**3GPP TSG-SA3 Meeting #124 S3-253445**

**Wuhan, China, 13 - 17 October 2025**

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| *CR-Form-v12.1* | | | | | | | | |
| **CHANGE REQUEST** | | | | | | | | |
|  | | | | | | | | |
|  |  | **CR** | **2191** | **rev** |  | **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:*** | Procedure to making some security parameters visible to RIs | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Source to WG:*** | Huawei, HiSilicon, CableLabs? | | | | | | | | | |
| ***Source to TSG:*** | S3 | | | | | | | | | |
|  |  | | | | | | | | | |
| ***Work item code:*** | RefinePRINS | | | | |  | ***Date:*** | | | 2025-10-13 |
|  |  | | | |  | |  | | |  |
| ***Category:*** | B |  | | | | | ***Release:*** | | | Rel-20 |
|  | *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:*** | | It was agreed in the Objective of the new WID that some security parameters that are relevant to RIs should be made visible to RIs. A solution is proposed to solve this problem. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Summary of change:*** | | Adding follwing procedures to make some security parameters visible to RIs:  1. SEPP's version should be negotiated to check whether SEPP supports to make some security parameters visible to RIs.  2. security parameters and the corresponding security parameters indication (used to indicate whether security parameters are included) are introduced.  3. Requirement of exchanging the security parameters visible to RIs is included.  4. The procedure to send security parameters to RIs is also introduced. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Consequences if not approved:*** | | RIs fail to receive security parameters. | | | | | | | | |
|  | |  | | | | | | | | |
| ***Clauses affected:*** | | 13.2.2.1, 13.2.2.2, 13.2.4.3, 13.2.4.7, 13.2.4.8 | | | | | | | | |
|  | |  | | | | | | | | |
|  | | **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:*** | |  | | | | | | | | |

\*\*\* 1st CHANGE \*\*\*

#### 13.2.4.3 Message reformatting in sending SEPP

##### 13.2.4.3.1 dataToIntegrityProtect

###### 13.2.4.3.1.1 clearTextEncapsulatedMessage

The clearTextEncapsulatedMessage is a JSON object that contains the non-encrypted portion of the original message.Specifically, it consists of the following objects:

1.a) Pseudo\_Headers – the JSON object that includes all the Pseudo Headers in the message.

- For HTTP Request messages, the object contains one entry for each of the ":method", ":path", ":scheme" and ":authority" pseudo headers. If the ":path" pseudoheader contains multiple parts separated by a slash (/) or includes a query parameter (following a "?"), an array is used to represent :path, with one element per part of the path (i.e. per "directory").

NOTE: This enables encryption of individual elements of the path (e.g. if SUPI is passed).

- For HTTP Response messages, the object contains the ":status" pseudo header.

1.b) HTTP\_Headers – the JSON object that includes all the Headers in the message.

All the headers of the request are put into a JSON array called HTTP\_Headers.Each entry contains a header name and value, where the value part can be an encoded index to the dataToIntegrityProtectAndCipher block, if the header value is encrypted.

1.c) Payload – the JSON object that includes the content of the payload of the HTTP message.

Each attribute or IE in the payload shall form a single entry in the Payload JSON object. If there is any attribute value that requires encryption, it shall be moved into the dataToIntegrityProtectAndCipher JSON object (clause 13.2.4.2), and the original value in this element shall be replaced by the index in the form {"encBlockIdx": <num>} where "num" is the index of the corresponding entry in the dataToIntegrityProtectAndCipher array.

1.d) Security parameters – the JSON object that includes the security parameters (as defined in 13.2.2.1) to be parsed by RIs.

The security parameters are included if the HTTP request message is the N32-f message before the procedure in clause 13.2.4.8.

###### 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 RI (e.g., cIPX or pIPX) that is authorized to update the message. This field shall always be present. When there is no RI that is authorized to update, the value of this field is set to null. The sending SEPP selects one of the RI 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).

\*\*\* 2nd CHANGE \*\*\*

#### 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.

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.

Alternatively to exchanging complete policies in steps 2a and 2c, the SEPPs may indicate a security profile.

NOTE: A security profile can for example include default modification policies and default data\_type encryption policies and/or a list of IEs to be protected, during the N32-c negotiation process. PRINS security profile specification is out of scope in 3GPP.

The security parameters that should be visible to the Roaming Intermediaries shall be sent over N32-f to the responding SEPP. The N32-f message that sends the security parameters shall be sent before the procedure in clause 13.2.4.8. The security parameters shall be placed in the clearTextEncapsulatedMessage block.

NOTE: The Roaming Intermediary can determine that the N32-f message includes security parameters based on the URL in the HTTP header.

3. The two SEPPs shall exchange Roaming Intermediary (RI) security information lists that contain information on RI public keys or certificates that are needed to verify RI 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. When the responding SEPP needs to initiate traffic, e.g., error reporting, in the reverse direction to the sending SEPP, 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.

NOTE: The second N32-c connection setup by the responding SEPP does not perform the negotiation of steps 1-4.

6. The two SEPPs start exchanging NF to NF service-related signalling over N32-f and tear down the N32-c connection. The SEPPs may initiate new N32-c TLS sessions for any further N32-c communication that may occur over time while application layer security is applied to N32-f.

\*\*\* 3rd CHANGE \*\*\*

#### 13.2.4.7 Message verification by the receiving SEPP

The receiving SEPP determines that the received message is an error message generated by the Roaming Hub as Roaming Intermediary based on the reformattedData IE.

If the received messages is not generated by a Roaming Hub :

- The receiving SEPP shall decrypt the JWE ciphertext using the shared session key and the following parameters obtained from the JWE object – Initialization Vector, Additional Authenticated Data value (clearTextEncapsulatedMessage in "aad") and JWE Authentication Tag ("tag").

- The receiving SEPP shall check the integrity and authenticity of the clearTextEncapsulatedMessage and the encrypted text by verifying the JWE Authentication Tag in the JWE object with the JWE AAD algorithm. The algorithm returns the decrypted plaintext (dataToIntegrityProtectAndCipher) only if the JWE Authentication Tag is correct.

- The receiving SEPP refers to the NF API in clearTextEncapsulatedMessage with values in the dataToIntegrityProtectAndCipher array.

- The receiving SEPP shall next verify RI provider updates, if included, by verifying the JWS signatures added by the Roaming Intermediaries. The SEPP shall verify the JWS signature, using the corresponding raw public key or certificate that is contained in the Roaming Intermediary’s security information list obtained during parameter exchange in the related N32-c connection setup or, alternatively, has been configured for the particular peer SEPP.

- The receiving SEPP shall then check that the raw public key or certificate of the JWS signature RI's Identity in the modifiedDataToIntegrity block matches to the RI provider referred to in the "authorizedIPX ID" field added by the sending SEPP, based on the information given in the RI provider security information list.

- The receiving SEPP shall check whether the modifications performed by the Roaming Intermediaries, i.e. cRI and pRI, were permitted by the respective modification policies. The receiving SEPP shall use the modification policy of the cRI obtained during parameter exchange in the related N32-c connection setup, and use the modification policy of pRI configured within the receiving SEPP.

- If this is the case, the receiving SEPP shall apply the patches in the Operations field in order, perform plausibility checks, and create a new HTTP request according to the "patched" clearTextEncapsulatedMessage.

- The receiving SEPP shall verify that the PLMN-ID contained in the headers and JSON elements of the incoming N32-f message, if any, matches the PLMN-ID in the related N32-f context. The receiving SEPP should also verify that the PLMN-ID contained in the incoming N32-f message matches the PLMN-ID in the trust anchor selected during the setup of the TLS VPN, if any. In case of a mismatch the SEPP should log the event.

NOTE 1: The above mismatch logging can be seen as an anomaly detection mechanism. It can both miss logging some anomalous events (false negatives) and log events that do not represent anomalies (false positives). False negatives occur, for example, when attack signalling carrying a particular PLMN-ID arrives over the TLS VPN or NDS/IP connection that is indeed associated with that PLMN-ID (e.g. generated by a compromised SEPP or an attacker with a fraudulently issued certificate), and false positives occur if signalling for a given PLMN-ID that previously arrived over a particular TLS VPN or NDS/IP connection starts arriving over a different one without this representing an attack. Such a change could be, for example, a result of contractual changes between PLMNs and RIs that were not configured before coming into effect. Dealing with false negatives and false positives is subject to operator policy and outside 3GPP scope.

If the received message is an error message generated by a Roaming Hub:

- The receiving SEPP shall check that the raw public key or certificate of the JWS signature RI's identity in the modifiedDataToIntegrityProtect block matches the adjacent Roaming Hub identity.

- The receiving SEPP dertermines the message in which the error occurred, based on the N32-f message ID.

- If the receiving SEPP determines from the error message that the Roaming Hub requires a modified request message, it can modify if allowed by the MNO's policy, and can resend the modified request message.

If security parameters exist, the receiving SEPP shall check whether such security parameters belong to the security parameters negotiated during the N32-c parameter exchange procedure.

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