**3GPP TSG-CT WG4 Meeting #99eC4-204xxx**

**E-Meeting, 18nd – 28th August 2020**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.0* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  | **29.573** | **CR** | **0046** | **rev** | **2** | **Current version:** | **16.3.0** |  |
|  | | | | | | | | |
| *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:*** | Correction of flow description | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | NTT DOCOMO | | | | | | | | | |
| ***Source to TSG:*** | CT4 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | TEI16 | | | | |  | ***Date:*** | | | 2020-08-11 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | **F** |  | | | | | ***Release:*** | | | Rel-16 |
|  | *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-12 (Release 12)* *Rel-13 (Release 13) Rel-14 (Release 14) Rel-15 (Release 15) Rel-16 (Release 16)* | |
|  |  | | | | | | | | | |
| ***Reason for change:*** | | Flows in clause 5 use “../XXXX” to indicate the resource. The use of “..” is better not used to represent the entire resource, as thisimplies as an relative path.  Therefore, it should be corrected to use “…” with three dots, which is what is also described in TS 29.500. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Flows in clause 5 are update so that the referred resources start with “…” with three dots instead of “..” with two. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | Incorrect resource described under clause 5. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 5.2.2, 5.2.3.2, 5.2.3.3, 5.2.3.4, 5.2.4, 5.2.5, 5.3.2.4, 5.4.2, 5.4.3 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **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 does not introduce changes to any OpenAPI specification file. | | | | | | | | |
|  | |  | | | | | | | | |
| ***This CR's revision history:*** | |  | | | | | | | | |

First Change

### 5.2.2 Security Capability Negotiation Procedure

The initiating SEPP shall initiate a Security Capability Negotiation procedure towards the responding SEPP to agree on a security mechanism to use for protecting NF service related signalling over N32-f. An end to end TLS connection shall be setup between the SEPPs before the initiation of this procedure. The procedure is described in Figure 5.2.2-1 below.



Figure 5.2.2-1: Security Capability Negotiation Procedure

1. The initiating SEPP issues a HTTP POST request towards the responding SEPP with the request body containing the "SecNegotiateReqData" IE carrying the following information

- Supported security capabilities (i.e PRINS and/or TLS)

2a. On successful processing of the request, the responding SEPP shall respond to the initiating SEPP with a "200 OK" status code and a POST response body that contains the following information

- Selected security capability (i.e PRINS or TLS)

The responding SEPP compares the initiating SEPP's supported security capabilities to its own supported security capabilities and selects, based on its local policy, a security mechanism, which is supported by both the SEPPs. If the selected security capability indicates any other capability other than PRINS, then the HTTP/2 connection initiated between the two SEPPs for the N32 handshake procedures shall be terminated. The negotiated security capability shall be applicable on both the directions. If the selected security capability is PRINS, then the two SEPPs may decide to create (if not available) / maintain HTTP/2 connection(s) where each SEPP acts as a client towards the other (which acts as a server). This may be used for later signalling of N32-f error reporting procedure (see clause 5.2.5) and N32-f context termination procedure (see clause 5.2.4).

2b. On failure, the responding SEPP shall respond to the initiating SEPP with an appropriate 4xx/5xx status code as specified in clause 6.1.4.2.

Next Change

#### 5.2.3.2 Parameter Exchange Procedure for Cipher Suite Negotiation

The parameter exchange procedure for cipher suite negotiation shall be performed after the security capability negotiation procedure if the selected security policy is PRINS.

The procedure is described in Figure 5.2.3.2-1 below.



Figure 5.2.3.2-1: Parameter Exchange Procedure for Cipher Suite Negotiation

1. The initiating SEPP issues a HTTP POST request towards the responding SEPP with the request body containing the "SecParamExchReqData" IE carrying the following information

- Supported cipher suites;

The supported cipher suites shall be an ordered list with the cipher suites mandated by 3GPP TS 33.501 [6] appearing at the top of the list.

The initiating SEPP also provides a N32-f context identifier for the responding SEPP to use towards the initiating SEPP for subsequent JOSE Protected Message Forwarding procedures over N32-f (see clause 5.3.3) when the responding SEPP acts as the forwarding SEPP.

2a. On successful processing of the request, the responding SEPP shall respond to the initiating SEPP with a "200 OK" status code and a POST response body that contains the following information

- Selected cipher suite

The responding SEPP compares the initiating SEPP's supported cipher suites to its own supported cipher suites and selects, based on its local policy, a cipher suite, which is supported by both the SEPPs. The responding SEPP's supported cipher suites shall be an ordered list with the cipher suites mandated by 3GPP TS 33.501 [6] appearing at the top of the list. The selected cipher suite is applicable for both the directions of communication between the SEPPs.

The responding SEPP also provides a N32-f context identifier for the initiating SEPP to use towards the responding SEPP for subsequent JOSE Protected Message Forwarding procedures over N32-f (see clause 5.3.3) when the initiating SEPP acts as the forwarding SEPP.

2b. On failure, the responding P-SEPP shall respond to the initiating SEPP with an appropriate 4xx/5xx status code as specified in clause 6.1.4.3.

Next Change

#### 5.2.3.3 Parameter Exchange Procedure for Protection Policy Exchange

The parameter exchange procedure for protection policy exchange may be performed after the Parameter Exchange Procedure for Cipher Suite Negotiation (see clause 5.2.3.2). If a HTTP/2 connection does not exist towards the peer SEPP at the time of initiating this procedure, the HTTP/2 connection shall be established. If the parameter exchange procedure for the protection policy exchange is not performed then the protection policies between the SEPP shall be exchanged out of bands.

The procedure is described in Figure 5.2.3.3-1 below.



Figure 5.2.3.3-1: Parameter Exchange Procedure for Protection Policy Negotiation

1. The initiating SEPP issues a HTTP POST request towards the responding SEPP with the request body containing the "SecParamExchReqData" IE carrying the following information

- Protection policy information

The protection policy information contains:

- API to IE mapping containing the mapping information of list of leaf IEs for each:

- Request/response and Subscribe / Unsubscribe service operation, identified by the API URI and method; and/or

- Callbacks (e.g Notification service operation), identified by the value of the 3GPP custom HTTP header "3gpp-Sbi-Callback" (see clause 5.2.3 of 3GPP TS 29.500 [4]).

- List of IE types that are to be protected across N32-f (i.e the data type encryption policy as specified in clause 13.2.3.2 of 3GPP TS 33.501 [6]); and

- Against each leaf IE in the API to IE mapping information, a boolean flag indicating whether that IE is allowed to be modified by an IPX on the side of the SEPP sending the protection policy information.

2a. On successful processing of the request, the responding SEPP shall respond to the initiating SEPP with a "200 OK" status code and a POST response body that contains the following information

- Selected protection policy information

The SEPPs shall store the selected protection policy information and shall apply this policy for subsequent message transfers over N32-f. The selected protection policy is applicable for both the directions of communication between the SEPPs.

The HTTP/2 connection used for the N32 handshake procedures may be terminated after the completion of this procedure.

2b. On failure, the responding SEPP shall respond to the initiating SEPP with an appropriate 4xx/5xx status code as specified in clause 6.1.4.3.

An illustration of how the protection policy is stored and looked up in the SEPP is provided in figure 5.2.3.3-2



Figure 5.2.3.3-2: Protection Policy Storage and Lookup in SEPP

During the N32-f message forwarding, the SEPP looks at a HTTP request or response it receives from an NF service consumer or NF service producer and then uses the above tables to decide which IEs and headers in the message it shall cipher and integrity protect and which IEs it shall allow the IPXes to modify.

Next Change

#### 5.2.3.4 Parameter Exchange Procedure for Security Information list Exchange

The initiating SEPP shall initiate a Security Information list exchange procedure towards the responding SEPP to exchange the Security Information lists that contain information on IPX public keys or certificates that are needed to verify IPX modifications at the receiving SEPP as specified in clause 13.2.2.2 of 3GPP TS 33.501 [6].

The procedure is described in Figure 5.2.3.4-1 below.



Figure 5.2.3.4-1: Parameter Exchange Procedure for Security Information List exchange

1. The initiating SEPP issues a HTTP POST request towards the responding SEPP with the request body containing the "SecParamExchReqData" IE carrying the following information:

- IPX provider identifier connected to the initiating SEPP;

- List of raw public keys or certificates for that IPX.

2a. On successful processing of the request, the responding SEPP shall respond to the initiating SEPP with a "200 OK" status code and a POST response body that contains the "SecParamExchRspData" IE carrying the following information:

- IPX provider identifier connected to the responding SEPP;

- List of raw public keys or certificates for that IPX.

2b. On failure, the responding SEPP shall respond to the initiating SEPP with an appropriate 4xx/5xx status code as specified in clause 6.1.4.3.

Next Change

### 5.2.4 N32-f Context Termination Procedure

After the completion of the security capability negotiation procedure and/or the parameter exchange procedures, an N32-f context is established between the two SEPPs. The "n32fContextId" of each SEPP is provided to the other SEPP. This context identifier shall be stored in each SEPP until the context is explicitly terminated by the N32-f context termination procedure. The SEPP that is initiating the N32-f context termination procedure shall use the HTTP method POST on the URI: {apiRoot}/n32c-handshake/v1/n32f-terminate. If a HTTP/2 connection does not exist towards the receiving SEPP, a HTTP/2 connection shall be created before initiating this procedure. The procedure is shown below in Figure 5.2.4-1.



Figure 5.2.4-1: N32f Context Termination Procedure

1. The initiating SEPP issues a HTTP POST request towards the responding SEPP with the request body containing the N32-f context id information that is to be terminated.

2a. On success, the responding SEPP, shall:

- stop sending any further messages over the N32-f towards the initiating SEPP;

- once all the ongoing N32-f message exchanges with the initiating SEPP are completed or timed out, delete the N32-f context identified by the "n32fContextId" provided in the request.

The N32-f HTTP/2 connections from the responding SEPP shall not be deleted if they terminate at an IPX, since that HTTP/2 connection may carry traffic towards other PLMN SEPPs as well. The responding SEPP shall return the status code "200 OK" together with an N32ContextInfo payload body that carries the "n32fContextId" of the initiating SEPP that the responding SEPP has stored.

The initiating SEPP shall:

- stop sending any further messages over the N32-f towards the responding SEPP;

- once all the ongoing N32-f message exchanges with the responding SEPP are completed or timed out, delete the local N32-f context identified by this "n32fContextId".

2b. On failure, the responding SEPP shall return an appropriate 4xx/5xx status code together with the "ProblemDetails" JSON body.

Next Change

### 5.2.5 N32-f Error Reporting Procedure

When a SEPP is not able to process a message it received over the N32-f interface due to errors, the error information is conveyed to the sending SEPP by using the N32-f error reporting procedure over the N32-c interface. The SEPP that is initiating the N32-f error reporting procedure shall use the HTTP method POST on the URI: {apiRoot}/n32c-handshake/v1/n32f-error. If a HTTP/2 connection does not exist towards the receiving SEPP, a HTTP/2 connection shall be created before initiating this procedure. The procedure is shown below in Figure 5.2.5-1.



Figure 5.2.5-1: N32f Error Reporting Procedure

1. The initiating SEPP issues a HTTP POST request towards the responding SEPP with the request body containing the N32-f error information that is to be reported.

2a. On success, the responding SEPP, shall:

- log that the N32-f request / response message identified by the "messageId" is not processed by the receiving SEPP;

The responding SEPP shall return the status code "204 No Content".

2b. On failure, the responding SEPP shall return an appropriate 4xx/5xx status code together with the "ProblemDetails" JSON body.

Next Change

#### 5.3.2.4 Message Forwarding to Peer SEPP

Once a SEPP reformats the HTTP/2 message into the "N32ReformattedReqMsg"/"N32ReformattedRspMsg" JSON object as specified in clause 5.3.2, the SEPP forwards the message to the receiving SEPP by invoking a HTTP POST method as shown in figure 5.3.2.4-1 below.



Figure 5.3.2.4-1 Message Forwarding between SEPP on N32-f

1. The initiating SEPP issues a HTTP POST request towards the responding SEPP with the request body containing the "N32ReformattedReqMsg" IE carrying the reformatted HTTP/2 message. The request message shall contain the "n32fContextId" information provided by the responding SEPP to the initiating SEPP earlier during the parameter exchange procedure (see clause 5.2.3). The responding SEPP shall use the "n32fContextId" information to:

- Locate the agreed cipher suite and protection policy;

- Locate the n32ContextId to be used in the response.

2a. On successful processing of the request, the responding SEPP shall:

- reconstruct the HTTP/2 message towards the NF service producer;

- forward the reconstructed HTTP/2 message to the NF service producer;

- wait for the response from the NF service producer; and then

- once the response from the NF service producer is received, respond to the initiating SEPP with a "200 OK" status code and a POST response body that contains the "N32ReformattedRspMsg". The "N32ReformattedRspMsg" shall contain the reformatted HTTP response message from the responding PLMN. The response message shall contain the "n32fContextId" information provided by the initiating SEPP to the responding SEPP earlier during the parameter exchange procedure (see clause 5.2.3).

The responding SEPP shall be able to map the response received from the NF service producer to the HTTP/2 stream ID for the corresponding response it needs to generate towards the initiating SEPP. The HTTP/2 stream ID and the HTTP/2 connection information on either side shall be used to derive this mapping.

2b. On failure, the responding SEPP shall respond to the initiating SEPP with an appropriate 4xx/5xx status code as specified in clause 6.2.4.2.

Next Change

### 5.4.2 Foreign FQDN to Telescopic FQDN Mapping Procedure

This procedure is initiated by an NF Service Consumer (typically an NRF or an NSSF) that needs to interact with a NF in a foreign PLMN (typically the corresponding NRF or NSSF), and to do so, it needs to build a telescopic FQDN of said NF (i.e. concatenation of the FQDN of the foreign FQDN, and the FQDN of the local SEPP), and then the resulting telescopic FQDN needs to be "flattened" (i.e. the FQDN of the NF in the foreign PLMN needs to be converted to a singel label). The procedure is described in Figure 5.4.2-1 below.



Figure 5.4.2-1: Foreign FQDN to Telescopic FQDN Mapping Procedure

1. The NF Service Consumer issues an HTTP GET request towards the local SEPP with a query parameter "foreign-fqdn" containing the FQDN of the NF in the foreign PLMN, that needs to be transformed into a flattened telescopic FQDN.

2a. On successful processing of the request, the responding SEPP shall respond to the NF Service Consumer with a "200 OK" status code and a response body that contains a JSON object of type "TelescopicMapping", containing as attributes the label to be used as first label in the telescopic FQDN, and the domain of the local SEPP to be appended after such first label. The resulting FQDN shall be used by the NF Consumer to setup a TLS session terminated in the local SEPP, where the SEPP shall present a server certificate with a wildcard domain matching the returned telescopic FQDN.

Next Change

### 5.4.3 Telescopic FQDN to Foreign FQDN Mapping Procedure

This procedure is initiated by an NF Service Consumer (typically another SEPP) that has received a service request with an unknown first label of a telescopic FQDN. Typically, this SEPP may interact with other SEPPs in the same PLMN in order to determine if there is an existing mapping for a given label to an FQDN of a foreign FQDN; this procedure is only expected to be used when multiple SEPPs are deployed in a PLMN. The procedure is described in Figure 5.4.3-1 below.



Figure 5.4.3-1: Foreign FQDN to Telescopic FQDN Mapping Procedure

1. The NF Service Consumer issues an HTTP GET request towards another SEPP with a query parameter "telescopic-label" containing the first label of a given telescopic FQDN, whose mapping towards an FQDN of an NF in a foreign PLMN needs to be verified.

2a. On successful processing of the request, the responding SEPP shall respond to the NF Service Consumer with a "200 OK" status code and a response body that contains a JSON object of type "TelescopicMapping", containing as attribute "foreignFqdn", containing the FQDN of the NF in the foreign PLMN.

End of change