3GPP TSG-T (Terminals) Meeting #26 ***TP-040274***
Athens, Greece
8 - 10 December 2004

3GPP TSG SA WG3 Security ó SA3#36 **S3-041134**
November 23-26, 2004
Shenzhen, China

Title: LS on MBMS work progress
Release: Rel-6
Work Item: MBMS

Source: SA3
To: TSG T, T3

Contact Person:
Name: **Mireille Pauliac**
Tel. Number: +33 442365441
E-mail Address: mireille.pauliac@gemplus.com

Attachments: S3-041125, S3-041127

---

1. Overall Description:

SA3 are currently finalising MBMS TS 33.246. Several CRs have been agreed at SA3#36 meeting to complete SA3 MBMS work on TS 33.246. SA3 would like to inform T3 and TSG T that some of these CRs impact the MBMS functionality specified in T3. Further T3ís work is needed to complete the Rel-6 MBMS service.

2. Actions:

To T3 group.

ACTION: SA3 kindly ask T3 to consider the attached SA3 CRs on MBMS in order to finalise the work in T3.

3. Date of Next TSG SA WG 3 Meetings:

TSG-SA3 Meeting #37	21-25 February 2005	Sophia-Antipolis, France
TSG-SA3 Meeting #38	25-29 April 2005	Switzerland (TBC)

3GPP TSG-SA3 Meeting #36 ***Tdoc ⌘S3-04111825***
Shenzhen, China, 23 ñ 26 November 2004

CR-Form-v7

CHANGE REQUEST

⌘ **33.246** **CR** **010** ⌘**rev** **3** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: | UICC apps⌘ [X] ME [X] Radio Access Network [ ] Core Network [X]

Title:	⌘	MBMS Transport of salt
Source:	⌘	SA WG3
Work item code:	⌘	MBMS
Date:	⌘	23/11/2004
Category:	⌘	**C**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction 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.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	The protection of the MBMS traffic will not meet the commonly required design goal of having a security level equivalent to the key size.
Summary of change:	⌘	The salt needed by SRTP is sent in the KEMAC payload of the MIKEY message containing the MTK.
Consequences if not approved:	⌘	The protection of the MBMS traffic may be vulnerable to pre-computation attacks.

Clauses affected:	⌘	6.4.5.3, 6.4.6.2, 6.5.4, D.3

		Y	N			
Other specs Affected:	⌘	X		Other core specifications	⌘	TS 31.102
			X	Test specifications		
			X	O&M Specifications		

Other comments:	⌘	

__FIRST_CHANGE__

6.4.5.3 MTK message structure

The structure of the MIKEY message carrying a MTK key is depicted in Figure 6.7. The actual key that is delivered is kept in the KEMAC payload. If MTK is to be used for streaming protection, then a 112 bit salt shall be added to the KEMAC payload in addition to the MTK. The network identity payloads (IDi) shall be used in MTK transport messages.

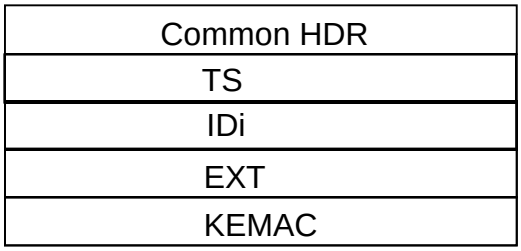

Figure 6.7: The logical structure of the MIKEY message used to deliver MTK

__SECOND_CHANGE__

6.4.6.2 MTK MIKEY Message Reception

When the MIKEY message arrives at the ME, the processing proceeds following the steps below (basically following section 5.3 of [9]).

1. The Data Type field of the common MIKEY header (HDR) is examined, and if it indicates an MTSK delivery, the MSK ID is extracted from the Extension Payload.
2. The Timestamp Payload is checked, and the message is discarded if the counter is larger or equal to the current MIKEY replay counter associated with the given MSK (the counter value is retrieved from MGV-S). To avoid issues with wrap around of the ID fields ``smaller than¥¥ should be in the sense of RFC1982 [10].
3. If the MTK ID extracted from the Extension payload is less than or equal to the current MTK ID (kept in the ME), the message shall be discarded.
4. The message is transported to MGV-F for further processing, cf 6.5.3.
5. The MGV-F replies success (i.e. sending the MTK and salt if available) or failure.

__THIRD_CHANGE__

6.5.4 MTK validation and derivation

When the MGV-F receives the MIKEY message, it first determines the type of message by reading the Data Type field in the common header. If the key inside the message is an MTK, MGV-F retrieves the MSK with the ID given by the Extension payload.

It is assumed that the MBMS service specific data, MSK and the sequence number SEQs, have been stored within a secure storage (MGV-S). Both MSK and SEQs were transferred to the MGV-S with the execution of the MSK update procedures. The initial value of SEQs is determined by the service provider.

The MGV-F shall only calculate and deliver the MBMS Traffic Keys (MTK) to the ME if the ptm-key information is deemed to be fresh.

The MGV-F shall compare the received SEQp, i.e. MTK ID from the MIKEY message with the stored SEQs. If SEQp is equal or lower than SEQs then the MGV-F shall indicate a failure to the ME. If SEQp is greater than SEQs then the MGV-F shall calculate the MAC as defined in [9] using the received MIKEY message and MSK as input. This MAC is compared with the MAC of the KEMAC payload in the MIKEY message. If the MAC verification is unsuccessful, then the MGV-F will indicate a failure to the ME. If the MAC verification is successful, then the MGV-F shall update SEQs with SEQp value and start the generation of MTK. The MGV-F provides the MTK to the ME.

The MGV-F shall update in MGV-S the counter value in the Time Stamp payload associated with the corresponding MSK ID.

In the case of streaming, SRTP requires a master key and a master salt. The MTK is used as master key, and the salt in the KEMAC payload is used as master salt.

NOTE: MIKEY includes functionality to derive further keys from MTK if needed by the security protocol. The key derivation is defined in section 4.1.3 of MIKEY [9].

__FOURTH_CHANGE__

D.3 MTK generation and validation

This procedure is part of the MTK generation and validation function as described in clause 6.5.4 (MTK validation and derivation).

The ME receives the MIKEY message (containing Header, Time stamp, Network ID, Key Group ID, MSK ID, MTK ID = SEQp, MSK_C[MTK||Salt (if salt is available)] and MAC). After performing some validity checks, the ME sends the whole message to the UICC. The UICC computes the MGV-F function as described in clause 6.5. (Validation and key derivation functions in MGV-F). After successful MGV-F procedure the UICC returns the MTK.

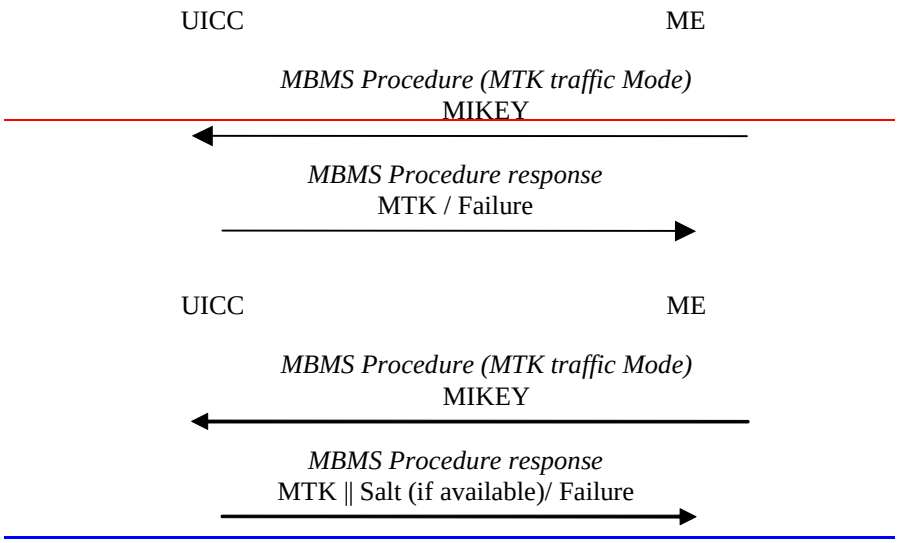

Figure D.3: MTK Generation and Validation

pop3GPP TSG-SA3 Meeting #36
Shenzen, China, 23 ñ 26 November

Tdoc ⌘ S3-041127

CR-Form-v7

CHANGE REQUEST

⌘ **33.246** **CR** **033** ⌘ **rev** **1** ⌘ Current version: **6.0.0** ⌘

*For **HELP** on using this form, see bottom of this page or look at the pop-up text over the ⌘ symbols.*

Proposed change affects: | UICC apps⌘ [X] ME [X] Radio Access Network [ ] Core Network [X]

Title:	⌘	Handling of MBMS identities and definition completion/modification
Source:	⌘	SA WG3
Work item code:	⌘	MBMS
Date:	⌘	25/11/2004
Category:	⌘	**C**
Release:	⌘	Rel-6

Use one of the following categories:
F *(correction)*
A *(corresponds to a correction 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.

Use one of the following releases:

2	*(GSM Phase 2)*
R96	*(Release 1996)*
R97	*(Release 1997)*
R98	*(Release 1998)*
R99	*(Release 1999)*
Rel-4	*(Release 4)*
Rel-5	*(Release 5)*
Rel-6	*(Release 6)*

Reason for change:	⌘	Specify the missing identification and how the identities are used and transported in the different field of the MIKEY delivery and response messages.
Summary of change:	⌘	MUK, MRK IDs are undefined and their use in MIKEY fields is unspecified. MTK, MSK, and Network ID are put into MIKEY extension field. MSK ID is redefined combining former MSK and Key Group IDs. Removed ID_i from the response message, since it is not needed and is not present in the MIKEY specification. The Network ID (MCC||MNC) is moved from the IDi field of MIKEY messages to the extension payload to obtain a uniform message structure for both MSK and MTK delivery. Network ID is renamed to Key Domain ID The Key IDs are carried as follows: For MSK delivery messages: MUK ID via IDi and IDr, MSK ID and Key Domain via extention payload. For MTK delivery messages: IDi and IDr and CSB are not used. MTK-ID, MSK-ID and Key Domain ID via extention payload. Adds dependency to IETF internet draft <draft-carrara-newtype-keyid-00.txt>
Consequences if not approved:	⌘	

Clauses affected:	⌘	2, 3.4 (New), 6.1, 6.3.2.1, 6.3.3.1, 6.4.1, 6.4.2, 6.4.4, 6.4.5, 6.4.6, 6.5.2, 6.5.3, 6.5.4, 6.6.2.1, Annex D

Other specs affected:	⌘	Y	N		⌘
		X		Other core specifications	TS 31.102
			X	Test specifications	
			X	O&M Specifications	

Other comments:	⌘	

********** NEXT CHANGE**********

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 22.146: "Multimedia Broadcast/Multicast Service; Stage 1".

[3] 3GPP TS 23.246: "Multimedia Broadcast/Multicast Service (MBMS); Architecture and Functional Description".

[4] 3GPP TS 33.102: "3G Security; Security Architecture".

[5] 3GPP TS 22.246: "MBMS User Services".

[6] 3GPP TS 33.220: "Generic Authentication Architecture (GAA); Generic Bootstrapping Architecture".

[7] 3GPP TS 31.102: "Characteristics of the USIM application".

[8] IETF RFC 2617 "HTTP Digest Authentication".

[9] IETF RFC 3830 "MIKEY: Multimedia Internet KEYing"

[10] IETF RFC 1982 "Serial Number Arithmetic".

[11] IETF RFC 3711 "Secure Real-time Transport Protocol".

[12] 3GPP TS 43.020: "Security related network functions".

[13] IETF internet draft "The Key ID Information Type for the General Extension Payload in MIKEYî <draft-carrara-newtype-keyid-00.txt>

** NEXT CHANGE ***

3.4 Conventions

All data variables in this specification are presented with the most significant substring on the left hand side and the least significant substring on the right hand side. A substring may be a bit, byte or other arbitrary length bitstring. Where a variable is broken down into a number of substrings, the leftmost (most significant) substring is numbered 0, the next most significant is numbered 1, and so on through to the least significant.

** NEXT CHANGE ***

6.1 Using GBA for MBMS

TS 33.220 [6] (Generic Bootstrapping Architecture) is used to agree keys that are needed to run an MBMS Multicast User service. MBMS imposes the following requirements on the MBMS capable UICCs and MEs:

- a UICC that contains MBMS key management functions shall implement GBA_U;
- a ME that supports MBMS shall implement GBA_U and GBA_ME, and shall be capable of utilising the MBMS key management functions on the UICC.

Before a user can access an MBMS User service, the UE needs to share GBA-keys with the BM-SC. If no valid GBA-keys are available at the UE, the UE shall perform a GBA run with the BSF of the home network as described within clause 5 of TS 33.220 [6]. The BM-SC will act as a NAF according to TS 33.220 [6].

The MSKs for an MBMS User service shall be stored on either the UICC or the ME. Storing the MSKs on the UICC requires a UICC that contains the MBMS management functions (and that is GBA aware) and requires that the BM-SC is GBA_U aware. As a result of the GBA_U run in these circumstances, the BM-SC will share a key Ks_ext_NAF with the ME and share a key Ks_int_NAF with the UICC. This key Ks_int_NAF is used by the BM-SC and the UICC as the key MUK to protect MSK deliveries to the UICC as described within clause 6.3. The key Ks_ext_NAF is used as the key MRK within the protocols as described within clause 6.2.

NOTE: A run of GBA_U on a GBA aware UICC will not allow the MSKs to be stored on the UICC, if the MBMS management functions are not present on the UICC.

In any other circumstance, a run of GBA results in the BM-SC sharing a key Ks_(ext)_NAF with the ME. This key Ks_(ext)_NAF is used by the BM-SC and the ME to derive the key MUK and the key MRK (MBMS Request Key). The key MUK is used to protect MSK deliveries to the ME as described within clause 6.3. The key MRK is used to authenticate the UE towards the BM-SC within the protocols as described within clause 6.2.

The MUK is identified by the combination of B-TID and NAF-ID and the MRK is defined by B-TID, where B-TID and NAF-ID are defined as specified in TS 33.220 [6].

*******NEXT CHANGE*********

6.3.2.1 MSK identification

Every MSK is uniquely identifiable by its ~~Network~~ Key Domain ID and~~, Key Group ID and~~ MSK ID

where

~~Network~~ Key Domain ID = MCC || MNC and is 3 bytes long. ~~It is carried in the IDi payload in MIKEY message~~

~~Key Group ID is 2 bytes long and is used to group keys together in order to allow redundant MSKs to be deleted. It is carried in the CSB ID field of MIKEY common header.~~

MSK ID is 4~~2~~ bytes long and with byte 0 and 1 containing the Key Group part, and byte 2 and 3 containing the Key Number part. The Key Number part is used to distinguish MSKs that have the same ~~Network~~ Key Domain ID and Key Group ~~ID~~part. Key Group part is used to group keys together in order to allow redundant MSKs to be deleted. The MSK ID ~~It~~ is carried in the extension payload ~~MSK ID field~~ of MIKEY extension payload.

NOTE: It needs to be ensured that the Key Group parts are unique within an operator, i.e. two BM-SCs within an operator shall not use the same Key Group value.

If the UE receives an MSK and already contains two other MSKs under the same ~~Network~~ Key Domain ID and Key Group part~~ID~~, then the UE shall delete the older of these two MSKs.

Editor's Note: The handling of MSKs may need some enhancement to cover download services, where the MSK is fetched after the UE has received the encrypted data.

********** NEXT CHANGE**********

6.3.3.1 MTK identification

Every MTK is uniquely identifiable by its ~~Network~~ Key Domain ID, ~~Key Group ID,~~ MSK ID and MTK ID

where

Key Domain~~Network~~ ID~~, Key Group ID~~ and MSK ID are as defined in clause 6.3.2.1.

Editor's Note: The format of MTK is ffs.

********** NEXT CHANGE**********

6.4 MIKEY message creation and processing in the ME

Editorís note: The need for salting keys in processing of MIKEY messages is for further study.

6.4.1 General

MIKEY is used to transport the MSKs and MTKs from the BM-SC to the UE. Clauses 6.4.2, 6.4.3, 6.4.4 and 6.4.5 describe how to create the MIKEY messages, while clause 6.4.6 describes the initial processing by the ME on these messages. The final processing is done by the MBMS key Generation and Validation Function (MGV-F) and is described in clause 6.5.

MIKEY shall be used with pre-shared keys as described in RFC 3830 [9].

To keep track of MSKs and MTKs, a new Extension Payload (EXT) [13] is added to MIKEY. The Extension Payload can contain the key types and identities of MSK and the MTK and Key Domain ID (see clause 6.3.2 and 6.3.3).

6.4.2 MIKEY common header

~~MIKEY shall be used with pre-shared keys as described in RFC 3830 [9].~~

MSKs shall be carried in MIKEY messages. ~~with a Data Type value of 0x07 in the MIKEY common header that signals that the message contains an MBMS MSK. This allows legacy MIKEY implementations to discard the message early in the processing stage.~~ The messages are sent point-to-point between the BM-SC and each UE. The messages use the MUK shared between the BM-SC and the UE as the pre-shared secret in MIKEY.

Once the MSK is in place in the UE, the UE can make use of the multicast MTK messages sent by the BM-SC. The MTK is carried in messages conforming to the structure defined by MIKEY and use the MSK as the pre-shared secret. ~~A Data Type value of 0x08 is used in the MIKEY common header to signal that the message contains an MBMS MTK.~~

~~To keep track of MSKs and MTKs, a new Extension Payload (EXT) is added to MIKEY. The Extension contains the identities of MSKs and the MTKs (see clause 6.3.2 and 6.3.3).~~

If the BM-SC requires an ACK for an MSK key update message this is indicated by setting the V-bit in the MIKEY common header. The UE shall then respond with a MIKEY message containing the verification payload. In the case the server does not receive an ACK, normal reliability constructions can be used, e.g., start a timer when the message is sent and then resend the message if no ACK is received before the timer expires.

The CSB ID field of MIKEY common header ~~shall carry the Key Group ID.~~is not used.

********** NEXT CHANGE**********

6.4.4 General extension payload

The MSK and MTK shall be delivered in messages that conform to the structure defined in RFC 3830 [9] (MIKEY). To be able to keep track of the key~~s~~ that is delivered in the message, a ~~new~~ general Extension Payload (EXT) with Type field value x is ~~defined~~ used that conforms to the structure defined in [13] ~~section 6.15 of RFC 3830 [9] (MIKEY)~~.

Editorís Note: The type value will be replaced by value requested from IANA.

The EXT includes a Key Domain ID and one or two Key Type ID sub-payloads depending on the message. These are used as follows.

For MSK delivery the EXT includes the Key Domain ID and a Key Type ID sub-payload. The Key Domain ID has the value as specified in clause 6.3.2.1. The Key Type ID sub-payload includes the type and ID of the key that is delivered in the message, i.e. the MSK ID, see Figure 6.x. The key that is used to protect the message, i.e. MUK, is identified as specified in clause 6.1.

For MTK delivery the EXT includes the Key Domain ID and two Key Type ID sub-payloads. The Key Domain ID has the value as specified in clause 6.3.2.1. The first Key Type ID sub-payload includes the type and ID of the key that is used to protect the message, i.e. the MSK ID, and the second Key Type ID sub-payload includes the type and ID of the key that is delivered in the message, i.e. the MTK ID, see Figure 6.y.

Editorís Note: The Key Domain ID needs to be added to [13]. It may need an extension payload type of its own.

~~The IDs of the involved keys are kept in the EXT, to enable the UE to look up the identity of the key which was used to protect the message, and which key is delivered in the message. This EXT is incorporated in the MIKEY messages (see Figure 6.4). When an MSK is delivered to a UE, the MIKEY message contains an EXT that holds the MUK ID of the MUK used to protect the delivery, and the MSK ID of the MSK delivered in the message. For messages that contain an MTK, the EXT contains the MSK ID of the MSK used to protect the delivery, and the MTK ID of the MTK contained in the message.~~ The MSK ID and MTK ID are increased by 1 every time the corresponding key is updated. It is possible that the same MTK is delivered several times in multicast, and the ME can then discard messages related to a key it already has instead of passing them to the MGV-F.

The MGV-F (see clause 6.5) protects itself from a possibly malicious ME by checking the integrity and freshness of the MIKEY message.

The format of the key IDs shall be represented by unsigned integer counters, different from zero. The reason for disallowing zero is that it is reserved for future use. Note that this means that there can only be 2^{n} - 1 different keys in use during the same session, where n is the number of bits in the ID field.

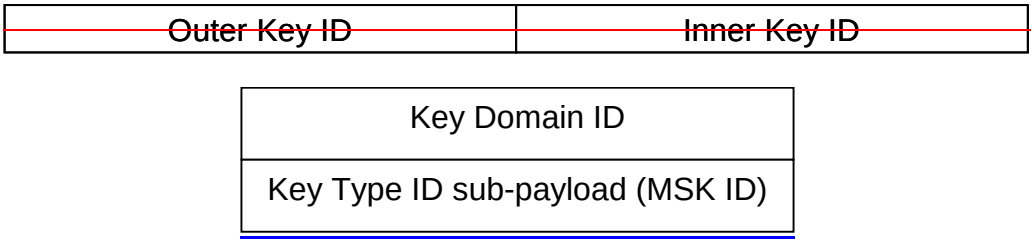

Figure 6.x4: Extension payload used with MIKEY MSK message

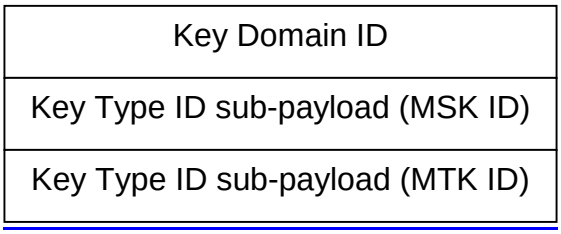

Figure 6.y: Extension payload used with MIKEY MTK message

The Inner Key ID is the ID of the key that is transported in the message (i.e. an MSK or MTK). The Outer Key ID is the ID of the key used as pre-shared secret for the key delivery (i.e. an MUK or MSK).

*******NEXT CHANGE*********

6.4.5 MIKEY message structure

6.4.5.1 MSK message structure

The structure of the MIKEY message carrying a MSK key is depicted in Figure 6.5. The actual key that is delivered is kept in the KEMAC payload. The MIKEY-RAND is used to derive e.g. encryption and authentication keys from the received keys. It is sent only in the initial MSK delivery message. The identity payloads of the initiator's and responder's IDs shall be included in the MSK transport messages. IDi is the ID of the BM-SC (i.e. NAF-ID) and IDr is the ID of the UEís username (i.e.B-TID). Security Policy (SP) payload includes information for the security protocol such as algorithms to use, key lengths, initial values for algorithms etc. The Key Validity Data subfield is present in the KEMAC payload when MSK is transported but it is not present for MTK transport. The field defines the Key Validity Time for MSK in terms of sequence number interval (i.e. lower limit of MTK ID and upper limit of MTK ID). The lower limit of the interval defines the SEQs to be used by the MGV-F (see clause 6.5).

Editor's Note: The type (URI or NAI) of identity payloads to use are for further study.

Editor's Note: The contents of the Security Policy payload depends on the used security protocols. RFC 3830 [9] (MIKEY) has defined Security Policy payload for SRTP, but for other security protocols there is a need to define new Security Policy payloads. The exact definitions of these are FFS.

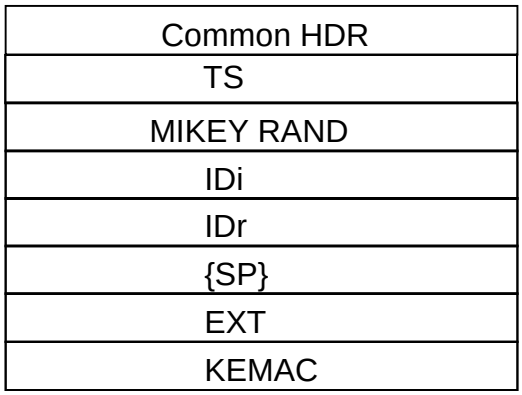

Figure 6.5: The logical structure of the MIKEY message used to deliver MSK. For use of brackets, cf. section 1.3 of RFC 3830 [9] (MIKEY)

6.4.5.2 MSK Verification message

If the BM-SC expects a response to the MSK-transport message (i.e., the V-bit in the MIKEY common header is equal to 1), the UE shall send a verification message as a response. The verification message shall be constructed according to section 3.1 of MIKEY, and shall consist of the following fields: HDR || TS ||~~ IDi_ ||~~ IDr || V, where ~~IDi is the ID of the BM-SC and~~ IDr is the ID of the UE. Note that the MAC included in the verification payload, shall be computed over both the initiator's and the responder's ID~~s~~ as well as the timestamp in addition to be computed over the response message as defined in RFC 3830 [9]. The key used in the MAC computation is the MUK_I.

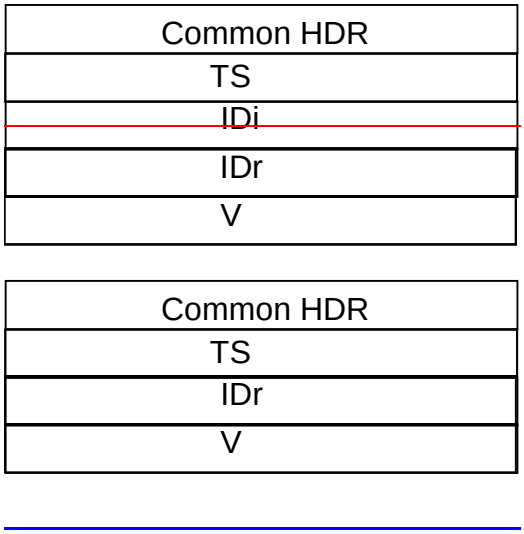

Figure 6.6: The logical structure of the MIKEY Verification message

The verification message shall not be sent as a response to MIKEY messages delivering MTK.

The verification message shall be constructed by the ME, except for the MAC field, and then be given to the MGV-F that will perform the MAC computation and will return the verification message appended with the MAC to the ME. The ME shall send the message to the BM-SC.

********** NEXT CHANGE**********

6.4.6 Processing of received messages in the ME

6.4.6.1 MSK MIKEY Message Reception

When the MIKEY message arrives at the ME, the processing proceeds following the steps below (basically following section 5.3 of RFC 3830 [9]).

1. The Extension Payload (~~Data Type field of the common MIKEY header (HDR)~~EXT) is examined, and if it indicates an MSK delivery protected with MUK, the MUK ID is received ~~extracted from the Extension Payload~~by combining IDi and IDr.
2. The Timestamp Payload is checked, and the message is discarded if the counter is larger or equal to the current MIKEY replay counter associated with the given MUK (the counter value is retrieved from MGV-S). To avoid issues with wrap around of the ID fields "smaller than" should be in the sense of RFC 1982 [10].
3. The Security Policy payload is stored if it was present.
4. The message is transported to MGV-F for further processing, cf clause 6.5.2.
5. The MGV-F replies success or failure.

6.4.6.2 MTK MIKEY Message Reception

When the MIKEY message arrives at the ME, the processing proceeds following the steps below (basically following section 5.3 of RFC 3830 [9]).

1. The Extension Payload (~~Data Type field of the common MIKEY header (HDR)~~ EXT) is examined, and if it indicates an MT~~S~~K delivery protected with MSK, the MSK ID is extracted from the Extension Payload.
2. The Timestamp Payload is checked, and the message is discarded if the counter is larger or equal to the current MIKEY replay counter associated with the given MSK (the counter value is retrieved from MGV-S). To avoid issues with wrap around of the ID fields ``smaller than~~¥~~should be in the sense of RFC 1982 [10].
3. If the MTK ID extracted from the Extension payload is less than or equal to the current MTK ID (kept in the ME), the message shall be discarded.
4. The message is transported to MGV-F for further processing, cf 6.5.3.
5. The MGV-F replies success (i.e. sending the MTK) or failure.

******** NEXT CHANGE********

6.5.2 MUK derivation

When a MUK has been installed in the MGV-S, i.e. as a result of a GBA run, it is used as pre-shared secret ~~together with the MIKEY_RAND and the Key Group ID from the MIKEY message to derive encryption and integrity keys (MUK_C and MUK_I) as defined in section 4.1.4 of MIKEY. MUK_I and MUK_C are~~ used to verify the integrity of the MSK transport message and decrypt the key carried in the KEMAC payload as described in RFC 3830 [9].

6.5.3 MSK processing~~validation and derivation~~

When the MGV-F receives the MIKEY message, it first determines the type of message by reading the ~~Data Type field in the common header~~EXT. If the key in the message is an MSK protected by MUK, MGV-F retrieves the MUK identified as specified in clause 6.1.~~with the ID given by the Extension payload~~.

~~The MAC in the KEMAC payload is verified using MUK_I, and the message is discarded if verification fails. If the MAC verification is successful the MUK_C is used to decrypt the Key Data sub-payload, and the MSK can be installed in the MGV-S. The MSK is used as pre-shared secret together with the MIKEY_RAND and the Key Group ID from the MIKEY message to derive (as specified in section 4.1.4 of RFC 3830 [9]) encryption and integrity keys (MSK_I and~~

~~MSK_C).~~ The integrity of the message is validated and the MSK is extracted from the KEMAC payload as described in Section 5 of [9] if the validation is successful. The Key Validity data is extracted from the message and stored (in the form of MTK ID interval). The lower limit of the interval defines the SEQs.

NOTE: The MSK is not necessarily updated in the message, since a MSK transport message can be sent e.g. to update the Key Validity data.

If message~~MAC verific~~validation is successful, then the MGV-F shall update in MGV-S the counter value in the Time Stamp payload associated with the corresponding MUK ID.

6.5.4 MTK processing~~validation and derivation~~

When the MGV-F receives the MIKEY message, it first determines the type of message by reading the ~~Data Type field in the common header~~EXT. If the key inside the message is an MTK protected by MSK, MGV-F retrieves the MSK with the ID given by the Extension payload.

It is assumed that the MBMS service specific data, MSK and the sequence number SEQs, have been stored within a secure storage (MGV-S). Both MSK and SEQs were transferred to the MGV-S with the execution of the MSK update procedures. The initial value of SEQs is determined by the service provider.

The MGV-F shall only calculate and deliver the MBMS Traffic Keys (MTK) to the ME if the ptm-key information is deemed to be fresh.

The MGV-F shall compare the received SEQp, i.e. MTK ID from the MIKEY message with the stored SEQs. If SEQp is equal or lower than SEQs then the MGV-F shall indicate a failure to the ME. If SEQp is greater than SEQs then the MGV-F shall verify the integrity of the MIKEY message according to RFC 3830 [9]. ~~calculate the MAC as defined in RFC 3830 [9] using the received MIKEY message and MSK as input. This MAC is compared with the MAC of the KEMAC payload in the MIKEY message~~. If the ~~MAC~~ verification is unsuccessful, then the MGV-F will indicate a failure to the ME. If the ~~MAC~~ verification is successful, then the MGV-F shall update SEQs with SEQp value and extract the ~~start the generation of~~ MTK from the message. The MGV-F then provides the MTK to the ME.

The MGV-F shall update in MGV-S the counter value in the Time Stamp payload associated with the corresponding MSK ID.

NOTE: MIKEY includes functionality to derive further keys from MTK if needed by the security protocol. The key derivation is defined in section 4.1.3 of RFC 3830 [9] (MIKEY).].

******** NEXT CHANGE********

6.6.2.1 Usage of SRTP

When it is required to protect MBMS streaming data SRTP (Secure Real-time Transport Protocol) as defined in RFC 3711 [11] shall be used. The MTK is carried to the UEs from the BM-SC using RFC 3830 [9] (MIKEY) with extensions defined according to this specification. MTK shall be used as the master key in SRTP key derivation to derive the SRTP session keys as defined in section 4.3 of RFC 3830 [9]. The correct MTK to use to decrypt the data is indicated using the MKI (Master Key identifier) field, which is included in the SRTP packets as defined in RFC 3711 [11]. The form of MKI shall be a concatenation of Network ID, ~~Key Group ID,~~ MSK ID and MTK ID, i.e. MKI = (Network ID || ~~Key Group ID ||~~ MSK ID || MTK ID).

If the SRTP packets are to be integrity protected, the SRTP authentication tag is appended to the packets as defined in RFC 3830 [9].

SRTP security policy parameters, such as encryption algorithm, are transported in MIKEY Security Policy payload as defined in section 6.10.1 in RFC 3830 [9].

******** NEXT CHANGE********

Annex D (normative): UICC-ME interface

D.1 MSK Update Procedure

This procedure is part of the MSK update procedure as described in clause 6.5 (Validation and key derivation functions in MGV-F).

The ME has previously performed a GBA_U bootstrapping procedure as described in TS 33.220. The UICC stores the corresponding Ks_int_NAF together with the NAF_Id associated with this particular bootstrapping procedure.

The ME receives a MIKEY message containing an MSK update procedure. After performing some validity checks, the ME sends the whole message to the UICC. The ~~ME~~ UICC ~~also includes in this request NAF_Id~~uses the MUK ID (included in the MIKEY message, see clause 6.1) to identify the stored Ks_int_NAF.

The UICC then uses Ks_int_NAF as the MUK value for MUK derivation and MSK validation and derivation (as described in clause 6.5.3).

After successful MSK Update procedure the UICC stores the ~~Network~~ Key Domain ID~~, Key Group ID~~, MSK ID, MSK and MSK Validity Time (in the form of MTK ID interval).

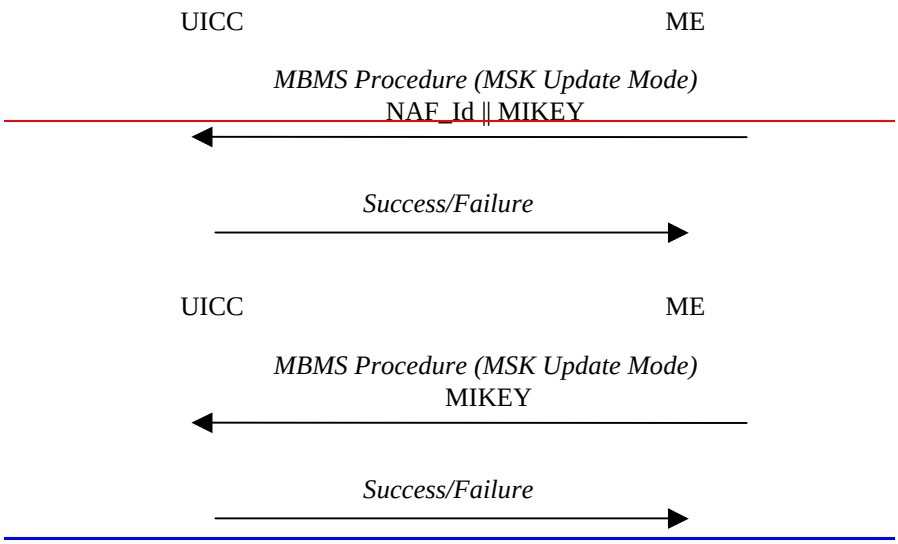

Figure D.1: MSK Update Procedure

D.2 MSK Verification Message Generation

This procedure is part of the MSK Verification Message as described in clause 6.4.5.2 (MSK Verification message).

The ME constructs the verification message in response to the MSK-transport message when it is required by BM-SC.

The ME shall then give the constructed MIKEY verification message, with an empty MAC field, to the UICC and the ME shall include NAF_id in this message. The ~~ME~~ UICC ~~also includes in this request NAF_Id~~ uses the MUK ID (see clause 6.1) to identify the stored Ks_int_NAF=MUK to be used in the MSK Verification Message Generation.

The UICC will verify that the Time Stamp MIKEY field correspond to the previous MSK Update procedure. Then, the UICC shall compute and send the MIKEY packet to the ME (including the calculated MAC field) as defined in clause 6.4.5.2. (MSK Verification message).

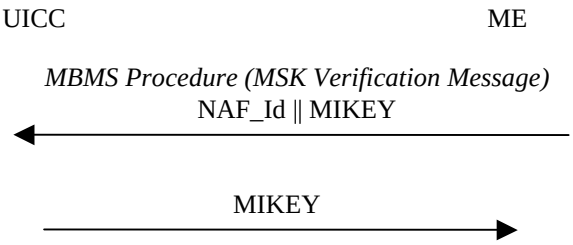

Figure D.2: MSK Verification Message

******** NEXT CHANGE********

D.3 MTK generation and validation

This procedure is part of the MTK generation and validation function as described in clause 6.5.4 (MTK validation and derivation).

The ME receives the MIKEY message (containing Header, Time stamp, ~~Network~~ Key Domain ID~~, Key Group ID~~, MSK ID, MTK ID = SEQp, MSK_C[MTK] and MAC). After performing some validity checks, the ME sends the whole message to the UICC. The UICC computes the MGV-F function as described in clause 6.5. (Validation and key derivation functions in MGV-F). After successful MGV-F procedure the UICC returns the MTK.

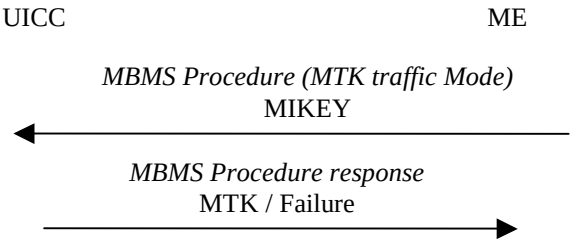

Figure D.3: MTK Generation and Validation