**3GPP TSG-SA3 Meeting #105-e *draft\_S3-214194-r1***

**e-meeting, 8 - 19 November 2021**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.1* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  |  | **CR** | **1236** | **rev** | **1** | **Current version:** |  |  |
|  | | | | | | | | |
| *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:*** | NRF deployments | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | Nokia, Nokia Shanghai Bell | | | | | | | | | |
| ***Source to TSG:*** | S3 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** |  | | | | |  | ***Date:*** | | |  |
|  |  | | | |  | |  | | |  |
| ***Category:*** | **F** |  | | | | | ***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:*** | | Multiple NRFs can be deployed in a PLMN, optionally using a hierarchical structure whereby an NRF may redirect or forward service requests to another NRF. One (or more) NRF can serve the entire PLMN, a set of network slices, or a single network slice.  33.875 identified in KI#8 as deployment model assumes that NFc needs to be registered at a local NRF or that NFc is known (as Oauth client) at a local NRF. It also assumes that one NRF is trusting the other NRF in the same PLMN.  When requesting an access token, NFc goes first to its local NRF, which authenticates NFc and then forwards or redirects the request to the target NRF, where a NFp has registered its services. In this case the local NRF authenticates the NFc and the target NRF (holding the policy for NFp services) provides the access token for NFp service.  Clarification text is required to address the various NRF deployments and how the NFc behaves when it has discovered a local NRF or if the NF Service Consumer requests an NRF, where the NF Service Producer is not registered. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Adding deployment options for NRFs as a new clause.  Clarfiy that the NF Service Consumer may have discovered a specific NRF in advance, e.g. a slice specific NRF, and can send its request directly to that NRF. If the NF Service Consumer requests an NRF, where the NF Service Producer is not registered (see NRF deployment options in 13.4.1.1.1a), the requested NRF needs to redirect/forward the service request to that NRF.  Further clarify that in a local NRF deployment, the NF Service Producer only gets the certificate of the local NRF. Thus, the local NRF would need to check if the NF Service Consumer is authorized and the NRF where the NF Service Producer is registered would need to trust the NRF which has verified the NF Service Consumer. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | Some NRF deployment scenarios are missed in 33.501, no alignment with CT4 29.510 | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 13.4.1.1.1a (new), 13.4.1.1.2 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **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's revision history:*** | | S3-214194 | | | | | | | | |

\*\*\*\*\*\*\*\*\*\*\*\*START OF CHANGES

##### 13.4.1.1.1a NRF deployments

An operator network implementation might deploy multiple NRFs. Without excluding other cases this might be due to network slicing or network segmentation. Each NRF might have different scope, for example:

- an NRF might serve the entire PLMN,

- an NRF might serve a set of network slices,

- an NRF might serve the shared functions of some slices,

- an NRF might be slice-specific, i.e., serve the dedicated functions of a single network slice,

- an NRF might serve a specific region(s),

- NRFs might support a hierarchical structure.

\*\*\*\*\*\*\*\*\*\*\*\*NEXT CHANGE

##### 13.4.1.1.2 Service Request Process

The complete service request is a two-step process including requesting an access token by NF Service Consumer (Step 1, i.e. 1a or 1b), and then verification of the access token by NF Service Producer (Step 2).

**Step 1: Access token request**

Pre-requisite:

- The NF Service consumer (OAuth2.0 client) is registered with the NRF (Authorization Server).

- The NF Service Producer (OAuth2.0 resource server) is registered with the NRF (Authorization Server) with "additional scope" information per NF type.

- The NRF and NF Service Producer share the required credentials.

- The NRF and NF have mutually authenticated each other.

**1a. Access token request** **for** **accessing services of NF Service Producers of a specific NF type**

The following procedure describes how the NF Service Consumer obtains an access token before service access to NF Service Producers of a specific NF type.

A NF Service Consumer that is aware of an NRF serving a NF Service Producer, e.g., a slice-specific NRF, can send its access token request directly to that NRF.

If the NF Service Consumer requests an access token from an NRF, where the NF Service Producer is not registered (see NRF deployment options in 13.4.1.1.1a), the requested NRF needs to redirect/forward the access token service request appropriately so that it reaches the NRF serving the NF Service Producer.



Figure 13.4.1.1.2-1: NF Service Consumer obtaining access token before NF Service access

1. The NF Service Consumer shall request an access token from the NRF in the same PLMN using the Nnrf\_AccessToken\_Get request operation. The message shall include the NF Instance Id(s) of the NF Service Consumer, the requested "scope" including the expected NF Service name(s) and optionally "additional scope" information (i.e. requested resources and requested actions (service operations) on the resources), NF type of the expected NF Service Producer instance and NF Service Consumer. The NF Service Consumer may also include a list of NSSAIs or list of NSI IDs for the expected NF Service Producer instances.

The message may include the NF Set ID of the expected NF Service Producer instances.

The message may include a list of S-NSSAIs of the NF Service Consumer.

2. The NRF may verify that the input parameters (e.g., NF type) in the access token request match with the corresponding ones in the public key certificate of the NF Service Consumer or those in the NF profile of the NF Service Consumer. The NRF checks whether the NF Service Consumer is authorized to access the requested service(s). If the NF Service Consumer is authorized, the NRF shall then generate an access token with appropriate claims included. The NRF shall digitally sign the generated access token based on a shared secret or private key as described in RFC 7515 [45]. If the NF Service Consumer is not authorized, the NRF shall not issue an access token to the NF Service Consumer.

The claims in the token shall include the NF Instance Id of NRF (issuer), NF Instance Id of the NF Service Consumer (subject), NF type of the NF Service Producer (audience), expected service name(s), (scope), expiration time (expiration) and optionally "additional scope" information (allowed resources and allowed actions (service operations) on the resources). The claims may include a list of NSSAIs or NSI IDs for the expected NF Service Producer instances. The claims may include the NF Set ID of the expected NF Service Producer instances.

3. If the authorization is successful, the NRF shall send access token to the NF Service Consumer in the Nnrf\_AccessToken\_Get response operation, otherwise it shall reply based on Oauth 2.0 error response defined in RFC 6749 [43]. The other parameters (e.g., the expiration time, allowed scope) sent by NRF in addition to the access token are described in TS 29.510 [68].

The NF Service Consumer may store the received token(s). Stored tokens may be re-used for accessing service(s) from NF Service Producer NF type listed in claims (scope, audience) during their validity time.

**1b. Access token request for accessing services of a specific NF Service Producer instance / NF Service Producer service instance**

The following steps describes how the NF Service Consumer obtains an access token before service access to a specific NF Service Producer instance / NF Service Producer service instance.

A NF Service Consumer that is aware of an NRF serving a NF Service Producer, e.g., a slice-specific NRF, can send its access token request directly to that NRF.

If the NF Service Consumer requests an access token from an NRF, where the NF Service Producer is not registered (see NRF deployment options in 13.4.1.1.1a), the requested NRF needs to redirect/forward the access token service request appropriately so that it reaches the NRF serving the NF Service Producer.

1. The NF Service Consumer shall request an access token from the NRF for a specific NF Service Producer instance / NF Service Producer service instance. The request shall include the NF Instance Id(s) of the requested NF Service Producer, the expected NF Service name, optionally "additional scope" information (allowed resources and allowed actions (service operations) on the resources) and NF Instance Id of the NF Service Consumer.

2. The NRF checks whether the NF Service Consumer is authorized to use the requested NF Service Producer instance/NF Service Producer service instance, and then proceeds to generate an access token with the appropriate claims included. If the NF Service Consumer is not authorized, the NRF shall not issue an access token to the NF Service Consumer.

The claims in the token shall include the NF Instance Id of NRF (issuer), NF Instance Id of the NF Service Consumer (subject), NF Instance Id or several NF Instance Id(s) of the requested NF Service Producer (audience), expected service name(s) (scope), optionally "additional scope" information (allowed resources and allowed actions (service operations) on the resources), and expiration time (expiration).

3. The token shall be included in the Nnrf\_AccessToken\_Get response sent to the NF Service Consumer. The NF Service Consumer may store the received token(s). Stored tokens may be re-used for accessing service(s) from NF Instance Id or several NF Instance Id(s) of the requested NF Service Producer instance listed in claims (scope, audience) during their validity time.

**Step 2: Service access request based on token verification**

The following figure and procedure describe how authorization is performed during Service request of the NF Service Consumer. Prior to the request, the NF Service Consumer may perform Nnrf\_NFDiscovery\_Request operation with the requested additional scopes to select a suitable NF Service Producer (resource server) which is able to authorize the Service Access request.



Figure 13.4.1.1.2-2: NF Service Consumer requesting service access with an access token

Pre-requisite: The NF Service Consumer is in possession of a valid access token before requesting service access from the NF Service Producer.

1. The NF Service Consumer requests service from the NF Service Producer. The NF Service Consumer shall include the access token.

The NF Service Consumer and NF Service Producer shall authenticate each other following clause 13.3.

2. The NF Service Producer shall verify the token as follows:

- The NF Service Producer ensures the integrity of the token by verifying the signature using NRF’s public key or checking the MAC value using the shared secret. If integrity check is successful, the NF Service Producer shall verify the claims in the token as follows:

NOTE: Void.

- It checks that the audience claim in the access token matches its own identity or the type of NF Service Producer. If a list of NSSAIs or list of NSI IDs is present, the NF Service Producer shall check that it serves the corresponding slice(s).

- If an NF Set ID present, the NF Service Producer shall check the NF Set ID in the claim matches its own NF Set ID.

- If scope is present, it checks that the scope matches the requested service operation.

- If the access token contains "additional scope" information (i.e. allowed resources and allowed actions (service operations) on the resources), it checks that the additional scope matches the requested service operation.

- It checks that the access token has not expired by verifying the expiration time in the access token against the current data/time.

- If the CCA is present in the service request, it may verify the CCA as specified in clause 13.3.8.3 and that the subject claim (i.e., the NF Instance Id of the NF Service Consumer) in the access token matches the subject claim in the CCA.

3. If the verification is successful, the NF Service Producer shall execute the requested service and responds back to the NF Service Consumer. Otherwise it shall reply based on Oauth 2.0 error response defined in RFC 6749 [43].

\*\*\*\*\*\*\*\*\*\*\*\*END OF CHANGES