
typeOfNumber	String	Type of number
numberingPlan	String	Numbering plan

Example:

The following example shows how SCS short message entity is updated by a PUT request.

```
{apiRoot}/nef/nf-common-component/v1/global/scsShortMessageEntity
```

Table 2-109 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{  
  "address": "1234997899",  
  "typeOfNumber": "INTERNATIONAL_NUMBER",  
  "numberingPlan": "PRIVATE_NUMBERING_PLAN"  
}
```

Example of the Response Body

```
{  
  "address": "1234997899",  
  "typeOfNumber": "INTERNATIONAL_NUMBER",  
}
```

```
    "numberingPlan": "PRIVATE_NUMBERING_PLAN"
  }
```

2.1.11.1.17 Fetch QoS Profiles

This service operation is used to fetch QoS Profiles.

Type: GET

URI: {apiRoot}/nef/common-component/v1/global/qosProfiles

Initiated By: CNCC

Table 2-110 Response Body Parameters

Field Name	Data Type	Description
mediaType	String	Media Type
maxReqBwDl	String	
maxReqBwUl	String	
minReqBwDl	String	
minReqBwUl	String	
qosReferenceId	String	QoS reference ID

Example:

The following example shows how QoS profiles is fetched by a GET request.

```
{apiRoot}/nef/common-component/v1/global/qosProfiles
```

Table 2-111 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
[
  {
    "mediaType": "AUDIO",
    "maxReqBwDl": "120 Gbps",
    "maxReqBwUl": "10 Gbps",
    "minReqBwDl": "100 Mbps",
    "minReqBwUl": "50 Mbps",
    "qosReferenceId": "qosReferenceId1"
  },
]
```

```

    {
      "mediaType": "VIDEO",
      "maxReqBwDl": "300 Gbps",
      "maxReqBwUl": "340 Gbps",
      "minReqBwDl": "200 Gbps",
      "minReqBwUl": "1 Tbps",
      "qosReferenceId": "qosReferenceId2"
    }
  ]

```

2.1.11.1.18 Update QoS Profiles

This service operation is used to update QoS Profiles.

Type: PATCH

URI: {apiRoot}/nef/common-component/v1/global/qosProfiles/{qosProfilesId}

Initiated By: CNCC

Table 2-112 Response Body Parameters

Field Name	Data Type	Description
mediaType	String	Media Type
maxReqBwDl	String	
maxReqBwUl	String	
minReqBwDl	String	
minReqBwUl	String	
qosReferenceId	String	QoS reference ID

Example:

The following example shows how QoS profiles are updated by a PATCH request.

```
{apiRoot}/nef/common-component/v1/global/qosProfiles/qosReferenceId1
```

Table 2-113 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```

{
  "qosReferenceId": "qosReferenceId1",

```

```

    "mediaType": "AUDIO",
    "maxReqBwDl": "100 bps",
    "maxReqBwUl": "20 bps",
    "minReqBwDl": "10 bps",
    "minReqBwUl": "10 bps"
  }

```

Example of the Response Body

```

{
  "qosReferenceId": "qosReferenceId1",
  "mediaType": "AUDIO",
  "maxReqBwDl": "100 bps",
  "maxReqBwUl": "20 bps",
  "minReqBwDl": "10 bps",
  "minReqBwUl": "10 bps"
}

```

2.1.11.1.19 Add QoS Profiles

This service operation is used to add QoS Profiles.

Type: POST

URI: {apiRoot}/nef/common-component/v1/global/qosProfiles/{qosProfilesId}

Initiated By: CNCC

Table 2-114 Response Body Parameters

Field Name	Data Type	Description
mediaType	String	Media Type
maxReqBwDl	String	
maxReqBwUl	String	
minReqBwDl	String	
minReqBwUl	String	
qosReferenceId	String	QoS reference ID

Example:

The following example shows how QoS profiles are added by a POST request.

```
{apiRoot}/nef/common-component/v1/global/qosProfiles/qosReferenceId3
```

Table 2-115 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable

Table 2-115 (Cont.) Supported Response Codes

Code	Description
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "qosReferenceId": "qosReferenceId3",
  "mediaType": "AUDIO",
  "maxReqBwDl": "100 bps",
  "maxReqBwUl": "20 bps",
  "minReqBwDl": "10 bps",
  "minReqBwUl": "10 bps"
}
```

Example of the Response Body

```
{
  "qosReferenceId": "qosReferenceId3",
  "mediaType": "AUDIO",
  "maxReqBwDl": "100 bps",
  "maxReqBwUl": "20 bps",
  "minReqBwDl": "10 bps",
  "minReqBwUl": "10 bps"
}
```

2.1.11.1.20 Delete QoS Profiles

This service operation is used to delete QoS Profiles.

Type: DELETE

URI: {apiRoot}/nef/common-component/v1/global/qosProfiles/{qosProfilesId}

Initiated By: CNCC

Example:

The following example shows how QoS profiles are deleted by a DELETE request.

{apiRoot}/nef/common-component/v1/global/qosProfiles/qosReferenceId3

Table 2-116 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found

Table 2-116 (Cont.) Supported Response Codes

Code	Description
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

2.1.11.1.21 Fetch PLMN ID Map

This service operation is used to fetch PLMN ID Map.

Type: GET

URI: {apiRoot}/nef/common-component/v1/global/plmnIdMap

Initiated By: CNCC

Table 2-117 Response Body Parameters

Field Name	Data Type	Description
plmnId	String	Public Land Mobile Network ID
mnc	String	Mobile Network Code
mcc	String	Mobile Country Code

Example:

The following example shows how PLMN ID Map is fetched by a GET request.

```
{apiRoot}/nef/common-component/v1/global/plmnIdMap
```

Table 2-118 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{  
  "mcc": "234",  
  "mnc": "123",  
  "plmnId": "plmnId1"  
}
```

2.1.11.1.22 Add PLMN ID Map

This service operation is used to add PLMN ID Map.

Type: POST

URI: {apiRoot}/nef/common-component/v1/global/plmnIdMap/{plmnId}

Initiated By: CNCC

Table 2-119 Response Body Parameters

Field Name	Data Type	Description
plmnId	String	Public Land Mobile Network ID
mnc	String	Mobile Network Code
mcc	String	Mobile Country Code

Example:

The following example shows how PLMN ID Map is added by a POST request.

```
{apiRoot}/nef/common-component/v1/global/plmnIdMap/plmn2
```

Table 2-120 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "mcc": "234",
  "mnc": "123",
  "plmnId": "plmn2"
}
```

Example of the Response Body

```
{
  "mcc": "234",
  "mnc": "123",
  "plmnId": "plmn2"
}
```

2.1.11.1.23 Edit PLMN ID Map

This service operation is used to edit PLMN ID Map.

Type: PATCH

URI: {apiRoot}/nef/common-component/v1/global/plmnIdMap/{plmnId}

Initiated By: CNCC

Table 2-121 Response Body Parameters

Field Name	Data Type	Description
plmnId	String	Public Land Mobile Network ID
mnc	String	Mobile Network Code
mcc	String	Mobile Country Code

Example:

The following example shows how PLMN ID Map is edited by a PATCH request.

```
{apiRoot}/nef/common-component/v1/global/plmnIdMap/plmn2
```

Table 2-122 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{  
  "mcc": "234",  
  "mnc": "122",  
  "plmnId": "plmn2"  
}
```

Example of the Response Body

```
{  
  "mcc": "234",  
  "mnc": "122",  
  "plmnId": "plmn2"  
}
```

2.1.11.1.24 Delete PLMN ID Map

This service operation is used to delete PLMN ID Map.

Type: DELETE

URI: {apiRoot}/nef/common-component/v1/global/plmnIdMap/{plmnId}

Initiated By: CNCC

Table 2-123 Response Body Parameters

Field Name	Data Type	Description
plmnId	String	Public Land Mobile Network ID
mnc	String	Mobile Network Code
mcc	String	Mobile Country Code

Example:

The following example shows how PLMN ID Map is deleted by a DELETE request.

```
{apiRoot}/nef/common-component/v1/global/plmnIdMap/plmn2
```

Table 2-124 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

2.1.11.1.25 Fetch TAI Map

This service operation is used to fetch TAI Map.

Type: GET

URI: {apiRoot}/nef/common-component/v1/global/taiMap

Initiated By: CNCC

Table 2-125 Response Body Parameters

Field Name	Data Type	Description
taid	String	Tracking Area ID

Table 2-125 (Cont.) Response Body Parameters

Field Name	Data Type	Description
plmnId	String	Public Land Mobile Network ID Note: This value should be pre-configured in plmnIdMap.
tac	String	Tracking Area Code
nid	String	Network ID

Example:

The following example shows how TAI Map is fetched by a GET request.

```
{apiRoot}/nef/common-component/v1/global/taiMap
```

Table 2-126 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "nid": "",
  "tac": "121",
  "taiId": "taiId1",
  "plmnId": "plmnId1"
}
```

2.1.11.1.26 Add to TAI MAP

This service operation is used to add TAI Map.

Type: POST

URI: {apiRoot}/nef/common-component/v1/global/taiMap/{taiId}

Initiated By: CNCC

Table 2-127 Response Body Parameters

Field Name	Data Type	Description
taid	String	Tracking Area ID

Table 2-127 (Cont.) Response Body Parameters

Field Name	Data Type	Description
plmnId	String	Public Land Mobile Network ID Note: This value should be pre-configured in plmnIdMap.
tac	String	Tracking Area Code
nid	String	Network ID

Example:

The following example shows how TAI Map is added by a POST request.

```
{apiRoot}/nef/common-component/v1/global/taiMap/2343
```

Table 2-128 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "taiId": "2343",
  "plmnId": "plmnId1",
  "tac": "4",
  "nid": ""
}
```

Example of the Response Body

```
{
  "taiId": "2343",
  "plmnId": "plmnId1",
  "tac": "4",
  "nid": ""
}
```

2.1.11.1.27 Edit TAI MAP

This service operation is used to edit TAI Map.

Type: PATCH

URI: {apiRoot}/nef/common-component/v1/global/taiMap/{taiId}

Initiated By: CNCC

Table 2-129 Response Body Parameters

Field Name	Data Type	Description
taId	String	Tracking Area ID
plmnId	String	Public Land Mobile Network ID Note: This value should be pre-configured in plmnIdMap.
tac	String	Tracking Area Code
nid	String	Network ID

Example:

The following example shows how TAI Map is edited by a PATCH request.

{apiRoot}/nef/common-component/v1/global/taiMap/2343

Table 2-130 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "taiId": "2343",
  "plmnId": "plmnId1",
  "tac": "44",
  "nid": ""
}
```

Example of the Response Body

```
{
  "taiId": "2343",
  "plmnId": "plmnId1",
  "tac": "44",
  "nid": ""
}
```

2.1.11.1.28 Delete from TAI Map

This service operation is used to delete TAI Map.

Type: DELETE

URI: {apiRoot}/nef/common-component/v1/global/taiMap/{taiId}

Initiated By: CNCC

Example:

The following example shows how TAI Map is deleted by a DELETE request.

{apiRoot}/nef/common-component/v1/global/taiMap/2343

Table 2-131 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

2.1.11.1.29 Fetch ECGI Map

This service operation is used to fetch ECGI Map.

Type: GET

URI: {apiRoot}/nef/common-component/v1/global/ecgiMap

Initiated By: CNCC

Table 2-132 Response Body Parameters

Field Name	Data Type	Description
ecgild	String	EUTRA Cell Global Identifier
plmnId	String	Public Land Mobile Network ID Note: This value should be pre-configured in plmnIdMap.
eutraCellId	String	E-UTRAN Cell Identifier
nid	String	Network ID

Example:

The following example shows how ECGI Map is fetched by a GET request.

{apiRoot}/nef/common-component/v1/global/ecgiMap

Table 2-133 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "nid": "",
  "ecgiId": "ecgiId1",
  "plmnId": "plmnId1",
  "eutraCellId": "23"
}
```

2.1.11.1.30 Add to ECGI Map

This service operation is used to add ECGI Map.

Type: POST

URI: {apiRoot}/nef/common-component/v1/global/ecgiMap/{ecgiID}

Initiated By: CNCC

Table 2-134 Response Body Parameters

Field Name	Data Type	Description
ecgild	String	EUTRA Cell Global Identifier
plmnId	String	Public Land Mobile Network ID Note: This value should be pre-configured in plmnIdMap.
eutraCellId	String	E-UTRAN Cell Identifier
nid	String	Network ID

Example:

The following example shows how ECGI Map is added by a POST request.

```
{apiRoot}/nef/common-component/v1/global/ecgiMap/334
```

Table 2-135 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "ecgiId": "334",
  "plmnId": "plmnId1",
  "eutraCellId": "44",
  "nid": ""
}
```

Example of the Response Body

```
{
  "ecgiId": "334",
  "plmnId": "plmnId1",
  "eutraCellId": "44",
  "nid": ""
}
```

2.1.11.1.31 Edit ECGI Map

This service operation is used to edit ECGI Map.

Type: PATCH

URI: {apiRoot}/nef/common-component/v1/global/ecgiMap/{ecgiID}

Initiated By: CNCC

Table 2-136 Response Body Parameters

Field Name	Data Type	Description
ecgild	String	EUTRA Cell Global Identifier
plmnId	String	Public Land Mobile Network ID Note: This value should be pre-configured in plmnIdMap.
eutraCellId	String	E-UTRAN Cell Identifier
nid	String	Network ID

Example:

The following example shows how ECGI Map is edited by a PATCH request.

```
{apiRoot}/nef/common-component/v1/global/ecgiMap/334
```

Table 2-137 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "ecgiId": "334",
  "plmnId": "plmnId1",
  "eutraCellId": "45",
  "nid": ""
}
```

Example of the Response Body

```
{
  "ecgiId": "334",
  "plmnId": "plmnId1",
  "eutraCellId": "45",
  "nid": ""
}
```

2.1.11.1.32 Delete from ECGI Map

This service operation is used to delete ECGI Map.

Type: DELETE

URI: {apiRoot}/nef/common-component/v1/global/ecgiMap/{ecgiID}

Initiated By: CNCC

Example:

The following example shows how ECGI Map is deleted by a DELETE request.

```
{apiRoot}/nef/common-component/v1/global/ecgiMap/334
```

Table 2-138 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

2.1.11.1.33 Fetch NCGI Map

This service operation is used to fetch NCGI Map.

Type: GET

URI: {apiRoot}/nef/common-component/v1/global/ncgiMap

Initiated By: CNCC

Table 2-139 Response Body Parameters

Field Name	Data Type	Description
ncgild	String	New Radio Cell Global Identifier
plmnId	String	Public Land Mobile Network ID Note: This value should be preconfigured in plmnIdMap.
nrCellId	String	NR Cell Identity
nid	String	Network ID

Example:

The following example shows how NCGI Map is fetched by a GET request.

```
{apiRoot}/nef/common-component/v1/global/ncgiMap
```

Table 2-140 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "nid": "",
  "ncgiId": "ncgiId1",
  "plmnId": "plmnId1",
  "nrCellId": "3"
}
```

2.1.11.1.34 Add to NCGI Map

This service operation is used to add NCGI Map.

Type: POST

URI: {apiRoot}/nef/common-component/v1/global/ncgiMap/{ncgiId}

Initiated By: CNCC

Table 2-141 Response Body Parameters

Field Name	Data Type	Description
ncgild	String	New Radio Cell Global Identifier
plmnId	String	Public Land Mobile Network ID Note: This value should be preconfigured in plmnIdMap.
nrCellId	String	NR Cell Identity
nid	String	Network ID

Example:

The following example shows how NCGI Map is added by a POST request.

```
{apiRoot}/nef/common-component/v1/global/ncgiMap/233
```

Table 2-142 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "ncgiId": "233",
  "plmnId": "plmnId1",
}
```

```

    "nrCellId": "4",
    "nid": ""
  }

```

Example of the Response Body

```

{
  "ncgiId": "233",
  "plmnId": "plmnId1",
  "nrCellId": "4",
  "nid": ""
}

```

2.1.11.1.35 Edit NCGI Map

This service operation is used to edit NCGI Map.

Type: PATCH

URI: {apiRoot}/nef/common-component/v1/global/ncgiMap/{ncgiId}

Initiated By: CNCC

Table 2-143 Response Body Parameters

Field Name	Data Type	Description
ncgild	String	New Radio Cell Global Identifier
plmnId	String	Public Land Mobile Network ID Note: This value should be preconfigured in plmnIdMap.
nrCellId	String	NR Cell Identity
nid	String	Network ID

Example:

The following example shows how NCGI Map is edited by a PATCH request.

```
{apiRoot}/nef/common-component/v1/global/ncgiMap/233
```

Table 2-144 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "ncgiId": "233",
  "plmnId": "plmnId1",
  "nrCellId": "44",
  "nid": ""
}
```

Example of the Response Body

```
{
  "ncgiId": "233",
  "plmnId": "plmnId1",
  "nrCellId": "44",
  "nid": ""
}
```

2.1.11.1.36 Delete from NCGI Map

This service operation is used to delete NCGI Map.

Type: DELETE

URI: {apiRoot}/nef/common-component/v1/global/ncgiMap/{ncgiId}

Initiated By: CNCC

Example:

The following example shows how NCGI Map is deleted by a DELETE request.

{apiRoot}/nef/common-component/v1/global/ncgiMap/233

Table 2-145 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

2.1.11.1.37 Fetch GNBID Map

This service operation is used to fetch GNBID Map.

Type: GET

URI: {apiRoot}/nef/common-component/v1/global/gNbIdMap

Initiated By: CNCC

Table 2-146 Response Body Parameters

Field Name	Data Type	Description
gNbId	String	Next Generation Node B ID
gNBValue	String	GNB value
bitLength	String	Bit length

Example:

The following example shows how GNBID Map is fetched by a GET request.

```
{apiRoot}/nef/common-component/v1/global/gNbIdMap
```

Table 2-147 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "gNbId": "gNbId1",
  "gNBValue": "1345",
  "bitLength": "23"
}
```

2.1.11.1.38 Add to GNBID Map

This service operation is used to add GNBID Map.

Type: POST

URI: {apiRoot}/nef/common-component/v1/global/gNbIdMap/{gNbId}

Initiated By: CNCC

Table 2-148 Response Body Parameters

Field Name	Data Type	Description
gNbId	String	Next Generation Node B ID
gNBValue	String	GNB value
bitLength	String	Bit length

Example:

The following example shows how GNBID Map is added by a POST request.

```
{apiRoot}/nef/common-component/v1/global/gNbIdMap/33
```

Table 2-149 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "gNbId": "33",
  "bitLength": "24"
  "gNBValue": "44",
}
```

Example of the Response Body

```
{
  "gNbId": "33",
  "bitLength": "24"
  "gNBValue": "44",
}
```

2.1.11.1.39 Edit GNBID Map

This service operation is used to edit GNBID Map.

Type: PATCH

URI: {apiRoot}/nef/common-component/v1/global/gNbIdMap/{gNbId}

Initiated By: CNCC

Table 2-150 Response Body Parameters

Field Name	Data Type	Description
gNbId	String	Next Generation Node B ID
gNBValue	String	GNB value
bitLength	String	Bit length

Example:

The following example shows how GNBID Map is edited by a PATCH request.

```
{apiRoot}/nef/common-component/v1/global/gNbIdMap/33
```

Table 2-151 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "gNbId": "33",
  "bitLength": "24"
  "gNBValue": "45",
}
```

Example of the Response Body

```
{
  "gNbId": "33",
  "bitLength": "24"
  "gNBValue": "45",
}
```

2.1.11.1.40 Delete from GNBID Map

This service operation is used to delete GNBID Map.

Type: DELETE

URI: {apiRoot}/nef/common-component/v1/global/gNbIdMap/{gNbId}

Initiated By: CNCC

Example:

The following example shows how GNBID Map is deleted by a DELETE request.

```
{apiRoot}/nef/common-component/v1/global/gNbIdMap/33
```

Table 2-152 Supported Response Codes

Code	Description
200	OK

Table 2-152 (Cont.) Supported Response Codes

Code	Description
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

2.1.11.1.41 Fetch globalRanNodeId Map

This service operation is used to fetch globalRanNodeId Map.

Type: GET

URI: {apiRoot}/nef/common-component/v1/global/globalRanNodeIdMap

Initiated By: CNCC

Table 2-153 Response Body Parameters

Field Name	Data Type	Description
eNbId	String	evolved NodeB ID
gNbId	String	Next Generation Node B ID Note: This value should be preconfigured in gNbIdMap.
n3IwfId	String	N3 IWF identifier
ngeNbId	String	NG-eNodeB identifier
nId	String	Network ID
plmnId	String	Public Land Mobile Network ID Note: This value should be preconfigured in plmnIdMap.
ranNodeid	String	RAN Node ID
tngfId	String	Trusted non-3GPP Gateway Function ID
wagfId	String	W-AGF identifier

Example:

The following example shows how globalRanNodeid Map is fetched by a GET request.

{apiRoot}/nef/common-component/v1/global/globalRanNodeIdMap

Table 2-154 Supported Response Codes

Code	Description
200	OK
400	Bad Request

Table 2-154 (Cont.) Supported Response Codes

Code	Description
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "nid": "",
  "eNbId": "",
  "gNbId": "gNbId1",
  "plmnId": "plmnId1",
  "tngfId": "",
  "wagfId": "",
  "n3IwfId": "",
  "ngeNbId": "",
  "ranNodeId": "ranNodeId1"
}
```

2.1.11.1.42 Add to globalRanNodeId Map

This service operation is used to add globalRanNodeId Map.

Type: POST

URI: {apiRoot}/nef/common-component/v1/global/globalRanNodeIdMap/
{globalRanNodeId}

Initiated By: CNCC

Table 2-155 Response Body Parameters

Field Name	Data Type	Description
eNbId	String	evolved NodeB ID
gNbId	String	Next Generation Node B ID Note: This value should be preconfigured in gNbIdMap.
n3IwfId	String	N3 IWF identifier
ngeNbId	String	NG-eNodeB identifier
nid	String	Network ID
plmnId	String	Public Land Mobile Network ID Note: This value should be preconfigured in plmnIdMap.
ranNodeId	String	RAN Node ID
tngfId	String	Trusted non-3GPP Gateway Function ID
wagfId	String	W-AGF identifier

Example:

The following example shows how globalRanNodeId Map is added by a POST request.

```
{apiRoot}/nef/common-component/v1/global/globalRanNodeIdMap/33
```

Table 2-156 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "ranNodeId": "33",
  "plmnId": "plmnId1",
  "n3IwfId": "",
  "gNbId": "gNbId1",
  "ngeNbId": "",
  "wagfId": "",
  "tngfId": "",
  "nid": "",
  "eNbId": ""
}
```

Example of the Response Body

```
{
  "ranNodeId": "33",
  "plmnId": "plmnId1",
  "n3IwfId": "",
  "gNbId": "gNbId1",
  "ngeNbId": "",
  "wagfId": "",
  "tngfId": "",
  "nid": "",
  "eNbId": ""
}
```

2.1.11.1.43 Edit globalRanNodeId Map

This service operation is used to edit globalRanNodeId Map.

Type: PATCH

URI: {apiRoot}/nef/common-component/v1/global/globalRanNodeIdMap/{globalRanNodeId}

Initiated By: CNCC

Table 2-157 Response Body Parameters

Field Name	Data Type	Description
eNbId	String	evolved NodeB ID
gNbId	String	Next Generation Node B ID Note: This value should be preconfigured in gNbIdMap.
n3IwfId	String	N3 IWF identifier
ngeNbId	String	NG-eNodeB identifier
nId	String	Network ID
plmnId	String	Public Land Mobile Network ID Note: This value should be preconfigured in plmnIdMap.
ranNodeId	String	RAN Node ID
tngfId	String	Trusted non-3GPP Gateway Function ID
wagfId	String	W-AGF identifier

Example:

The following example shows how globalRanNodeId Map is edited by a PATCH request.

{apiRoot}/nef/common-component/v1/global/globalRanNodeIdMap/33

Table 2-158 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "ranNodeId": "33",
  "plmnId": "plmnId1",
  "n3IwfId": "1",
  "gNbId": "gNbId1",
  "ngeNbId": "",
  "wagfId": "",
  "tngfId": "",
  "nId": ""
}
```

```
"eNbId": ""
}
```

Example of the Response Body

```
{
  "ranNodeId": "33",
  "plmnId": "plmnId1",
  "n3IwfId": "1",
  "gNbId": "gNbId1",
  "ngeNbId": "",
  "wagfId": "",
  "tngfId": "",
  "nid": "",
  "eNbId": ""
}
```

2.1.11.1.44 Delete from globalRanNodeId Map

This service operation is used to delete globalRanNodeId Map.

Type: DELETE

URI: {apiRoot}/nef/common-component/v1/global/globalRanNodeIdMap/{globalRanNodeId}

Initiated By: CNCC

Example:

The following example shows how globalRanNodeId Map is deleted by a DELETE request.

```
{apiRoot}/nef/common-component/v1/global/globalRanNodeIdMap/33
```

Table 2-159 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

2.1.11.1.45 Fetch geoZoneIdToSpatialValidity Map

This service operation is used to fetch geoZoneIdToSpatialValidity Map.

Type: GET

URI: {apiRoot}/nef/common-component/v1/global/geoZoneIdToSpatialValidityMap

Initiated By: CNCC

Table 2-160 Response Body Parameters

Field Name	Data Type	Description
zone	String	Zone ID
prald	String	PRA ID
ecgiList	String	EUTRA Cell Global Identifier Note: This value should be preconfigured in ecgiMap.
ncgiList	String	New Radio Cell Global Identifier Note: This value should be preconfigured in ncgiMap.
presenceState	String	Allowed values are IN_AREA, OUT_OF_AREA, UNKNOWN, and INACTIVE
additionalPrald	String	Additional PRA ID
globaleNbldList	String	Next Generation Node B ID Note: This value should be preconfigured in gNbldMap.
trackingAreaList	String	Tracking Area ID Note: This value should be preconfigured in taiMap.
globalRanNodeBIdList	String	Next Generation Node B ID Note: This value should be preconfigured in gNbldMap.

Example:

The following example shows how geoZoneIdToSpatialValidity Map is fetched by a GET request.

```
{apiRoot}/nef/common-component/v1/global/geoZoneIdToSpatialValidityMap
```

Table 2-161 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "zone": "zone1",
  "praId": "",
  "ecgiList":
```

```

    [
        "ecgiId1",
        "334",
        "ecgid3"
    ],
    "ncgiList":
    [
        "ncgiId1",
        "233"
    ],
    "presenceState": "",
    "additionalPraId": "",
    "globaleNbIdList": [],
    "trackingAreaList":
    [
        "taiId1"
    ],
    "globalRanNodeIdList":
    [
        "ranNodeId1"
    ]
}

```

2.1.11.1.46 Add to geoZoneIdToSpatialValidity Map

This service operation is used to add geoZoneIdToSpatialValidity Map.

Type: POST

URI: {apiRoot}/nef/common-component/v1/global/geoZoneIdToSpatialValidityMap/
{geoZoneIdToSpatialValidityMap}

Initiated By: CNCC

Table 2-162 Response Body Parameters

Field Name	Data Type	Description
zone	String	Zone ID
praid	String	PRA ID
ecgiList	String	EUTRA Cell Global Identifier Note: This value should be preconfigured in ecgiMap.
ncgiList	String	New Radio Cell Global Identifier Note: This value should be preconfigured in ncgiMap.
presenceState	String	Allowed values are IN_AREA, OUT_OF_AREA, UNKNOWN, and INACTIVE
additionalPraId	String	Additional PRA ID
globaleNbIdList	String	Next Generation Node B ID Note: This value should be preconfigured in gNbIdMap.
trackingAreaList	String	Tracking Area ID Note: This value should be preconfigured in taiMap.

Table 2-162 (Cont.) Response Body Parameters

Field Name	Data Type	Description
globalRanNodeIdList	String	Next Generation Node B ID Note: This value should be preconfigured in gNbIdMap.

Example:

The following example shows how geoZoneIdToSpatialValidity Map is added by a POST request.

```
{apiRoot}/nef/common-component/v1/global/geoZoneIdToSpatialValidityMap/zone2
```

Table 2-163 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "zone": "zone2",
  "praId": "",
  "additionalPraId": "",
  "presenceState": "UNKNOWN",
  "trackingAreaList":
  [
    "taiId1"
  ],
  "ecgiList":
  [
    "ecgiId1"
  ],
  "ncgiList":
  [
    "ncgiId1"
  ],
  "globalRanNodeIdList":
  [
    "ranNodeId1"
  ],
  "globaleNbIdList":
  [
```

```

        "ranNodeId1"
    ]
}

```

2.1.11.1.47 Edit geoZoneIdToSpatialValidity Map

This service operation is used to edit geoZoneIdToSpatialValidity Map.

Type: PATCH

URI: {apiRoot}/nef/common-component/v1/global/geoZoneIdToSpatialValidityMap/{geoZoneIdToSpatialValidityMap}

Initiated By: CNCC

Table 2-164 Response Body Parameters

Field Name	Data Type	Description
zone	String	Zone ID
prald	String	PRA ID
ecgiList	String	EUTRA Cell Global Identifier Note: This value should be preconfigured in ecgiMap.
ncgiList	String	New Radio Cell Global Identifier Note: This value should be preconfigured in ncgiMap.
presenceState	String	Allowed values are IN_AREA, OUT_OF_AREA, UNKNOWN, and INACTIVE
additionalPrald	String	Additional PRA ID
globaleNbldList	String	Next Generation Node B ID Note: This value should be preconfigured in gNbldMap.
trackingAreaList	String	Tracking Area ID Note: This value should be preconfigured in taiMap.
globalRanNodeIdList	String	Next Generation Node B ID Note: This value should be preconfigured in gNbldMap.

Example:

The following example shows how geoZoneIdToSpatialValidity Map is edited by a PATCH request.

```
{apiRoot}/nef/common-component/v1/global/geoZoneIdToSpatialValidityMap/zone2
```

Table 2-165 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden

Table 2-165 (Cont.) Supported Response Codes

Code	Description
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "zone": "zone2",
  "praId": "1",
  "additionalPraId": "",
  "presenceState": "UNKNOWN",
  "trackingAreaList":
  [
    "taiId1"
  ],
  "ecgiList":
  [
    "ecgiId1"
  ],
  "ncgiList":
  [
    "ncgiId1"
  ],
  "globalRanNodeIdList":
  [
    "ranNodeId1"
  ],
  "globaleNbIdList":
  [
    "ranNodeId1"
  ]
}
```

Example of the Response Body

```
{
  "zone": "zone2",
  "praId": "1",
  "additionalPraId": "",
  "presenceState": "UNKNOWN",
  "trackingAreaList":
  [
    "taiId1"
  ],
  "ecgiList":
  [
    "ecgiId1"
  ]
}
```

```

    ],
    "ncgiList":
    [
        "ncgiId1"
    ],
    "globalRanNodeIdList":
    [
        "ranNodeId1"
    ],
    "globaleNbIdList":
    [
        "ranNodeId1"
    ]
}

```

2.1.11.1.48 Delete from geoZoneIdToSpatialValidity Map

This service operation is used to delete geoZoneIdToSpatialValidity Map.

Type: DELETE

URI: {apiRoot}/nef/common-component/v1/global/geoZoneIdToSpatialValidityMap/
{geoZoneIdToSpatialValidityMap}

Initiated By: CNCC

Example:

The following example shows how geoZoneIdToSpatialValidity Map is deleted by a DELETE request.

{apiRoot}/nef/common-component/v1/global/geoZoneIdToSpatialValidityMap/zone2

Table 2-166 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

2.1.11.1.49 Fetch Target NF Communication Profile Mapping

This service operation is used to fetch target NF communication profile mapping.

Type: GET

URI: {apiRoot}/nef/common-component/v1/fivegcagent/
targetNfCommunicationProfileMapping

Initiated By: CNCC

Table 2-167 Response Body Parameters

Field Name	Data Type	Description
nfType	String	NF Type
activeProfile	String	Active Profile Name

Example:

The following example shows how target NF communication profile mapping is fetched by a GET request.

```
{apiRoot}/nef/common-component/v1/fivegcagent/
targetNfCommunicationProfileMapping
```

Table 2-168 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
[
  {
    "nfType": "BSF",
    "activeProfile": "modelA"
  },
  {
    "nfType": "CHF",
    "activeProfile": "customCHFModelID"
  },
  {
    "nfType": "GMLC",
    "activeProfile": "customGMLCModelID"
  },
  {
    "nfType": "PCF",
    "activeProfile": "modelA"
  },
  {
    "nfType": "UDM",
    "activeProfile": "customUDMModelID"
  },
  {
    "nfType": "UDR",
```

```

        "activeProfile": "customUDRModelD"
    }
]

```

2.1.11.1.50 Fetch Communication Profiles

This service operation is used to fetch communication profiles.

Type: GET

URI: {apiRoot}/nef/common-component/v1/fivegcagent/communicationProfiles

Initiated By: CNCC

Table 2-169 Response Body Parameters

Field Name	Data Type	Description
discoveryHeaderParams	JSON	Discovery Headers Parameter details
communicationProfileName	String	Communication Profile Name
sendRoutingBindingHeader	Boolean	Flag to enable/disable Routing Binding Header
sendDiscoverHeaderInitMsg	Boolean	Flag to enable/disable Discovery Header init Message
sendDiscoverHeaderSubsMsg	Boolean	Flag to enable/disable Discovery Header Subs Message

Table 2-170 discoveryHeaderParams Parameters

Field Name	Data Type	Description
targetNfType	String	target NF type
discoveryServices	Array	List of Discovery Services
preferredLocality	String	Preferred Locality
supportedFeatures	String	Supported Features

Example:

The following example shows how communication profiles is fetched by a GET request.

```
{apiRoot}/nef/common-component/v1/fivegcagent/communicationProfiles
```

Table 2-171 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
[
  {
    "discoveryHeaderParams":
    {
      "targetNfType": "BSF",
      "discoveryServices": [
        "nbsf-management"
      ],
      "preferredLocality": null,
      "supportedFeatures": null
    },
    "communicationProfileName": "customBSFModelID",
    "sendRoutingBindingHeader": true,
    "sendDiscoverHeaderInitMsg": true,
    "sendDiscoverHeaderSubsMsg": true
  },
  {
    "discoveryHeaderParams":
    {
      "targetNfType": "CHF",
      "discoveryServices": [
        "nchf-convergedcharging"
      ],
      "preferredLocality": null,
      "supportedFeatures": null
    },
    "communicationProfileName": "customCHFModelID",
    "sendRoutingBindingHeader": true,
    "sendDiscoverHeaderInitMsg": true,
    "sendDiscoverHeaderSubsMsg": true
  },
  {
    "discoveryHeaderParams":
    {
      "targetNfType": "GMLC",
      "discoveryServices": [
        "ngmlc-loc"
      ],
      "preferredLocality": null,
      "supportedFeatures": null
    },
    "communicationProfileName": "customGMLCModelID",
    "sendRoutingBindingHeader": true,
    "sendDiscoverHeaderInitMsg": true,
    "sendDiscoverHeaderSubsMsg": true
  },
  {
    "discoveryHeaderParams":
    {
      "targetNfType": "PCF",
      "discoveryServices": [
        "npcf-policyauthorization"
      ],
      "preferredLocality": null,
```

```

        "supportedFeatures": null
    },
    "communicationProfileName": "customPCFModelID",
    "sendRoutingBindingHeader": true,
    "sendDiscoverHeaderInitMsg": true,
    "sendDiscoverHeaderSubsMsg": true
},
{
    "discoveryHeaderParams":
    {
        "targetNfType": "UDM",
        "discoveryServices": [
            "nudm-ee",
            "nudm-sdm",
            "nudm-uecm"
        ],
        "preferredLocality": null,
        "supportedFeatures": null
    },
    "communicationProfileName": "customUDMModelID",
    "sendRoutingBindingHeader": true,
    "sendDiscoverHeaderInitMsg": true,
    "sendDiscoverHeaderSubsMsg": true
},
{
    "discoveryHeaderParams":
    {
        "targetNfType": "UDR",
        "discoveryServices": [
            "nudr-dr"
        ],
        "preferredLocality": null,
        "supportedFeatures": null
    },
    "communicationProfileName": "customUDRModelID",
    "sendRoutingBindingHeader": true,
    "sendDiscoverHeaderInitMsg": true,
    "sendDiscoverHeaderSubsMsg": true
}
]

```

2.1.11.1.51 Fetch AF Config Map

This service operation is used to fetch AF Config Map.

Type: GET

URI: {apiRoot}/nef/apd-manager/v1/af/configmap/

Initiated By: CNCC

Example:

The following example shows how AF Config Map is fetched by a GET request.

```
{apiRoot}/nef/apd-manager/v1/af/configmap/
```

Table 2-172 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
[
  {
    "id": "afservice1",
    "config":
    {
      "dnn": "dan",
      "snssai":
      {
        "sst": 233,
        "sd": "FF0000"
      }
    }
  }
]
```

2.1.11.1.52 Update AF Config Map

This service operation is used to update AF Config Map.

Type: POST

URI: {apiRoot}/nef/apd-manager/v1/af/configmap/{afServiceId}

Initiated By: CNCC

Table 2-173 Response Body Parameters

Field Name	Data Type	Description
id	String	AF Service ID
config	JSON	DNN and SNSSAI configuration

Table 2-174 config Parameters

Field Name	Data Type	Description
dnn	String	Data Network Name

Table 2-174 (Cont.) config Parameters

Field Name	Data Type	Description
snssai	JSON	Single Network Slice selection assistance Information

Table 2-175 SNSSAI Parameters

Field Name	Data Type	Description
sst	Integer	Slice Service Type
sd	String	Service Differentiator

Example:

The following example shows how AF Config Map is updated by a POST request.

```
{apiRoot}/nef/apd-manager/v1/af/configmap/afservice1
```

Table 2-176 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{
  "id": "afservice1",
  "config":
  {
    "dnn": "dan",
    "snssai":
    {
      "sst": 233,
      "sd": "FF0000"
    }
  }
}
```

Example of the Response Body

```
{
  "id": "afservice1",
```

```
"config":
{
  "dnn": "dan",
  "snssai":
  {
    "sst": 233,
    "sd": "FF0000"
  }
}
```

2.1.11.2 Configuring CAPIF Features

This section provides information about enabling CAPIF features.

Note

apiroot should point to network-ingressgateway service.

2.1.11.2.1 Fetching Logging Data

This service operation is used to fetch logging data.

Type: GET

URI: {apiRoot}/capif/nf-common-component/v1/{servicename}/logging

Initiated By: CNCC

Table 2-177 Response Body Parameters

Field Name	Data Type	Description
appLogLevel	string	This parameter sets the application log level.
packageLogLevel	JSON	This parameter sets the package log level.

Table 2-178 packageLogLevel Parameters

Field Name	Data Type	Description
packageName	string	Name of the package for which log level has to be changed.
logLevelForPackage	string	This is the value of log level.

Example:

The following example shows how logging data is fetched by a GET request.

{apiRoot}/capif/nf-common-component/v1/afmgr/logging

Table 2-179 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

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

2.1.11.2.2 Updating Logging Data

This service operation is used to update logging data.

Type: PUT

URI: {apiRoot}/capif/nf-common-component/v1/{servicename}/logging

Initiated By: CNCC

Example:

The following example shows how logging data is updated by a PUT request.

{apiRoot}/capif/nf-common-component/v1/afmgr/logging

Example of the Request Body

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

Example of the Response Body

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

2.1.11.2.3 Fetching Published Services from CAPIF

This service operation is used to fetch published services.

Type: GET

URI: {apiRoot}/capif-configuration/v1/apimgr/nef_services

Initiated By: CNCC

Table 2-180 Response Body Parameters

Field Name	Data Type	Description
publishedService	string	Contains the String Concatenation of RegSec of NEF and the API Name of the service published from NEF to CAPIF separated by "::".

Example:

The following example shows how published services are fetched by a GET request.

{apiRoot}/capif-configuration/v1/apimgr/nef_services

Table 2-181 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "publishedService": "Site1RegSec::3gpp-as-session-with-qos"
},
{
```

```

    "publishedService": "Site1RegSec::3gpp-monitoring-event"
  },
  {
    "publishedService": "Site1RegSec::3gpp-traffic-influence"
  },
  {
    "publishedService": "Site2RegSec::3gpp-as-session-with-qos"
  },
  {
    "publishedService": "Site2RegSec::3gpp-monitoring-event"
  }
}

```

2.1.11.2.4 Create Discovery Group

This service operation is used to create discovery group.

Type: POST

URI: {apiRoot}/capif-configuration/v1/discovery-group/nefServicesGroup

Initiated By: CNCC

Table 2-182 Response Body Parameters

Field Name	Data Type	Description
groupId	String	The unique discovery group identifier.
nefServices	List of String	List of Regsec and apinames published to CAPIF (to be picked from the response of fetching published services from CAPIF).

Example:

The following example shows how discovery group are fetched by a GET request.

{apiRoot}/capif-configuration/v1/discovery-group/nefServicesGroup

Table 2-183 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```

{
  "groupId": "group4",
  "nefServices":

```

```
[
  "Site1RegSec::3gpp-as-session-with-qos",
  "Site1RegSec::3gpp-monitoring-event",
  "Site1RegSec::3gpp-traffic-influence",
  "Site2RegSec::3gpp-monitoring-event"
]
```

Example of the Response Body

```
{
  "groupId": "group4",
  "nefServices":
  [
    "Site1RegSec::3gpp-as-session-with-qos",
    "Site1RegSec::3gpp-monitoring-event",
    "Site1RegSec::3gpp-traffic-influence",
    "Site2RegSec::3gpp-monitoring-event"
  ]
}
```

2.1.11.2.5 Get Discovery Group

This service operation is used to fetch discovery group.

Type: GET

URI: {apiRoot}/capif-configuration/v1/discovery-group/nefServicesGroup/{groupId}

Initiated By: CNCC

Example:

The following example shows how discovery group are fetched by a GET request.

{apiRoot}/capif-configuration/v1/discovery-group/nefServicesGroup/group4

Example of the Response Body

```
{
  "groupId": "group4",
  "nefServices":
  [
    "Site1RegSec::3gpp-as-session-with-qos",
    "Site1RegSec::3gpp-monitoring-event",
    "Site1RegSec::3gpp-traffic-influence",
    "Site2RegSec::3gpp-monitoring-event"
  ]
}
```

2.1.11.2.6 Get All Discovery Group

This service operation is used to fetch all discovery group.

Type: GET

URI: {apiRoot}/capif-configuration/v1/discovery-group/nefServicesGroup

Initiated By: CNCC

Example:

The following example shows how all discovery group are fetched by a GET request.

{apiRoot}/capif-configuration/v1/discovery-group/nefServicesGroup

Example of the Response Body

```
{
  "groupId": "group4",
  "nefServices":
  [
    "Site1RegSec::3gpp-as-session-with-qos",
    "Site1RegSec::3gpp-monitoring-event",
    "Site1RegSec::3gpp-traffic-influence",
    "Site2RegSec::3gpp-monitoring-event"
  ]
},
{
  "groupId": "group1",
  "nefServices":
  [
    "Site1RegSec::3gpp-as-session-with-qos",
    "Site1RegSec::3gpp-monitoring-event"
  ]
},
{
  "groupId": "group2",
  "nefServices":
  [
    "Site1RegSec::3gpp-monitoring-event",
    "Site1RegSec::3gpp-traffic-influence",
    "Site2RegSec::3gpp-monitoring-event"
  ]
}
```

2.1.11.2.7 Delete Discovery Group

This service operation is used to delete all discovery group.

Type: DELETE

URI: {apiRoot}/capif-configuration/v1/discovery-group/nefServicesGroup/{groupId}

Initiated By: CNCC

Example:

The following example shows how all discovery group are deleted by a DELETE request.

{apiRoot}/capif-configuration/v1/discovery-group/nefServicesGroup/{group4}

2.1.11.2.8 Pre-provision the Invoker

This service operation is used to pre-provision invoker.

Type: POST

URI: {apiRoot}/capif-configuration/v1/pre-provisioning-console/apiInvokers/{api-invoker-name}

Initiated By: CNCC

Table 2-184 Response Body Parameters

Field Name	Data Type	Description
apiInvokerName	String	The name of the API invoker.
valid_from	DateTime	The date and time from when the API invoker provisioning starts.
valid_to	DateTime	The date and time by when the API Invoker provisioning remains valid.
group_id	String	The unique identity of the AEF group assigned to the API Invoker.
accessToken	String	The security access token for authentication purpose. This token shall be provided in onboarding request body (onboardingSecret parameter).
onboardingUri	String	The onboarding URI that must be accessed by API Invoker to onboard NEF.

Example:

The following example shows how invoker is pre-provisioned by a POST request.

```
{apiRoot}/capif-configuration/v1/pre-provisioning-console/apiInvokers/Invoker1
```

Table 2-185 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Request Body

```
{  
  "apiInvokerName": "Invoker1",  
  "valid_from": "2021-12-12T17:28:32Z",  
  "valid_to": "2023-11-11T17:28:32Z",  
}
```

```
    "group_id": "Group1"
  }
```

Example of the Request Body

```
{
  "accessToken":
    "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpbnZva2VyTmFtZSI6Ikludm9rZXIiOiwiZ3Jv
    dXBjZCI6Ikdya3VwMSIsImVudm9rZXJJZCI6IjllYzEwN2E2LTgxNjktNDBhZi1iM2I1LTljNjI0NT
    I1ZjBkMSIsInN1YiI6IkJwYmVudm9rZXJQcmVQcm92aXNpb25pbmciLCJpYXQiOiE3MTUyNzk0MDAs
    ImV4cCI6MTcxNjQwMjYwMH0.PY-CNG59pNhFfdja-_F-z-
    GzwYyG14xLQ9iJtXLIaRDoUD3kHjcfcl77s3NwVDgDTmTKWpP01I-YaAwn5-
    wWdfjZ0_U6AyBKHW1J_vnnpLoBU7-
    LkL41WDh1axJSMAkdQQqWBprra2NrG0Gmzd05RSzuXePtTlggir70ku_a9VmKfvYCV06mr3HfMDE2m
    4w2bh9n65yDC863afjANKcE7Hmlgve9p_iRRFR9U9UKqttzNVIHaMSuHPIq5URoZemcCXLFLSiPwx1
    rJq_MxjwnCVYR0vu0AxVteatYa8D5Gw5EhLawszGgXwtvLCDqkAyQyh7npNQck8Umn-9LhuYSVQ",
  "onboardingUri": "http://occapif-ext-ingress-gateway:80/apiRoot/api-
  invoker-management/v1/onboardedInvokers",
  "apiInvokerName": "Invoker1",
  "valid_from": "2024-05-09T18:30:00Z",
  "valid_to": "2024-05-22T18:30:00Z",
  "group_id": "Group1"
}
```

2.1.11.2.9 Get Invoker Details

This service operation is used to fetch invoker details.

Type: GET

URI: {apiRoot}/capif-configuration/v1/pre-provisioning-console/apiInvokers/{api-invoker-name}

Initiated By: CNCC

Example:

The following example shows how invoker details are fetched by a GET request.

```
{apiRoot}/capif-configuration/v1/pre-provisioning-console/apiInvokers/
{Invoker1}
```

Example of the Response Body

```
{
  "accessToken":
    "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpbnZva2VyTmFtZSI6Ikludm9rZXIiOiwiZ3Jv
    dXBjZCI6Ikdya3VwMSIsImVudm9rZXJJZCI6IjllYzEwN2E2LTgxNjktNDBhZi1iM2I1LTljNjI0NT
    I1ZjBkMSIsInN1YiI6IkJwYmVudm9rZXJQcmVQcm92aXNpb25pbmciLCJpYXQiOiE3MTUyNzk0MDAs
    ImV4cCI6MTcxNjQwMjYwMH0.PY-CNG59pNhFfdja-_F-z-
    GzwYyG14xLQ9iJtXLIaRDoUD3kHjcfcl77s3NwVDgDTmTKWpP01I-YaAwn5-
    wWdfjZ0_U6AyBKHW1J_vnnpLoBU7-
    LkL41WDh1axJSMAkdQQqWBprra2NrG0Gmzd05RSzuXePtTlggir70ku_a9VmKfvYCV06mr3HfMDE2m
    4w2bh9n65yDC863afjANKcE7Hmlgve9p_iRRFR9U9UKqttzNVIHaMSuHPIq5URoZemcCXLFLSiPwx1
    rJq_MxjwnCVYR0vu0AxVteatYa8D5Gw5EhLawszGgXwtvLCDqkAyQyh7npNQck8Umn-9LhuYSVQ",
  "onboardingUri": "http://occapif-ext-ingress-gateway:80/apiRoot/api-
```

```
invoker-management/v1/onboardedInvokers",
  "apiInvokerName": "Invoker1",
  "valid_from": "2024-05-09T18:30:00Z",
  "valid_to": "2024-05-22T18:30:00Z",
  "group_id": "Group1"
}
```

2.1.11.2.10 Get All Invoker Details

This service operation is used to fetch all invoker details.

Type: GET

URI: {apiRoot}/capif-configuration/v1/pre-provisioning-console/apiInvokers

Initiated By: CNCC

Example:

The following example shows how all invoker details are fetched by a GET request.

```
{apiRoot}/capif-configuration/v1/pre-provisioning-console/apiInvokers
```

Example of the Response Body

```
{
  "accessToken":
    "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpbnZva2VyTmFtZSI6Ikludm9rZXIiOiwiZ3Jv
    dXBjZCI6IkdYb3VwMSIsImVudm9rZXJJZCI6IjllYzEwN2E2LTgxNjktNDBhZi1iM2I1LTljNjI0NT
    I1ZjBkMSIsInN1YiI6IkJwYUJludm9rZXJQcmVQcm92aXNpb25pbmciLCJpYXQiOiE3MTUyNzk0MDAs
    ImV4cCI6MTcxNjQwMjYwMH0.PY-CNG59pNhFfdja-_F-z-
    GzwYyG14xLQ9iJtXlIaRDoUD3kHjcfcl77s3NwVDgDTmTKWpP01I-YaAwn5-
    wwdfjZ0_U6AyBKHW1J_vnnpLoBU7-
    LkL41WDh1axJSMAkdQQqwbPrRa2NrG0Gmzd05RSzuXePtTlggir70ku_a9VmKfVYCV06mr3HfMDE2m
    4w2bh9n65yDC863afjANKce7Hmlgve9p_iRRFR9U9UKqttzNVIHaMSuHPIq5URoZemcCXFLSiPw1
    rJq_MxjwnCVYR0vu0AxVteatYa8D5Gw5EhLawszGgXwtvLCDqkAyQyh7npNQck8Umn-9LhuYSVQ",
  "onboardingUri": "http://occapif-ext-ingress-gateway:80/apiRoot/api-
  invoker-management/v1/onboardedInvokers",
  "apiInvokerName": "Invoker1",
  "valid_from": "2024-05-09T18:30:00Z",
  "valid_to": "2024-05-22T18:30:00Z",
  "group_id": "Group1"
},
{
  "accessToken":
    "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpbnZva2VyTmFtZSI6Ikludm9rZXIiOiwiZ3Jv
    dXBjZCI6IkdYb3VwMSIsImVudm9rZXJJZCI6IjllYzEwN2E2LTgxNjktNDBhZi1iM2I1LTljNjI0NT
    I1ZjBkMSIsInN1YiI6IkJwYUJludm9rZXJQcmVQcm92aXNpb25pbmciLCJpYXQiOiE3MTUyNzk0MDAs
    ImV4cCI6MTcxNjQwMjYwMH0.PY-CNG59pNhFfdja-_F-z-
    GzwYyG14xLQ9iJtXlIaRDoUD3kHjcfcl77s3NwVDgDTmTKWpP01I-YaAwn5-
    wwdfjZ0_U6AyBKHW1J_vnnpLoBU7-
    LkL41WDh1axJSMAkdQQqwbPrRa2NrG0Gmzd05RSzuXePtTlggir70ku_a9VmKfVYCV06mr3HfMDE2m
    4w2bh9n65yDC863afjANKce7Hmlgve9p_iRRFR9U9UKqttzNVIHaMSuHPIq5URoZemcCXFLSiPw1
    rJq_MxjwnCVYR0vu0AxVteatYa8D5Gw5EhLawszGgXwtvLCDqkAyQyh7npNQck8Umn-9LhuYSVQ",
  "onboardingUri": "http://occapif-ext-ingress-gateway:80/apiRoot/api-
  invoker-management/v1/onboardedInvokers",
```

```

    "apiInvokerName": "Invoker2",
    "valid_from": "2024-05-09T18:30:00Z",
    "valid_to": "2024-05-22T18:30:00Z",
    "group_id": "Group2"
  },
  {
    "accessToken":
    "eyJ0eXAiOiJKV1QiLCJhbGciOiJSUzI1NiJ9.eyJpbnZva2VyTmFtZSI6Ikludm9rZXIxIiwia3JvdXBjZCI6Ikd5b3VwMSIsImVudm9rZXJJZCI6IjllYzEwN2E2LTgxNjktNDBhZi1iM2I1LTljNjI0NTI1ZjBkMSIsInN1YiI6IkFwaUludm9rZXJQcmVQcm92aXNpb25pbmciLCJpYXQiOiE3MTUyNzk0MDAsImV4cCI6MTcxNjQwMjYwMH0.PY-CNG59pNhFfdja-_F-z-GzwYyG14xLQ9iJtXlIaRDoUD3kHjcfcl77s3NwVdgDTmTKWpP0I1-YaAwN5-wWdfjZ0_U6AyBKHW1J_vnnpLoBU7-LkL41WDh1axJSmAKdQQqWBprra2NrG0Gmzd05RSzuXePtTlggir70ku_a9VmKfvYCV06mr3HfMDE2m4w2bh9n65yDC863afjANKcE7Hmlgve9p_iRRFR9U9UKqttzNVIHaMSuHPIq5URoZemcCXFLSiPxl1rJq_MxjwnCVYR0vu0AxVteatYa8D5Gw5EhLawszGgXwtvLCDqkAyQyh7npNQck8Umn-9LhuYSVQ",
    "onboardingUri": "http://occapif-ext-ingress-gateway:80/apiRoot/api-invoker-management/v1/onboardedInvokers",
    "apiInvokerName": "Invoker3",
    "valid_from": "2024-05-09T18:30:00Z",
    "valid_to": "2024-05-22T18:30:00Z",
    "group_id": "Group3"
  }
}

```

2.1.11.2.11 Delete Invoker

This service operation is used to delete invoker.

Type: DELETE

URI: {apiRoot}/capif-configuration/v1/pre-provisioning-console/apiInvokers/{invokerName}

Initiated By: CNCC

Example:

The following example shows how invoker is deleted by a DELETE request.

```
{apiRoot}/capif-configuration/v1/pre-provisioning-console/apiInvokers/Invoker1
```

2.1.11.2.12 Get Access Token

This service operation is used to fetch access token.

Type: GET

URI: {apiroot}/capif/nf-common-component/v1/afmgr/accessToken

Initiated By: CNCC

Table 2-186 Response Body Parameters

Field Name	Data Type	Description
expiryTime	string	To set the validity of the token to access NEF services. Note: - Maximum value can be 86400 sec. Even if a value greater than that is set, it is internally still considered as 86400 sec only. - While obtaining the access token for a API Invoker, if time (in sec) left in its expiry becomes lesser than configured expiryTime, then the system generates the token whose expiry is set to the time (in sec) that the API Invoker has left.

Example:

The following example shows how access token are fetched by a GET request.

```
{apiroot}/capif/nf-common-component/v1/afmgr/accessToken
```

Table 2-187 Supported Response Codes

Code	Description
200	OK
400	Bad Request
403	Forbidden
404	Not Found
406	Not Acceptable
429	Too Many Requests
500	Internal Server Error
503	Service Unavailable

Example of the Response Body

```
{
  "expiryTime":67800
}
```

2.1.11.2.13 Update Access Token Expiry Time

This service operation is used to update access token expiry time.

Type: PUT

URI: {apiroot}/capif/nf-common-component/v1/afmgr/accessToken

Initiated By: CNCC

Example:

The following example shows how access token expiry time are updated by a PUT request.

```
{apiroot}/capif/nf-common-component/v1/afmgr/accessToken
```

Example of the Response Body

```
{  
  "expiryTime":34800  
}
```

Example of the Response Body

```
{  
  "expiryTime":34800  
}
```