**3GPP TSG-SA3 Meeting #100e *S3-201843-r2***

**e-meeting, 17 -24 August 2020**

|  |
| --- |
| *CR-Form-v12.0* |
| **CHANGE REQUEST** |
|  |
|  | **33.501** | **CR** | **0916** | **rev** | **1** | **Current version:** | **15.9.0** |  |
|  |
| *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:***  | Update the N32-f context ID negotiation procedure |
|  |  |
| ***Source to WG:*** | Huawei, Hisilicion, Ericsson |
| ***Source to TSG:*** | S3 |
|  |  |
| ***Work item code:*** | 5GS\_Ph1-SEC |  | ***Date:*** | 21/07/2020 |
|  |  |  |  |  |
| ***Category:*** | **F** |  | ***Release:*** | Rel-15 |
|  | *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 canbe 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:*** | An LS S3-201479 that was sent to CT4 in SA3 #99e described that SA3 agrees to align TS 33.501 with TS 29.573 on the N32-f context IDs negotiation procedure. The changes in the TS 33.501 are required. |
|  |  |
| ***Summary of change:*** | Update the N32-f context ID negotiation procedure, and remove the N32-f precontext ID related operations. |
|  |  |
| ***Consequences if not approved:*** | Misallignment between SA3 and CT4 specification. |
|  |  |
| ***Clauses affected:*** | 3.1 |
|  |  |
|  | **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:*** |  |

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Start of the 1st change\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

#### 13.2.2.1 General

When the negotiated security mechanism to use over N32, according to the procedure in clause 13.5, is PRINS (described in clause 13.2), the SEPPs use the established TLS connection (henceforth referred to as N32-c connection) to negotiate the N32-f specific associated security configuration parameters required to enforce application layer security on HTTP messages exchanged between the SEPPs. A second N32-c connection is established by the receiving SEPP to enable it to not only receive but also send HTTP Requests.

The N32-c connection is used for the following purposes:

- Key agreement: The SEPPs independently export keying material associated with the first N32-c connection between them and use it as the pre-shared key for generating the shared session key required.

- Parameter exchange: The SEPPs exchange security related configuration parameters that they need to protect HTTP messages exchanged between the two Network Functions (NF) in their respective networks.

- Error handling: The receiving SEPP sends an error signalling message to the peer SEPP when it detects an error on the N32-f interface.

The following security related configuration parameters may be exchanged between the two SEPPs:

a. Modification policy. A modification policy, as specified in clause 13.2.3.4, indicates which IEs can be modified by an IPX provider of the sending SEPP.

b. Data-type encryption policy. A data-type encryption policy, as specified in 13.2.3.2, indicates which types of data will be encrypted by the sending SEPP.

c. Cipher suites for confidentiality and integrity protection, when application layer security is used to protect HTTP messages between them.

d. N32-f context ID. As specified in clause 13.2.2.4.1, N32-f context ID identifies the set of security related configuration parameters applicable to a protected message received from a SEPP in a different PLMN.

#### 13.2.2.2 Procedure for Key agreement and Parameter exchange

1. The two SEPPs shall perform the following cipher suite negotiation to agree on a cipher suite to use for protecting NF service related signalling over N32-f.

1a. The SEPP which initiated the first N32-c connection shall send a Security Parameter Exchange Request message to the responding SEPP including the initiating SEPP’s supported cipher suites. The cipher suites shall be ordered in initiating SEPP’s priority order. The SEPP shall provide an initiating SEPP’s N32-f context ID for the responding SEPP.

1b. The responding SEPP shall compare the received cipher suites to its own supported cipher suites and shall select, based on its local policy, a cipher suite, which is supported by both initiating SEPP and responding SEPP.

1c. The responding SEPP shall send a Security Parameter Exchange Response message to the initiating SEPP including the selected cipher suite for protecting the NF service related signalling over N32. The responding SEPP shall provide a responding SEPP’s N32-f context ID for the initiating SEPP

1d. Void.

2. The two SEPPs may perform the following exchange of Data-type encryption policies and Modification policies. Both SEPPs shall store protection policies sent by the peer SEPP:

2a. The SEPP which initiated the first N32-c connection shall send a Security Parameter Exchange Request message to the responding SEPP including the initiating SEPP’s Data-type encryption policies, as described in clause 13.2.3.2, and Modification policies, as described in clause 13.2.3.4.

2b. The responding SEPP shall store the policies if sent by the initiating SEPP.

2c. The responding SEPP shall send a Security Parameter Negotiation Response message to the initiating SEPP with the responding SEPP’s suite of protection policies.

2d. The initiating SEPP shall store the protection policy information if sent by the responding SEPP.

3. The two SEPPs shall exchange IPX security information lists that contain information on IPX public keys or certificates that are needed to verify IPX modifications at the receiving SEPP.

4. The two SEPPs shall export keying material from the TLS session established between them using the TLS export function. For TLS 1.2, the exporter specified in RFC 5705 [61] shall be used. For TLS 1.3, the exporter described in section 7.5 of RFC 8446 [60] shall be used. The exported key shall be used as the master key to derive session keys and IVs for the N32-f context as specified in clause 13.2.4.4.1.

5. The responding SEPP in the first N32-c connection shall now setup a second N32-c connection by establishing a mutually authenticated TLS connection with the peer SEPP.

6. The two SEPPs start exchanging NF to NF service related signalling over N32-f and shall keep the TLS session open for:

- any further N32-c communication that may occur over time while application layer security is applied to N32-f, or

- any further N32-c and N32-f communication, if TLS is used to protect N32-f.

 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Start of the 2nd change\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

##### 13.2.2.4.1 N32-f context ID

The N32-f context ID is used to refer to an N32-f context. The SEPPs shall create the N32-f context ID during the N32-c negotiation and use it over N32-f to inform the reveiving peer which security context to use for decryption of a received message.

 The initiating SEPP shall send the initiating SEPP’s N32-f context ID to the responding SEPP which the responding SEPP shall use to identify the N32-f connection with this initiating SEPP. Vice versa, the responding SEPP shall send the responding SEPP’s N32-f context ID to the initiating SEPP which the initiating SEPP shall use to identify the N32-f connection with this responding. To avoid collision of the N32-f context ID value, the SEPPs shall select the N32-f context ID as a random value during the exchange over N32-c.

During transfer of application data over N32-f, the SEPP shall include the N32-f context ID in a separate IE in the metadata part of the JSON structure, see clause 13.2.4.2. The receiving SEPP shall use this information to apply the correct key and parameters during decryption and validation.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*Start of the 3rd change\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

###### 13.2.4.3.1.2 metadata

The JSON object containing information added by the sending SEPP. It shall contain:

a) N32-f **message ID**: Unique identifier (64-bit integer) representing a HTTP Request/Response transaction between two SEPPs. The N32-f message ID is generated by the sending SEPP and included in the HTTP Request sent over the N32 interface. The receiving SEPP uses the same N32-f message ID when it responds back with a HTTP Response. The N32-f message ID is included in the metadata portion of the JSON structure.

b) **authorizedIPX** **ID**: String identifying the first hop IPX (cIPX or pIPX) that is authorized to update the message. This field shall always be present. When there is no IPX that is authorized to update, the value of this field is set to null. The sending SEPP selects one of the IPX providers from the list exchanged with the other SEPP during parameter exchange over N32-c and includes its identifier value in this field.

c) **N32-f context ID**: Unique identifier representing the N32-f context information used for protecting the message. This is exchanged during parameter exchange over N32-c (clause 13.2.2.4.1).

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*End of the changes\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*