**3GPP TSG-SA WG3 Meeting #100-e *S3-201647***

**e-Meeting, 17-21 Aug 2020** *revision of S3-20wxyz*

|  |
| --- |
| *CR-Form-v12.0* |
| **CHANGE REQUEST** |
|  |
|  |  | **CR** | **0149** | **Rev** | 1 | **Current version:** |  |  |
|  |
| *For* [***HELP***](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:***  | [33.180] R16 Group regroup and user regroup security |
|  |  |
| ***Source to WG:*** | Motorola Solutions |
| ***Source to TSG:*** | S3 |
|  |  |
| ***Work item code:*** | MCXSec2 |  | ***Date:*** | 17-Aug-2020 |
|  |  |  |  |  |
| ***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 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:*** | Key management for group regroup with preconfigured group and user regroup with preconfigured group is needed. |
|  |  |
| ***Summary of change:*** | Add new clauses for group regroup with preconfigured group security and user regroup with preconfigured group security. |
|  |  |
| ***Consequences if not approved:*** | Group regroup with preconfigured group and user regroup with preconfigured group cannot be secured. |
|  |  |
| ***Clauses affected:*** | X (new), X.1 (new), X.2 (new), F.1.Y (new) |
|  |  |
|  | **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 change 1 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# 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 23.379: "Functional architecture and information flows to support Mission Critical Push To Talk (MCPTT); Stage 2".

[3] 3GPP TS 22.179: "Mission Critical Push To Talk (MCPTT); Stage 1".

[4] 3GPP TS 33.210: ''3G security; Network Domain Security (NDS); IP network layer security''.

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

[6] 3GPP TS 33.203: "3G security; Access security for IP-based services".

[7] 3GPP TS 33.179 Release 13: "Security of Mission Critical Push To Talk (MCPTT) over LTE".

[8] 3GPP TS 33.328: ''IP Multimedia Subsystem (IMS) media plane security''.

[9] IETF RFC 6507: ''Elliptic Curve-Based Certificateless Signatures for Identity-Based Encryption (ECCSI)''.

[10] IETF RFC 6508: ''Sakai-Kasahara Key Encryption (SAKKE)''.

[11] IETF RFC 6509: ''MIKEY-SAKKE: Sakai-Kasahara Key Encryption in Multimedia Internet KEYing (MIKEY)''.

[12] IETF RFC 3550: ''RTP: A Transport Protocol for Real-Time Applications''.

[13] IETF RFC 3711: "The Secure Real-time Transport Protocol (SRTP)".

[14] 3GPP TS 33.401: "3GPP System Architecture Evolution (SAE); Security architecture".

[15] 3GPP TS 23.228: "IP Multimedia Subsystem (IMS); Stage 2".

[16] 3GPP TS 33.222: "Generic Authentication Architecture (GAA); Access to network application functions using Hypertext Transfer Protocol over Transport Layer Security (HTTPS)".

[17] 3GPP TS 33.220: "Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture (GBA)".

[18] NIST FIPS 180-4: "Secure Hash Standard (SHS)".

[19] IETF RFC 6749: "The OAuth 2.0 Authorization Framework".

[20] IETF RFC 6750: "The OAuth 2.0 Authorization Framework: Bearer Token Usage".

[21] OpenID Connect 1.0: "OpenID Connect Core 1.0 incorporating errata set 1", <http://openid.net/specs/openid-connect-core-1_0.html>.

[22] IETF RFC 3830: "MIKEY: Multimedia Internet KEYing".

[23] IETF RFC 3602: "The AES-CBC Cipher Algorithm and Its Use with IPsec".

[24] IETF RFC 4771: "Integrity Transform Carrying Roll-Over Counter for the Secure Real-time Transport Protocol (SRTP)".

[25] IETF RFC 6043: "MIKEY-TICKET: Ticket-Based Modes of Key Distribution in Multimedia Internet KEYing (MIKEY)".

[26] IETF RFC 7714: ''AES-GCM Authenticated Encryption in the Secure Real-time Transport Protocol (SRTP)''.

[27] W3C: "XML Encryption Syntax and Processing Version 1.1", <https://www.w3.org/TR/xmlenc-core1/>.

[28] W3C: "XML Signature Syntax and Processing (Second Edition)", <http://www.w3.org/TR/xmldsig-core/>.

[29] IETF RFC 5905: "Network Time Protocol Version 4: Protocol and Algorithms Specification".

[30] IETF RFC 5480: "Elliptic Curve Cryptography Subject Public Key Information".

[31] IETF RFC 6090: "Fundamental Elliptic Curve Cryptography Algorithms".

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

[33] IETF RFC 7662: "OAuth 2.0 Token Introspection".

[34] IETF RFC 3394: "Advanced Encryption Standard (AES) Key Wrap Algorithm".

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

[36] 3GPP TS 23.280: "Common functional architecture to support mission critical services; Stage 2".

[37] 3GPP TS 23.281: "Functional architecture and information flows for mission critical video; Stage 2".

[38] 3GPP TS 23.282: "Functional model and information flows for Mission Critical Data".

[39] 3GPP TS 23.002: "Network Architecture".

[40] IETF RFC 2045: "Multipurpose Internet Mail Extensions (MIME) Part One: Format of Internet Message Bodies".

[41] IETF RFC 2392: "Content-ID and Message-ID Uniform Resource Locators".

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

[43] IETF RFC 5116: "An Interface and Algorithms for Authenticated Encryption".

[45] IETF RFC 7521: "Assertion Framework for OAuth 2.0 Client Authentication and Authorization Grants".

[46] IETF RFC 7523: "JSON Web Token (JWT) Profile for OAuth 2.0 Client Authentication and Authorization Grants".

[47] 3GPP TS 22.280: " Mission Critical Services Common Requirements; Stage 1".

[48] 3GPP TS 23.283: " Mission Critical Communication Interworking with Land Mobile Radio Systems; Stage 2".

[49] 3GPP TS 24.379: "Mission Critical Push To Talk (MCPTT) call control; Protocol specification."

[50] 3GPP TS 24.282: "Mission Critical Data (MCData) signalling control; Protocol specification. "

[51] IETF RFC 3711 Errata ID 3712, <https://www.rfc-editor.org/errata/eid3712>.

[52] IANA: "Multimedia Internet KEYing (MIKEY) Payload Name Spaces", <https://www.iana.org/assignments/mikey-payloads/mikey-payloads.xhtml>.

[53] IETF RFC 7636: "Proof Key for Code Exchange by OAuth public clients".

[zz] IETF RFC 4868: "Using HMAC-SHA-256, HMAC-SHA-384, and HMAC-SHA-512 with IPsec".

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* END of change 1 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* START of change 2 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

## 3.2 Abbreviations

For the purposes of the present document, the abbreviations given in 3GPP TR 21.905 [1] and the following apply. An abbreviation defined in the present document takes precedence over the definition of the same abbreviation, if any, in 3GPP TR 21.905 [1].

CMS Configuration Management Server

CS Crypto Session

CSB-ID Crypto Session Bundle Identifier

CSC Common Services Core

CSK Client-Server Key

CSK-ID Client-Server Key Identifier

DPCK MCData Payload Cipher Key

DPPK MCData Payload Protection Key

DPPK-ID MCData Payload Protection Key Identifier

GBA Generic Bootstrapping Architecture

GMK Group Master Key

GMK-ID Group Master Key Identifier

GMS Group Management Server

GUK-ID Group User Key Identifier

IdM Identity Management

IdMS Identity Management Server

InK Integrity Key

InK-ID Integrity Key Identifier

InterKMRec Interworking Key Management Record

InterKMRec-ID Interworking Key Management Record Identifier

InterSD Interworking Security Data

IWF InterWorking Function

JSON JavaScript Object Notation

JWS JSON Web Signature

JWT JSON Web Token

KDF Key Derivation Function

KFC Key for Floor Control

KFC-ID Key for Floor Control Identifier

KMS Key Management Server

MBCP Media Burst Control Protocol

MCData Mission Critical Data

MCPTT Mission Critical Push to Talk

MCVideo Mission Critical Video

MCX Mission Critical Services

MKFC Multicast Key for Floor Control

MSCCK MBMS subchannel control key

MSRP Message Session Relay Protocol

MuSiK Multicast Signalling Key

MKI Master Key Identifier

NGMI Next Generation Mobile Intelligence

NTP Network Time Protocol

NTP-UTC Network Time Protocol – Coordinated Universal Time

OIDC OpenID Connect

PCK Private Call Key

PCK-ID Private Call Key Identifier

PKCE Proof Key for Code Exchange

PSK Pre-Shared Key

RGK Regroup Key

RGK-SEED Regroup Key Seed

SEG Security Gateway

SeGy Security Gateway

SPK Signalling Protection Key

SRTCP Secure Real-Time Transport Control Protocol

SRTP Secure Real-Time Transport Protocol

SSRC Synchronization Source

TBCP Talk Burst Control Protocol

TGK Traffic Generating Key

TrK KMS Transport Key

TrK-ID KMS Transport Key Identifier

UID User Identifier for MIKEY-SAKKE (referred to as the 'Identifier' in RFC 6509 [11])

XPK XML Protection Key

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* END of change 2 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* START of change 3 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

# X Group regroup and user regroup security

## X.1 Group regroup with preconfigured group

Group regroup with preconfigured group is defined in TS 23.379 [2]. The basis of this feature is to allow an authorized MC user to dynamically create a temporary group consisting of a set of predefined groups (i.e. all the members of the predefined groups become members of this new regroup group).

Security for group regroup with predefined group is based on a GMK obtained through a predefined group configuration. The group configuration to be used (i.e. the GMK) is identified in the initial preconfigured regroup request message. All users that participate in a secure group regroup call must have the predefined group configuration and GMK.

While some users may not be invited into the group regroup, they may still have access to the group configuration containing the GMK and therefore key diversity for the group regroup is imperative. In order to create key diversity, a randomly generated 128-bit Regroup Key Seed (RGK-SEED) shall be included in the preconfigured regroup request message. The combination of the preconfigured GMK and RGK-SEED shall create the Regroup Key (RGK) for the regroup group session as defined in F.1.Y.

The RGK-SEED is provided by the MC service server to those users invited to the regroup group call using either unicast or multicast on the downlink. To protect the RGK-SEED during transit, the preconfigured regroup request message from the initiating MC client to the MC service server carries the RGK-SEED encrypted on the initiating client’s CSK. The MC service server decrypts the RGK-SEED from the initiating client’s CSK and in turn, delivers the preconfigured regroup request message to each participating MC client. For unicast downlink, the MC service server protects the RGK-SEED with each target client’s individual CSK. For multicast downlink, the MC service server protects the RGK-SEED with the MuSiK.

NOTE: Multicast delivery of the RGK-SEED does not provide the same level of security compared to unicast delivery. With multicast, any UE that has the MuSiK and access to the regroup group configuration may see the broadcast of the RGK-SEED and has the potential to generate the RGK whether invited to the call or not.

The MC service server shall store the RGK-SEED for the duration of the group regrouping so that new users added to the regroup group or late-entry of users to the regroup regroup may receive the seed. The MC service server shall securely key manage the RGK-SEED as described in clause 5.11.1.

Figure X.1-1 provides an example showing unicast and multicast distribution of the RGK-SEED for a preconfigured group regroup call.



Figure X.1-1: Key establishment for group regroup with preconfigured group

1. The authorized user of MC client 1 initiates the group regroup procedure.

2. MC client 1 sends the preconfigured regroup request message to the MC service server. The preconfigured regroup request message contains the set of group regroup information listed in TS 23.379 [2] together with a randomly generated RGK-SEED protected on the initiating client’s CSK.

3. The MC service server performs the group regroup validation steps described in TS 23.379 [2]. In addition, the MC service server decrypts the RGK-SEED from the initiating client’s CSK and securely stores it.

4. The MC service server sends a unicast preconfigured regroup request message to MC client 2. The preconfigured regroup request message includes the RGK-SEED encrypted on client 2’s CSK.

5. The MC service server sends a multicast preconfigured regroup request message to MC clients 3, 4 and 5. The multicast preconfigured regroup request message includes the RGK-SEED encrypted on multicast MuSiK.

6a. MC client 2 notifies the user of the regrouping.

6b. MC clients 3, 4 and 5 notify their users of the regrouping.

7. MC client 2 may optionally send a preconfigured regroup response to the MC service server to acknowledge the regrouping action.

NOTE: MC clients 3, 4 and 5 received the preconfigured regroup request message via multicast and therefore do not send a preconfigured regroup response to the MC service server.

8. The MC service server affiliates the regrouped MC clients to the regroup group.

9. The MC service server sends a preconfigured regroup response to MC client 1.

10a. MC client 1 calculates the RGK to be used for the regroup group session. This step may happen any time after step 1.

10b. MC client 2 calculates the RGK to be used for the regroup group session. This step may happen any time after step 6a.

10c. MC clients 3, 4 and 5 calculate the RGK to be used for the regroup group session. This step may happen any time after step 6b.

## X.2 User regroup with preconfigured group

User regroup with preconfigured group is defined in TS 23.379 [2]. The basis of this feature is to allow the MC operator to dynamically create a temporary group consisting of a specified set of users (i.e. all the users identified in the user regroup become members of this new temporary group).

Security for user regroup with predefined group is based on a GMK obtained through a predefined group configuration. The group configuration to be used (i.e. the GMK) is identified in the initial preconfigured regroup request message. All users that participate in a secure user regroup call must have the predefined group configuration and GMK.

While some users may not be invited into the user regroup call, they still may have access to the group configuration containing the GMK and therefore key diversity for the user regroup is imperative. In order to create key diversity, a randomly generated 128-bit Regroup Key Seed (RGK-SEED) shall be included in the preconfigured regroup request message. The combination of the preconfigured GMK and RGK-SEED shall create the Regroup Key (RGK) for the regroup group or user regroup session as described in F.1.Y.

The RGK-SEED is provided to those users invited to the user regroup call using unicast on the downlink. Multicast is not used on the downlink for user regroup. To protect the RGK-SEED during transit, the preconfigured regroup request message from the initiating MC client to the MC service server carries the RGK-SEED encrypted on the initiating client’s CSK. The MC service server decrypts the RGK-SEED from the initiating client’s CSK and in turn, delivers the preconfigured regroup request message unicast to each participating MC client with the RGK-SEED encrypted on each target client’s individual CSK.

The MC service server temporarily stores the RGK-SEED for the duration of the user regrouping so that new users added to the user regroup or late-entry of users may receive the seed. The MC service server shall securely key manage the RGK-SEED as described in clause 5.11.1.

Figure X.2-1 provides an example showing the distribution of the RGK-SEED for a preconfigured user regroup call.



Figure X.2-1: Key establishment for user regroup with preconfigured group

1. The authorized user of MC client 1 initiates the user regroup procedure.

2. MC client 1 sends the preconfigured regroup request message to the MC service server. The preconfigured regroup request message contains the set of user regroup information listed in TS 23.379 [2] together with a randomly generated RGK-SEED protected on the initiating client’s CSK.

3. The MC service server performs the validation steps described in TS 23.379 [2]. In addition, the MC service server decrypts the RGK-SEED from the initiating client’s CSK and securely stores it.

4a. The MC service server sends a unicast preconfigured regroup request message to MC client 2. The preconfigured regroup request message includes the RGK-SEED encrypted on client 2’s CSK.

4b. The MC service server sends a unicast preconfigured regroup request message to MC client 3. The preconfigured regroup request message includes the RGK-SEED encrypted on client 3’s CSK.

5a. MC client 2 notifies the user of the regrouping.

5b. MC client 3 notifies the user of the regrouping.

6a. MC client 2 may send a preconfigured regroup response to the MC service server to acknowledge the regrouping action.

6b. MC client 3 may send a preconfigured regroup response to the MC service server to acknowledge the regrouping action.

7. The MC service server affiliates the regrouped MC clients to the user regroup group.

8. The MC service server sends a preconfigured regroup response to MC client 1.

9a. MC client 1 calculates the RGK to be used for the user regroup session. This step may happen any time after step 1.

9b. MC client 2 calculates the RGK to be used for the user regroup session. This step may happen any time after step 5a.

9c. MC client 3 calculates the RGK to be used for the user regroup session. This step may happen any time after step 5b.\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* END of change 3 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* START of change 4 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

### F.1.Y Calculation of regroup key for preconfigured groups

The following key deriviation function, as defined in rfc4868 [zz], shall be used for calculating the RGK key;

RGK = PRF-HMAC-SHA-256 (Key, Data)

The GMK of the preconfigured group shall be the input ‘Key’ and the RGK-SEED shall be the input ‘Data’. The 128 least significant bits of the 256 bits of the key derivation function output shall be used as the RGK.

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* END of change 4 \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*