**3GPP TSG-CT WG4 Meeting #107-bis-eC4-220xyz**

**E-Meeting, 17th – 21st January 2022 (was C4-220216)**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.1* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  | **29.510** | **CR** | **0654** | **rev** | **1** | **Current version:** | **17.4.0** |  |
|  | | | | | | | | |
| *For* [***HE******LP***](http://www.3gpp.org/3G_Specs/CRs.htm#_blank)*on using this form: comprehensive instructions can be found at* [*http://www.3gpp.org/Change-Requests*](http://www.3gpp.org/Change-Requests)*.* | | | | | | | | |
|  | | | | | | | | |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Proposed change affects:*** | UICC apps |  | ME |  | Radio Access Network |  | Core Network | **X** |

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | | | | | | | | | | |
| ***Title:*** | Resource-based authorization on the NRF APIs | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | Ericsson | | | | | | | | | |
| ***Source to TSG:*** | CT4 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | SBIProtoc17 | | | | |  | ***Date:*** | | | 2022-01-10 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | **B** |  | | | | | ***Release:*** | | | Rel-17 |
|  | *Use one of the following categories:* ***F*** *(correction)* ***A*** *(mirror corresponding to a change in an earlier release)* ***B*** *(addition of feature),* ***C*** *(functional modification of feature)* ***D*** *(editorial modification)*  Detailed explanations of the above categories can be found in 3GPP [TR 21.900](http://www.3gpp.org/ftp/Specs/html-info/21900.htm). | | | | | | | | *Use one of the following releases: Rel-8 (Release 8) Rel-9 (Release 9) Rel-10 (Release 10) Rel-11 (Release 11) … Rel-15 (Release 15) Rel-16 (Release 16) Rel-17 (Release 17) Rel-18 (Release 18)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | When Oauth2 -based authorization is used on an API, it is possible to indicate in the access token (via the "scope" claim) the specific access to resources and operations that are allowed for a given consumer.  This mechanism has been introduced in 3GPP APIs since Rel-16 onwards, and has been added progressively to each API, as soon as relevant use cases are identified for it.  For the NRF APIs, it is possible to have, among others, the following scenarios:  - A given NRF consumer (e.g. an NF Service Producer) may be authorized to register its NF Profile in NRF (PUT request on the "/nf-instances/{nfInstanceId}" resource of the NFManagement API), but it is not authorized to retrieve the full catalog of registered NF Instances in the NRF (GET request on the "/nf-instances" resource of the NFManagement API).  - A given NRF consumer (e.g. a pure NF Service Consumer) may be authorized to issue discovery requests, and to subscribe to changes on the discovered NF profiles (POST request on the "/subscriptions" resource and PATCH/DELETE requests on the "/subscriptions/{subscriptionID}" resource on the NFManagement API), but it is not authorized to register its NFProfile in the NRF.  So, it is quite frequent to have authorization scenarios, where more granularity is desirable, than the existing usage of a "single scope per service", which basically means "allowed/not-allowed" for all the resources and operations of the entire API. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Add scopes per resource/operation on the NFManagement and NFDiscovery APIs. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | Authorization of the NRF APIs based on Oauth2 have a sub-optimal granularity and does not leverage well-established features (resource-based authorization) currently used in many other 3GPP APIs. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 6.1.8, 6.2.8, A.2, A.3 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **Y** | **N** |  | | | |  | | |
| ***Other specs*** | |  | **X** | Other core specifications | | | | TS/TR ... CR ... | | |
| ***affected:*** | |  | **X** | Test specifications | | | | TS/TR ... CR ... | | |
| ***(show related CRs)*** | |  | **X** | O&M Specifications | | | | TS/TR ... CR ... | | |
|  | |  | | | | | | | | |
| ***Other comments:*** | | This CR introduces backwards-compatible new features with impact on OpenAPI specification files:  - TS29510\_Nnrf\_NFManagement.yaml  - TS29510\_Nnrf\_NFDiscovery.yaml | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

\* \* \* First Change \* \* \* \*

### 6.1.8 Security

As indicated in 3GPP TS 33.501 [15], the access to the Nnrf\_NFManagement API may be authorized by means of the OAuth2 protocol (see IETF RFC 6749 [16]), using the "Client Credentials" authorization grant, where the NRF plays the role of the authorization server.

If Oauth2 authorization is used on the Nnrf\_NFManagement API, an NF Service Consumer, prior to consuming services offered by the Nnrf\_NFManagement API, shall obtain a "token" from the authorization server, by invoking the Access Token Request service, as described in clause 5.4.2.2.

NOTE: When multiple NRFs are deployed in a network, the NRF used as authorization server is the same NRF where the Nnrf\_NFManagement service is invoked by the NF Service Producer.

The Nnrf\_NFManagement API defines the following scopes for OAuth2 authorization:

Table 6.1.8-1: Oauth2 scopes defined in Nnrf\_NFManagement API

|  |  |
| --- | --- |
| Scope | Description |
| "nnrf-nfm" | Access to the Nnrf\_NFManagement API |
| "nnrf-nfm:nf-instances:read" | Acess to read the nf-instances resource |
| "nnrf-nfm:nf-instance-id:write" | Access to create/modify/delete an individual nf-instance-id resource |
| "nnrf-nfm:subscriptions:write" | Access to create/modify/delete subscription resources |

\* \* \* Next Change \* \* \* \*

### 6.2.8 Security

As indicated in 3GPP TS 33.501 [15], the access to the Nnrf\_NFDiscovery API may be authorized by means of the OAuth2 protocol (see IETF RFC 6749 [16]), using the "Client Credentials" authorization grant, where the NRF plays the role of the authorization server.

If Oauth2 authorization is used, an NF Service Consumer, prior to consuming services offered by the Nnrf\_NFDiscovery API, shall obtain a "token" from the authorization server, by invoking the Access Token Request service, as described in clause 5.4.2.2.

NOTE: When multiple NRFs are deployed in a network, the NRF used as authorization server is the same NRF where the Nnrf\_NFDiscovery service is invoked by the NF Service Consumer.

The Nnrf\_NFDiscovery API defines the following scopes for OAuth2 authorization:

Table 6.2.8-1: Oauth2 scopes defined in Nnrf\_NFDiscovery API

|  |  |
| --- | --- |
| Scope | Description |
| "nnrf-disc" | Access to the Nnrf\_NFDiscovery API |
| "nnrf-disc:scp-domain:read" | Access to read the scp-domain-routing-info resource |
| "nnrf-disc:scp-domain-subs:write" | Access to create/delete a scp-domain subscription resource |

\* \* \* Next Change \* \* \* \*

## A.2 Nnrf\_NFManagement API

*(... text not shown for clarity ...)*

paths:

/nf-instances:

get:

summary: Retrieves a collection of NF Instances

operationId: GetNFInstances

tags:

- NF Instances (Store)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-nfm

- oAuth2ClientCredentials:

- nnrf-nfm

- nnrf-nfm:nf-instances:read

parameters:

- name: nf-type

in: query

description: Type of NF

required: false

schema:

$ref: '#/components/schemas/NFType'

*(... text not shown for clarity ...)*

options:

summary: Discover communication options supported by NRF for NF Instances

operationId: OptionsNFInstances

tags:

- NF Instances (Store)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-nfm

responses:

'200':

description: OK

content:

application/json:

schema:

$ref: '#/components/schemas/OptionsResponse'

*(... text not shown for clarity ...)*

/nf-instances/{nfInstanceID}:

get:

summary: Read the profile of a given NF Instance

operationId: GetNFInstance

tags:

- NF Instance ID (Document)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-nfm

parameters:

- name: nfInstanceID

in: path

description: Unique ID of the NF Instance

required: true

schema:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/NfInstanceId'

*(... text not shown for clarity ...)*

put:

summary: Register a new NF Instance

operationId: RegisterNFInstance

tags:

- NF Instance ID (Document)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-nfm

- oAuth2ClientCredentials:

- nnrf-nfm

- nnrf-nfm:nf-instance-id:write

parameters:

- name: nfInstanceID

in: path

required: true

description: Unique ID of the NF Instance to register

schema:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/NfInstanceId'

*(... text not shown for clarity ...)*

patch:

summary: Update NF Instance profile

operationId: UpdateNFInstance

tags:

- NF Instance ID (Document)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-nfm

- oAuth2ClientCredentials:

- nnrf-nfm

- nnrf-nfm:nf-instance-id:write

parameters:

- name: nfInstanceID

in: path

required: true

description: Unique ID of the NF Instance to update

schema:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/NfInstanceId'

*(... text not shown for clarity ...)*

delete:

summary: Deregisters a given NF Instance

operationId: DeregisterNFInstance

tags:

- NF Instance ID (Document)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-nfm

- oAuth2ClientCredentials:

- nnrf-nfm

- nnrf-nfm:nf-instance-id:write

parameters:

- name: nfInstanceID

in: path

required: true

description: Unique ID of the NF Instance to deregister

schema:

$ref: 'TS29571\_CommonData.yaml#/components/schemas/NfInstanceId'

*(... text not shown for clarity ...)*

/subscriptions:

post:

summary: Create a new subscription

operationId: CreateSubscription

tags:

- Subscriptions (Collection)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-nfm

- oAuth2ClientCredentials:

- nnrf-nfm

- nnrf-nfm:subscriptions:write

parameters:

- name: Content-Encoding

in: header

description: Content-Encoding, described in IETF RFC 7231

schema:

type: string

*(... text not shown for clarity ...)*

/subscriptions/{subscriptionID}:

patch:

summary: Updates a subscription

operationId: UpdateSubscription

tags:

- Subscription ID (Document)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-nfm

- oAuth2ClientCredentials:

- nnrf-nfm

- nnrf-nfm:subscription-id:write

parameters:

- name: subscriptionID

in: path

required: true

description: Unique ID of the subscription to update

schema:

type: string

pattern: '^([0-9]{5,6}-)?[^-]+$'

*(... text not shown for clarity ...)*

delete:

summary: Deletes a subscription

operationId: RemoveSubscription

tags:

- Subscription ID (Document)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-nfm

- oAuth2ClientCredentials:

- nnrf-nfm

- nnrf-nfm:subscription-id:write

parameters:

- name: subscriptionID

in: path

required: true

description: Unique ID of the subscription to remove

schema:

type: string

pattern: '^([0-9]{5,6}-)?[^-]+$'

*(... text not shown for clarity ...)*

components:

securitySchemes:

oAuth2ClientCredentials:

type: oauth2

flows:

clientCredentials:

tokenUrl: '/oauth2/token'

scopes:

nnrf-nfm: Access to the Nnrf\_NFManagement API

nnrf-nfm:nf-instances:read: Acess to read the nf-instances resource

nnrf-nfm:nf-instance-id:write: Access to create/modify/delete an individual nf-instance-id resource

nnrf-nfm:subscriptions:write: Access to create/modify/delete subscription resources

schemas:

*(... text not shown for clarity ...)*

\* \* \* Next Change \* \* \* \*

## A.3 Nnrf\_NFDiscovery API

*(... text not shown for clarity ...)*

paths:

/nf-instances:

get:

summary: Search a collection of NF Instances

operationId: SearchNFInstances

tags:

- NF Instances (Store)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-disc

parameters:

- name: Accept-Encoding

in: header

description: Accept-Encoding, described in IETF RFC 7231

schema:

type: string

*(... text not shown for clarity ...)*

/searches/{searchId}:

get:

operationId: RetrieveStoredSearch

tags:

- Stored Search (Document)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-disc

parameters:

- $ref: '#/components/parameters/searchId'

- name: Accept-Encoding

in: header

description: Accept-Encoding, described in IETF RFC 7231

schema:

type: string

*(... text not shown for clarity ...)*

/searches/{searchId}/complete:

get:

operationId: RetrieveCompleteSearch

tags:

- Complete Stored Search (Document)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-disc

parameters:

- $ref: '#/components/parameters/searchId'

- name: Accept-Encoding

in: header

description: Accept-Encoding, described in IETF RFC 7231

schema:

type: string

*(... text not shown for clarity ...)*

/scp-domain-routing-info:

get:

operationId: SCPDomainRoutingInfoGet

tags:

- SCP Domain Routing Information (Document)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-disc

- oAuth2ClientCredentials:

- nnrf-disc

- nnrf-disc:scp-domain:read

parameters:

- name: local

in: query

description: Indication of local SCP Domain Routing Information

required: false

schema:

type: boolean

default: false

*(... text not shown for clarity ...)*

/scp-domain-routing-info-subs:

post:

summary: Create a new subscription

operationId: ScpDomainRoutingInfoSubscribe

tags:

- SCP Domain Routing Information Subscriptions (Collection)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-disc

- oAuth2ClientCredentials:

- nnrf-disc

- nnrf-disc:scp-domain-subs:write

parameters:

- name: Content-Encoding

in: header

description: Content-Encoding, described in IETF RFC 7231

schema:

type: string

*(... text not shown for clarity ...)*

/scp-domain-routing-info-subs/{subscriptionID}:

delete:

summary: Deletes a subscription

operationId: ScpDomainRoutingInfoUnsubscribe

tags:

- Individual SCP Domain Routing Information Subscription (Document)

security:

- {}

- oAuth2ClientCredentials:

- nnrf-disc

- oAuth2ClientCredentials:

- nnrf-disc

- nnrf-disc:scp-domain-subs-id:write

parameters:

- name: subscriptionID

in: path

required: true

description: Unique ID of the subscription to remove

schema:

type: string

*(... text not shown for clarity ...)*

components:

securitySchemes:

oAuth2ClientCredentials:

type: oauth2

flows:

clientCredentials:

tokenUrl: '/oauth2/token'

scopes:

nnrf-disc: Access to the Nnrf\_NFDiscovery API

nnrf-disc:scp-domain:read: Access to read the scp-domain-routing-info resource

nnrf-disc:scp-domain-subs:write: Access to create/delete a scp-domain subscription resource

*(... text not shown for clarity ...)*

\* \* \* End of Changes \* \* \* \*