**3GPP TSG-SA3 Meeting #102-e S3-210383-r1**

**e-meeting, 18 – 29 January 2021, Online**

|  |
| --- |
| *CR-Form-v12.0* |
| **CHANGE REQUEST** |
|  |
|  | **33.501** | **CR** | 1047 | **rev** | **1** | **Current version:** | **16.5.0** |  |
|  |
| *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:***  | Mirror: align the JSON format on encryption IE with CT4 in Rel16 |
|  |  |
| ***Source to WG:*** | Huawei, Hisilicon, Nokia, Nokia Shanghai Bell |
| ***Source to TSG:*** | S3 |
|  |  |
| ***Work item code:*** | 5GS\_Ph1-SEC |  | ***Date:*** | 2020-12-25 |
|  |  |  |  |  |
| ***Category:*** | **A** |  | ***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 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-15 (Release 15)Rel-16 (Release 16)Rel-17 (Release 17)Rel-18 (Release 18)* |
|  |  |
| ***Reason for change:*** | In SA3 #101e meeting, it was agreed to align the JSON format on encryption IE with CT4. Hence, the figure 13.2.4.2-1 Example of JSON representation of a reformatted HTTP message should be updated according to the CT4 specification. |
|  |  |
| ***Summary of change:*** | Figure update to align with CT4. |
|  |  |
| ***Consequences if not approved:*** | Figure is not align with CT4 specification. |
|  |  |
| ***Clauses affected:*** | 2, 13.2.4.2, 13.2.4.3.1.1, 13.2.4.3.2, 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:*** |  |

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

# 2 References

The following documents contain provisions which, through reference in this text, constitute provisions of the present document.

- References are either specific (identified by date of publication, edition number, version number, etc.) or non‑specific.

- For a specific reference, subsequent revisions do not apply.

- For a non-specific reference, the latest version applies. In the case of a reference to a 3GPP document (including a GSM document), a non-specific reference implicitly refers to the latest version of that document *in the same Release as the present document*.

[1] 3GPP TR 21.905: "Vocabulary for 3GPP Specifications".

[2] 3GPP TS 23.501: "System Architecture for the 5G System".

[3] 3GPP TS 33.210: "3G security; Network Domain Security (NDS); IP network layer security".

[4] IETF RFC 4303: "IP Encapsulating Security Payload (ESP)".

[5] 3GPP TS 33.310: "Network Domain Security (NDS); Authentication Framework (AF)".

[6] IETF RFC 4301: "Security Architecture for the Internet Protocol".

[7] 3GPP TS 22.261: "Service requirements for next generation new services and markets".

[8] 3GPP TS 23.502: "Procedures for the 5G System".

[9] 3GPP TS 33.102: "3G security; Security architecture".

[10] 3GPP TS 33.401: "3GPP System Architecture Evolution (SAE); Security architecture".

[11] 3GPP TS 33.402: "3GPP System Architecture Evolution (SAE); Security aspects of non-3GPP accesses".

[12] IETF RFC 5448: " Improved Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA')".

Editor’s note: This reference will be removed and references to it updated when the IETF updates the RFC and publishes a new RFC that supercedes this RFC.

[13] 3GPP TS 24.301: " Non-Access-Stratum (NAS) protocol for Evolved Packet System (EPS); Stage 3".

[14] 3GPP TS 35.215: " Specification of the 3GPP Confidentiality and Integrity Algorithms UEA2 & UIA2; Document 1: UEA2 and UIA2 specifications".

[15] NIST: "Advanced Encryption Standard (AES) (FIPS PUB 197)".

[16] NIST Special Publication 800-38A (2001): "Recommendation for Block Cipher Modes of Operation".

[17] NIST Special Publication 800-38B (2001): "Recommendation for Block Cipher Modes of Operation: The CMAC Mode for Authentication".

[18] 3GPP TS 35.221: " Specification of the 3GPP Confidentiality and Integrity Algorithms EEA3 & EIA3; Document 1: EEA3 and EIA3 specifications".

[19] 3GPP TS 23.003: "Numbering, addressing and identification".

[20] 3GPP TS 22.101: "Service aspects; Service principles".

[21] IETF RFC 4187: "Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA)".

[22] 3GPP TS 38.331: "NR; Radio Resource Control (RRC); Protocol specification".

[23] 3GPP TS 38.323: "NR; Packet Data Convergence Protocol (PDCP) specification".

[24] 3GPP TS 33.117: "Catalogue of general security assurance requirements".

[25] IETF RFC 7296: "Internet Key Exchange Protocol Version 2 (IKEv2)"

[26] Void

[27] IETF RFC 3748: "Extensible Authentication Protocol (EAP)".

[28] 3GPP TS 33.220: "Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA)".

[29] SECG SEC 1: Recommended Elliptic Curve Cryptography, Version 2.0, 2009. Available <http://www.secg.org/sec1-v2.pdf>

[30] SECG SEC 2: Recommended Elliptic Curve Domain Parameters, Version 2.0, 2010. Available at <http://www.secg.org/sec2-v2.pdf>

[31] 3GPP TS 38.470: "NG-RAN; F1 General aspects and principles".

[32] 3GPP TS 38.472: "NG-RAN; F1 signalling transport".

[33] 3GPP TS 38.474: "NG-RAN; F1 data transport".

[34] 3GPP TS 38.413: "NG-RAN; NG Application Protocol (NGAP)"

[35] 3GPP TS 24.501: "Non-Access-Stratum (NAS) protocol for 5G System (5GS); Stage 3".

[36] 3GPP TS 35.217: "Specification of the 3GPP Confidentiality and Integrity Algorithms UEA2 & UIA2; Document 3: Implementors' test data".

[37] 3GPP TS 35.223: "Specification of the 3GPP Confidentiality and Integrity Algorithms EEA3 & EIA3; Document 3: Implementors' test data".

[38] IETF RFC 5216: "The EAP-TLS Authentication Protocol".

[39] IETF RFC 4346: "The Transport Layer Security (TLS) Protocol Version 1.1".

[40] IETF RFC 5246: "The Transport Layer Security (TLS) Protocol Version 1.2".

[41] 3GPP TS 38.460: "NG-RAN; E1 general aspects and principles".

[42] Void.

[43] IETF RFC 6749: "OAuth2.0 Authorization Framework".

[44] IETF RFC 7519: "JSON Web Token (JWT)".

[45] IETF RFC 7515: "JSON Web Signature (JWS)".

[46] IETF RFC 7748: "Elliptic Curves for Security".

[47] IETF RFC 7540: " Hypertext Transfer Protocol Version 2 (HTTP/2)".

[48] IETF RFC 5280: "Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile".

[49] IETF RFC 6960: "X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP".

[50] IETF RFC 6066: "Transport Layer Security (TLS) Extensions: Extension Definitions".

[51] 3GPP TS 37.340: "Evolved Universal Terrestrial Radio Access (E-UTRA) and NR; Multi-connectivity; Stage 2".

[52] 3GPP TS 38.300: "NR; NR and NG-RAN Overall Description; Stage 2".

[53] 3GPP TS 33.122: "Security Aspects of Common API Framework for 3GPP Northbound APIs".

[54] 3GPP TS28.533: " Management and orchestration; Architecture framework".

[55] 3GPP TS28.531: "Management and orchestration of networks and network slicing; Provisioning".

[56] Void

[57] IETF RFC 7542: "The Network Access Identifier".

[58] IETF RFC 6083: " Datagram Transport Layer Security (DTLS) for Stream Control Transmission Protocol (SCTP)".

[59] IETF RFC 7516: "JSON Web Encryption (JWE)".

[60] IETF RFC 8446: "The Transport Layer Security (TLS) Protocol Version 1.3".

[61] IETF RFC 5705,"Keying Material Exporters for Transport Layer Security (TLS)".

[62] IETF RFC 5869 "HMAC-based Extract-and-Expand Key Derivation Function (HKDF)".

[63] NIST Special Publication 800-38D: "Recommendation for Block Cipher Modes of Operation: Galois Counter Mode (GCM) and GMAC".

[64] IETF RFC 6902: "JavaScript Object Notation (JSON) Patch".

[65] 3GPP TS 31.115: "Secured packet structure for (Universal) Subscriber Identity Module (U)SIM Toolkit applications.

[66] 3GPP TS 31.111: "Universal Subscriber Identity Module (USIM), Application Toolkit (USAT)".

[67] Internet draft draft-ietf-emu-rfc5448bis: "Improved Extensible Authentication Protocol Method for 3rd Generation Authentication and Key Agreement (EAP-AKA')".

[68] 3GPP TS 29.510: "5G System; Network function repository services".

[69] 3GPP TS 36.331: "Radio Resource Control (RRC); Protocol specification".

[70] 3GPP TS 29.505: "5G System; Usage of the Unified Data Repository services for Subscription Data; Stage 3".

[71] 3GPP TS 24.302: "Access to the 3GPP Evolved Packet Core (EPC) via non-3GPP access networks; Stage 3".

[72] 3GPP TS 23.216: "Single Radio Voice Call Continuity (SRVCC)".

[73] 3GPP TS 29.500: "Technical Realization of Service Based Architecture".

[74] 3GP TS 29.500: "5G System; Technical Realization of Service Based Architecture; Stage 3".

[75] IEEE TSN network aspects: see 3GPP TS 23.501 [2] references [95], [96], [97], [98], [104], and [107].

[76] Internet draft draft-ietf-emu-eap-tls13: "Using EAP-TLS with TLS 1.3"

[77] IETF RFC 8446: "The Transport Layer Security (TLS) Protocol Version 1.3".

[78] 3GPP TS 38.401: "NG-RAN; Architecture description".

[79] 3GPP TS 23.316: "Wireless and wireline convergence access support for the 5G System (5GS)"

[80] IEEE Std 802.11-2016 (Revision of IEEE Std 802.11-2012) - IEEE Standard for Information technology—Telecommunications and information exchange between systems Local and metropolitan area networks—Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications.

[81] IETF RFC 2410 "The NULL Encryption Algorithm and Its Use With IPsec".

[82] Void

[83] RFC 7858: "Specification for DNS over Transport Layer Security (TLS)".

[84] RFC 8310: "Usage Profiles for DNS over TLS and DNS over DTLS".

[85] RFC 4890: "Recommendations for Filtering ICMPv6 Messages in Firewalls".

[86] 3GPP TS 23.273: "5G System (5GS) Location Services (LCS); Stage 2".

[87] 3GPP TS 38.305: "Stage 2 functional specification of User Equipment (UE) positioning in NG-RAN".

[88] 3GPP TS 36.300: "Evolved Universal Terrestrial Radio Access (E-UTRA) and Evolved Universal Terrestrial Radio Access (E-UTRAN); Overall description; Stage 2".

[89] IANA: "Transport Layer Security (TLS) Parameters".

[90] RFC 2818: "HTTP Over TLS".

[91] 3GPP TS 33.535: "Authentication and key management for applications based on 3GPP credentials in the 5G System (5GS)".

[xx] 3GP TS 29.573: "5G System; Public Land Mobile Network (PLMN) Interconnection".

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

#### 13.2.4.2 Overall Message payload structure for message reformatting at SEPP

The SEPP reformats an HTTP message received from an internal Network Function into two temporary JSON objects that will be intput to JWE:

a. The **dataToIntegrityProtect**, containing information that is only integrity protected. It consists of the following:

- clearTextEncapsulationMessage: contains the complete original HTTP message, excluding attribute values which require encryption and, including the pseudo-header fields, HTTP headers and HTTP message body.

- metadata: contains SEPP generated information i.e. authorizedIPX ID, N32-f message ID and N32-f context ID.

b. The **dataToIntegrityProtectAndCipher**: contains attribute values of the original message that require both encryption and integrity protection.

For the details of JSON representation of a reformatted HTTP message, please refer to TS 29.573 [xx]

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

###### 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.

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

##### 13.2.4.3.2 dataToIntegrityProtectAndCipher

The dataToIntegrityProtectAndCipher is a JSON array that contains all the attribute values that require both encryption and integrity protection. Attribute values may come from any part of the original HTTP message – Pseudo\_Headers, HTTP\_Headers and Payload.

The JSON array shall contain one array entry per attribute value that needs encryption. Each array entry represents the value of the attribute to be protected, and the index in the array is used to reference the protected value within the dataToIntegrityProtect block. This associates each attribute in the dataToIntegrityProtectAndCipher block with the original attribute in the dataToIntegrityProtect block. This is needed to reassemble the original message at the receiving SEPP.

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

#### 13.2.4.7 Message verification by the receiving SEPP

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 data-type placement mapping table to re-construct the original reformatted message by updating corresponding entries in clearTextEncapsulatedMessage with values in the dataToIntegrityProtectAndCipher array.

The receiving SEPP shall next verify IPX provider updates, if included, by verifying the JWS signatures added by the intermediaries. The SEPP shall verify the JWS signature, using the corresponding raw public key or certificate that is contained in the IPX provider’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. It shall then check that the raw public key or certificate of the JWS signature IPX's Identity in the modifiedDataToIntegrity block matches to the IPX provider referred to in the "authorizedIPX ID" field added by the sending SEPP, based on the information given in the IPX provider security information list.

The receiving SEPP shall check whether the modifications performed by the intermediaries were permitted by the respective modification policies. 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 incoming N32-f message matches the PLMN-ID in the related N32-f context.

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

#### 13.2.4.8 Procedure

The following clause illustrates the message flow between the two SEPPs with modifications from cIPX and pIPX.



Figure 13.2.4.8-1 Message flow between two SEPPs

1. The cSEPP receives an HTTP request message from a network function. If the message contains a telescopic FQDN, the cSEPP removes its domain name from this FQDN to obtain the original FQDN as described in clause 13.1.

2. The cSEPP shall reformate the HTTP Request message as follows:

a. The cSEPP shall generate blocks (JSON objects) for integrity protected data and encrypted data, and protecting them:

The cSEPP shall encapsulate the HTTP request into a clearTextEncapsulatedMessage block containing the following child JSON objects:

- Pseudo\_Headers

- HTTP\_Headers with one element per header of the original request.

- Payload that contains the message body of the original request.

For each attribute that require end-to-end encryption between the two SEPPs, the attribute value is copied into a dataToIntegrityProtectAndCipher JSON object and the attribute's value in the clearTextEncapsulatedMessage is replaced by the index of attribute value in the dataToIntegrityProtectAndCipher block.

The cSEPP shall create a metadata block that contains the N32-f context ID, message ID generated by the cSEPP for this request/response transaction and next hop identity.

The cSEPP shall protect the dataToIntegrityProtect block and the dataToIntegrityProtectAndCipher block as per clause 13.2.4.4. This results in a single JWE object representing the protected HTTP Request message.

b. The cSEPP shall generate payload for the SEPP to SEPP HTTP message:

 The JWE object becomes the payload of the new HTTP message generated by cSEPP.

3. The cSEPP shall use HTTP POST to send the HTTP message to the first intermediary.

4. The first intermediary (e.g. visited network's IPX provider) shall create a new modifiedDataToIntegrityProtect JSON object with three elements:

a. The Operations JSON patch document contains modifications performed by the first intermediary as per RFC 6902 [64].

b. The first intermediary shall include its own identity in the Identity field of the modifiedDataToIntegrityProtect.

c. The first intermediary shall copy the "tag" element, present in the JWE object generated by the cSEPP, into the modifiedDataToIntegrityProtect object. This acts as a replay protection for updates made by the first intermediary.

The intermediary shall execute JWS on the modifiedDataToIntegrityProtect JSON object and append the resulting JWS object to the message.

5. The first intermediary shall send the modified HTTP message request to the second intermediary (e.g. home network's IPX) as in step 3.

6. The second intermediary shall perform further modifications as in step 4 if required. The second intermediary shall further execute JWS on the modifiedDataToIntegrityProtect JSON object and shall append the resulting JWS object to the message.

7. The second intermediary shall send the modified HTTP message to the pSEPP as in step 3.

 NOTE 1: The behaviour of the intermediaries is not normative, but the pSEPP assumes that behaviour for processing the resulting request.

8. The pSEPP receives the message and shall perform the following actions:

- The pSEPP extracts the serialized values from the components of the JWE object.

- The pSEPP invokes the JWE AEAD algorithm to check the integrity of the message and decrypt the dataToIntegrityProtectAndCipher block. This results in entries in the encrypted block becoming visible in cleartext.

- The pSEPP updates the clearTextEncapsulationMessage block in the message by replacing the references to the dataToIntegrityProtectAndCipher block with the referenced decrypted values from the dataToIntegrityProtectAndCipher block.

- The pSEPP then verifies IPX provider updates of the attributes in the modificationsArray. It checks whether the modifications performed by the intermediaries were permitted by policy.

 The pSEPP further verifies that the PLMN-ID contained in the message is equal to the "Remote PLMN-ID" in the related N32-f context.

- The pSEPP updates the modified values of the attributes in the clearTextEncapsulationMessage in order.

The pSEPP shall re-assemble the full HTTP Request from the contents of the clearTextEncapsulationMessage.

9. The pSEPP shall send the HTTP request resulting from step 8 to the home network's NF.

10.-18. These steps are analogous to steps 1.-9.

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