3GPP TSG SA WG3 Security — S3#31 **S3-030723**
18 - 21 November 2003
Munich, Germany

Agenda Item:
Source: Ericsson

Title: Migration of MIKEY in MBMS key management

Document for: Discussion/Decision

1. Introduction

In SA3 Ad hoc meeting in Antwerp a concern was raised against SRTP (Secure RTP) [1] that it may suffer from session key pre-calculation problem in certain circumstances where session key derivation is protected inside a tamper resistant module. Ericsson has studied the issue and come to the conclusion that the problem is mitigated by providing a solution in key management level, i.e. in MIKEY (Multimedia Internet KEYing) [2] protocol.

Two different solutions have been proposed for MBMS key management, ME based and UICC based. A possible MBMS key management protocol should enable smooth migration from ME based solution to UICC based solution in case migration is required.

This contribution describes two solutions to solve the identified pre-calculation problem in MIKEY protocol and how these solutions can be used to implement both ME and UICC based MBMS key management with MIKEY and thus enable migration.

2. MIKEY key hierarchy and pre-calculation attack

MIKEY key hierarchy is depicted in figure 1. It includes two levels of keys and a random value. The TEK (traffic encryption key) is derived from the TGK (TEK generation key) and the random value TEK-RAND. The TEK is delivered to the security protocol, which may derive further keys. The TGK is sent to the UE in PTP (point-to-point) manner in initial keying and also during re-keying. The TEK-RAND is sent to the UE only in the initial key exchange. It is not updated during re-keying. The update frequency of the TGK value is operator configurable.

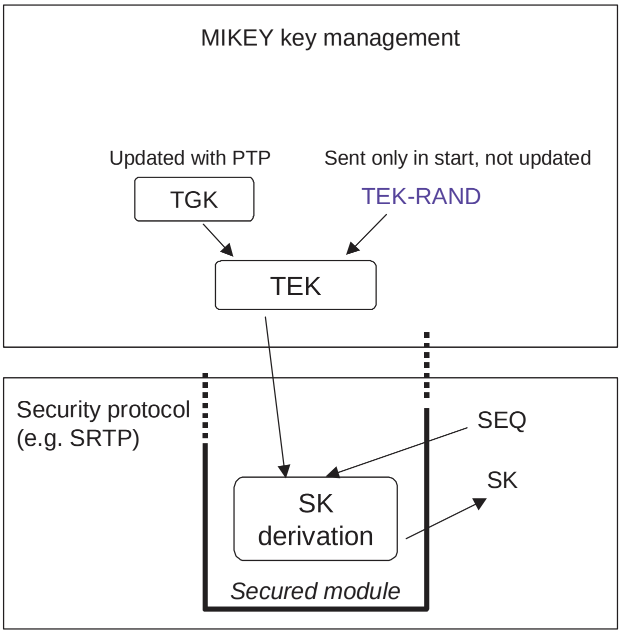

Figure 1 MIKEY key hierarchy and SRTP pre-calculation attack

The figure shows as an example how SRTP [1] derives its own session key SK from the TEK and SRTP sequence number SEQ, that is received with the user traffic in SRTP packet header.

The SRTP pre-calculation attack is shown also. In case the SK derivation function in SRTP is insulated in secured module, it may be possible for an attacker to get future session keys from the secured module without knowing the TEK. This is possible by sending the SEQ to the module and the module will respond with the corresponding session key.

Ericsson has studied the issue and come to the conclusion that the problem is mitigated by providing a solution in key management level, i.e. in MIKEY. Two alternative solutions are described in sections 3 and 4.

3. Alternative 1. TEK based solution

In alternative 1 the pre-calculation is mitigated by adding a new random value TEK-RAND2 as input to TEK generation, see figure 2. The TEK-RAND2 is sent in point-to-multipoint manner with the user traffic to the UE and it may change quite frequently. Always when the TEK-RAND2 changes, a new TEK is derived using both TEK-RAND and TEK-RAND2. TGK and TEK-RAND are the same as in chapter 2.

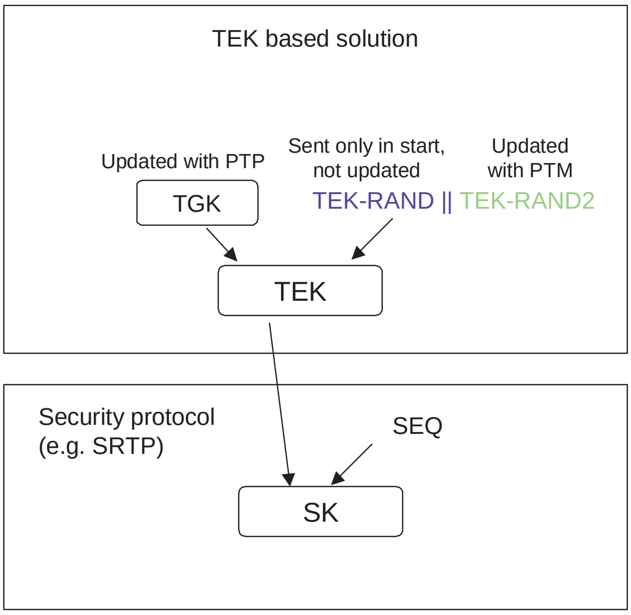

Figure 2 TEK based solution

Usage of TEK based solution in MBMS key management from the UE perspective is depicted in figure 3. UICC based scenario is shown on the left side and ME based scenario is shown on the right side. The complete view is described in accompanying pseudo CRs [3],[4].

The main difference in the implementations is the storage of TGK and derivation of TEK. In UICC based implementation these have been implemented on UICC.

This means that when MIKEY receives a PTP initial keying or re-keying message, it forwards the message to the UICC since it cannot decrypt the content. Also, when MIKEY receives a changed MKI value (TGK-ID || TEK-RAND2) from the security protocol, MIKEY requests UICC to calculate a new TEK. Note that the TGK-ID || TEK-RAND2 are carried in the security protocol with the user traffic in PTM (point-to-multipoint) manner. For example MKI field in SRTP can be used for this purpose.

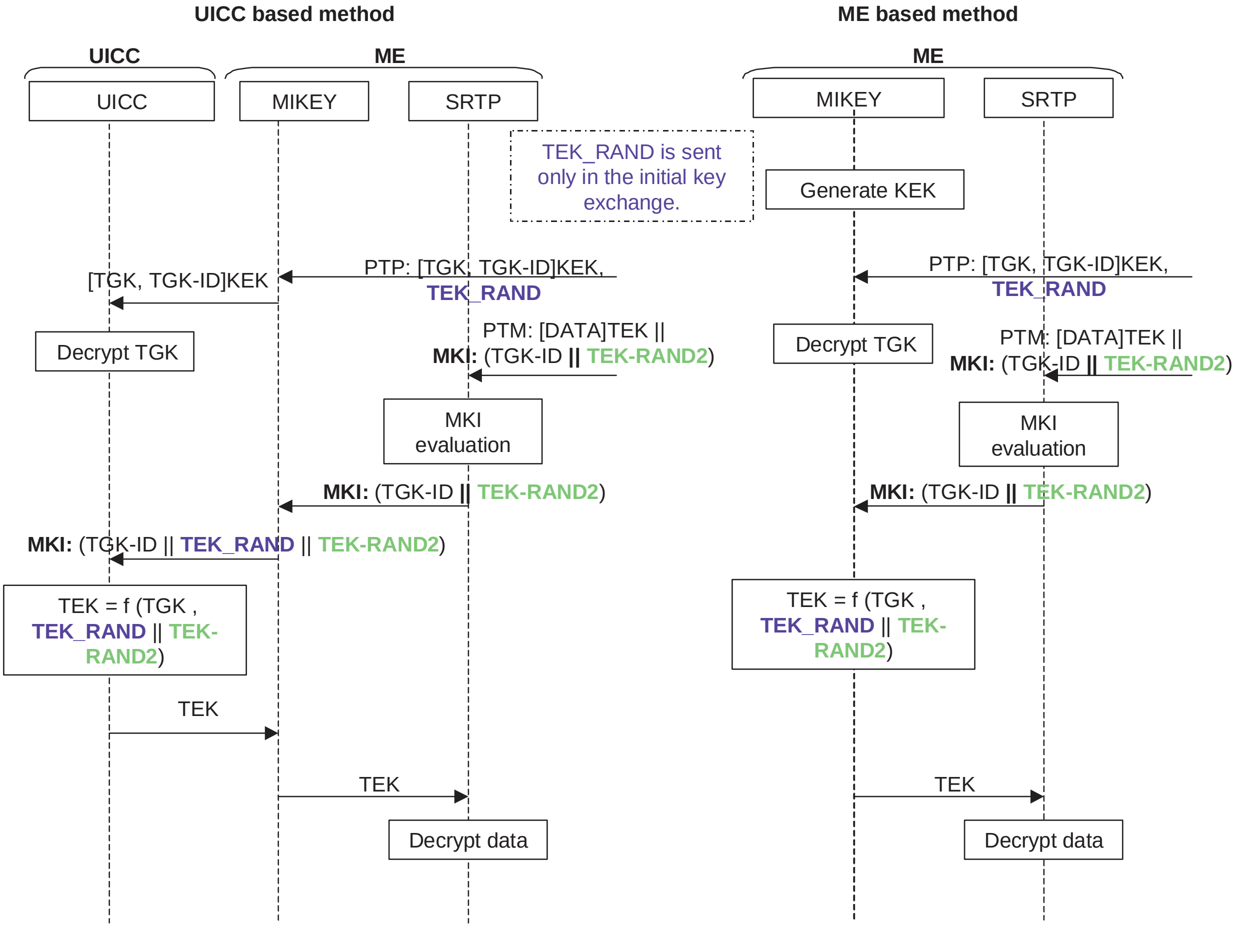
UICC based method
ME based method
UICC
ME
UICC
MIKEY
SRTP
ME
MIKEY
SRTP
TEK_RAND is sent only in the initial key exchange.
Generate KEK
PTP: [TGK, TGK-ID]KEK, TEK_RAND
[TGK, TGK-ID]KEK
PTP: [TGK, TGK-ID]KEK, TEK_RAND
Decrypt TGK
PTM: [DATA]TEK || MKI: (TGK-ID || TEK-RAND2)
Decrypt TGK
PTM: [DATA]TEK || MKI: (TGK-ID || TEK-RAND2)
MKI evaluation
MKI evaluation
MKI: (TGK-ID || TEK-RAND2)
MKI: (TGK-ID || TEK-RAND2)
MKI: (TGK-ID || TEK_RAND || TEK-RAND2)
TEK = f (TGK , TEK_RAND || TEK-RAND2)
TEK = f (TGK , TEK_RAND || TEK-RAND2)
TEK
TEK
TEK
Decrypt data
Decrypt data

Figure 3 Migration with TEK based solution

4. Alternative 2. TGK based solution

In alternative 2 the pre-calculation is mitigated by adding a new key derivation layer between TGK and TEK, see figure 4. The new key SK-TGK is derived using TGK and a new random value TGK-RAND. The TGK-RAND is sent in point-to-multipoint manner with the user traffic to the UE and it may change quite frequently. Always when the TGK-RAND changes, a new SK-TGK (short term TGK) is derived. This triggers the TEK derivation with TEK-RAND. TGK and TEK-RAND are the same as in chapter 2.

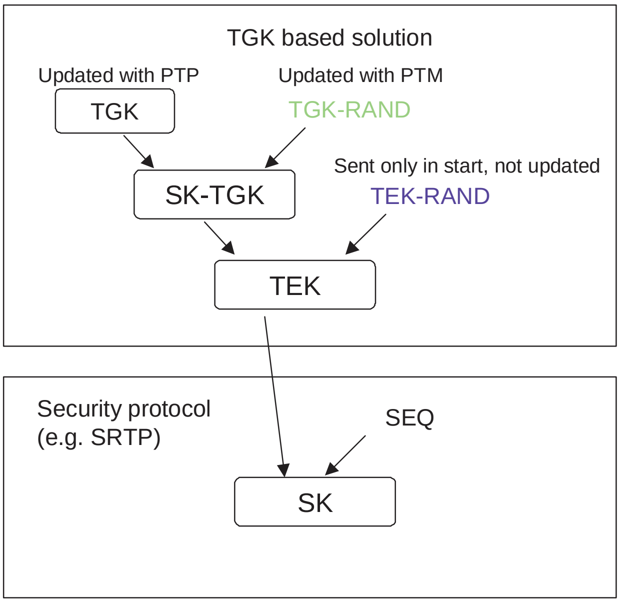

Figure 4 TGK based solution

Usage of TGK based solution in MBMS key management from the UE perspective is depicted in figure 5. UICC based scenario is shown on the left side and ME based scenario is shown on the right side.

The main difference in the implementations is the storage of TGK and derivation of SK-TGK. In UICC based implementation these have been implemented on UICC.

This means that when MIKEY receives a PTP initial keying or re-keying message, it forwards the message to the UICC since it cannot decrypt the content. Also, when MIKEY receives a changed MKI value (TGK-ID || TGK-RAND) from the security protocol, MIKEY requests UICC to calculate a new SK-TGK. Note that the TGK-ID || TGK-RAND are carried in the security protocol with the user traffic in PTM (point-to-multipoint) manner. For example MKI field in SRTP can be used for this purpose.

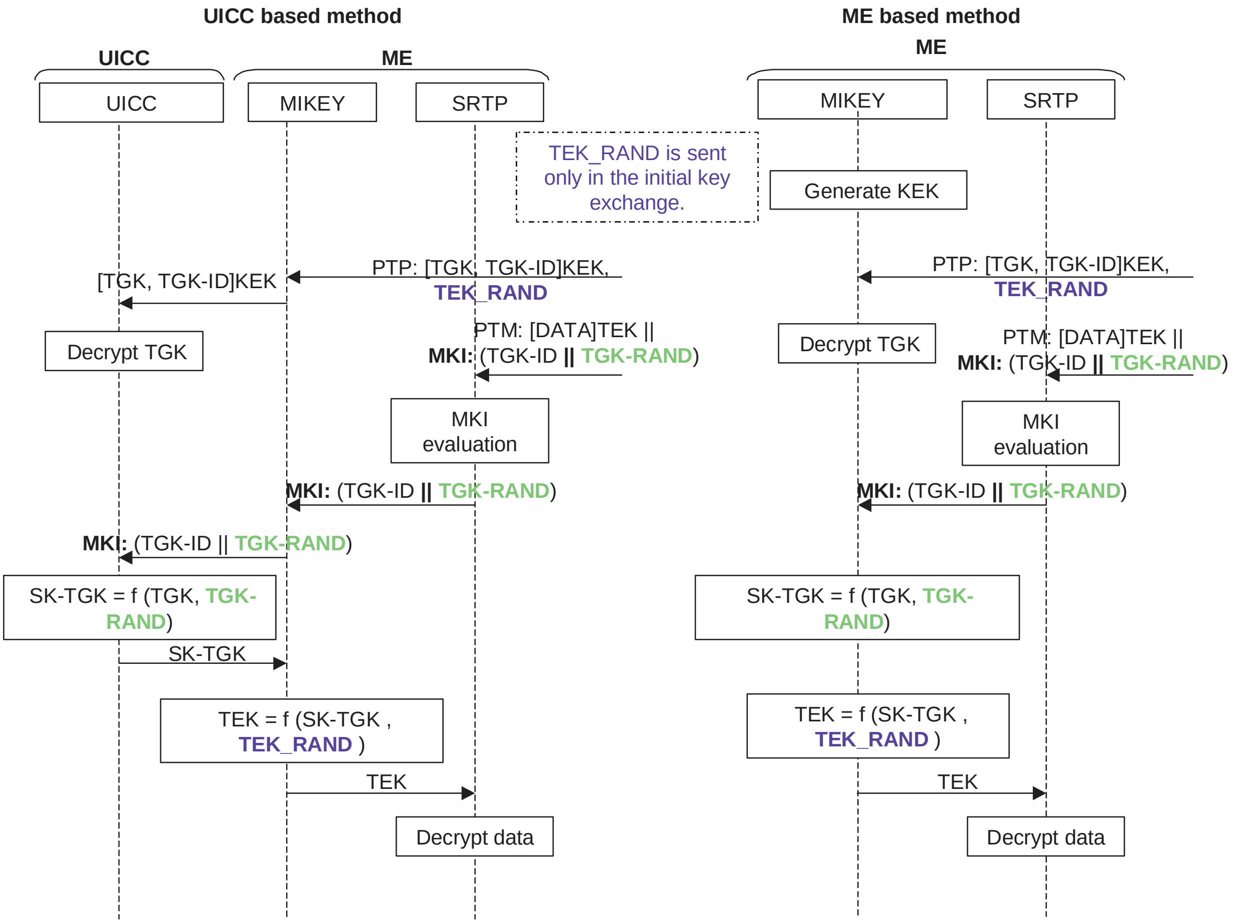
UICC based method
ME based method
UICC
ME
ME
UICC
MIKEY
SRTP
MIKEY
SRTP
TEK_RAND is sent only in the initial key exchange.
Generate KEK
PTP: [TGK, TGK-ID]KEK, TEK_RAND
[TGK, TGK-ID]KEK
PTP: [TGK, TGK-ID]KEK, TEK_RAND
Decrypt TGK
PTM: [DATA]TEK || MKI: (TGK-ID || TGK-RAND)
Decrypt TGK
PTM: [DATA]TEK || MKI: (TGK-ID || TGK-RAND)
MKI evaluation
MKI evaluation
MKI: (TGK-ID || TGK-RAND)
MKI: (TGK-ID || TGK-RAND)
MKI: (TGK-ID || TGK-RAND)
SK-TGK = f (TGK, TGK-RAND)
SK-TGK = f (TGK, TGK-RAND)
SK-TGK
TEK = f (SK-TGK , TEK_RAND)
TEK = f (SK-TGK , TEK_RAND)
TEK
TEK
Decrypt data
Decrypt data

Figure 5 Migration with TGK based solution

5. Recommendation

Alternative 1 and 2 provide the same level of security, but alternative 1 is simpler, since it introduces only a new random input to existing key derivation, whereas alternative 2 introduces a new key derivation function and a new random value. Therefore alternative 1 is the preferred solution for MBMS key management protocol.

6. IETF considerations

MIKEY internet draft has come from IESG review with several good comments to further enhance MIKEY. Ericsson therefore thinks that there is a small window of opportunity to further enhance MIKEY to consider the SA3 requirements for MBMS. Therefore Ericsson will initiate this discussion with IETF and report back what the progress is.

7. Migration

The chosen key management solution should provide smooth and backward compatible migration from ME based solution to UICC based solution in order to enable introduction of new types of services in the future and continuity of MBMS services between releases.

MIKEY provides smooth migration path to UICC based method, since it can be used with both ME and UICC based methods. The backward compatibility could be handled when the user tries to register for the service before the MBMS encryption keys are provided to the user.

8. Proposal

The ME and UICC based key management solutions are based on different trust models. The ME based solution assumes that the ME is trusted given the value of the services. UICC based solution assumes that the ME is not trusted and that the keys can be leaked. Note that not only has the key to be leaked but the users need also to put the key into the ME for operation. What is the probability for this scenario to have a compelling impact on operators sales for the mainstream market?
It is proposed that before making a decision on key management solution SA3 should take a standpoint on the trust model that is applied in MBMS. That is, whether ME is trusted or not.

Given that the ME is trusted it is proposed that ME based key management is adopted in Release 6 and that migration to UICC based key management is adopted in future releases (or already in release 6) if SA3 so decides.

It is proposed to adopt MIKEY as key management protocol for MBMS. MIKEY can support both ME and UICC based methods. MIKEY also provides smooth migration path to UICC based method.

9. Conclusion

MIKEY enables both ME and UICC based key management. This document and accompanying pseudo-CRs [3], [4] describe how MIKEY is used in ME and UICC based key management.

Ericsson is in the process to further enhance MIKEY based on received feedback from IESG review. This gives us we believe a small window of opportunity to further enhance MIKEY to consider also MBMS requirements. However Ericsson is not in the position right now to signal how easy this is from an IETF point of view but Ericsson will report the progress and the general enhancement work on MIKEY.

10. References

[1] The Secure Real-time Transport Protocol, draft-ietf-avt-srtp-09.txt, work in progress

[2] MIKEY: Multimedia Internet KEYing, draft-ietf-msec-mikey-07.txt, work in progress

[3] TD S3-030xxx: ME based MBMS key management with MIKEY, Ericsson, SA3#31, Munich

[4] TD S3-030xxx: UICC based MBMS key management with MIKEY, Ericsson, SA3#31, Munich

3GPP TSG-SA3 Meeting #31
Munich, Germany, 18-21 November 2003

Tdoc ⌘S3-03xxxx

CR-Form-v7

CHANGE REQUEST

⌘	**TS 33.246**	**CR**	**CRNum**	⌘**rev**	**-**	⌘	Current version:	0.2.1	⌘

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⌘ □ ME ☒ Radio Access Network □ Core Network ☒

Title:	⌘	ME based MBMS Key management with MIKEY		
Source:	⌘	Ericsson		
Work item code:	⌘	MBMS	***Date:*** ⌘	11/11/2003
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:	⌘	Key management has not been specified
Summary of change:	⌘	ME based MBMS key management is performed with MIKEY protocol
Consequences if not approved:	⌘	

Clauses affected:	⌘	6				
		Y	**N**			
Other specs Affected:	⌘		**X**	Other core specifications	⌘	
			X	Test specifications		
			X	O&M Specifications		
Other comments:	⌘					

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

6 Security mechanisms

6.1 Authentication and authorisation of a user

Editor's note: this section will contain the details of how a user joins a particular Multicast Service

6.2 Key management

Editor's note: this section will contain the details of how the keys are updated in a Multicast Service

Editor's note: MIKEY is being considered as the method for carrying keys. Possible optimisations were proposed at the ad-hoc in Antwerp (S3z030010). One identified issue was the possible need to terminate MIKEY in the UICC and/or terminal in the combined method. The use of MIKEY relates to the PTP delivery of a key

6.2.1 Overview

The overview of MBMS security functional architecture is depicted in figure.

- The SM is general session manager in BM-SC that uses key management services of MIKEY protocol.

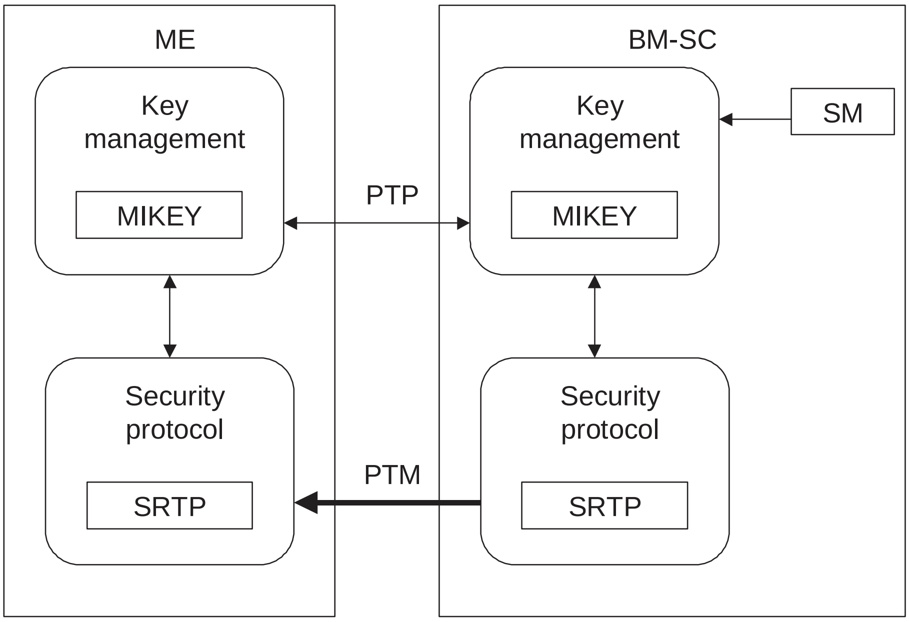

Figure 2: MBMS security functional architecture

6.2.2 TGK generation

The SM generates a value for Tek Generation Key (TGK) and a value for TEK_RAND. These actions are independent of the actions of individual UEs.

6.2.3 PTP initial keying procedure

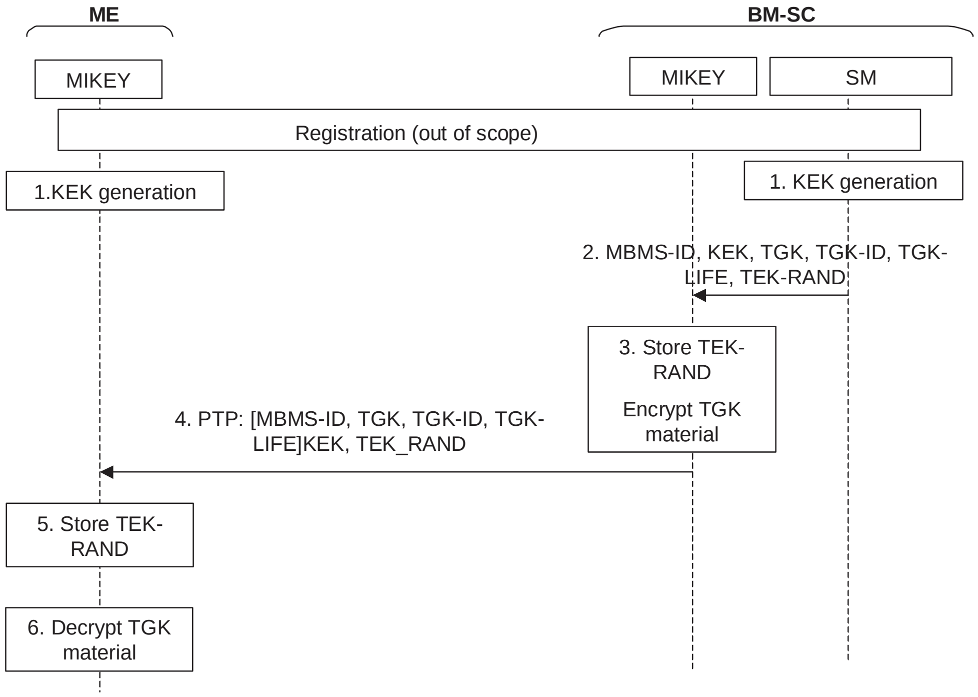

Figure 3: Initial keying procedure

Before the initial key exchange is performed, the user has been registered to the service and she has been authenticated and authorised.

> Editor's note: The authentication of the user is FFS and it could be done e.g. with GBA. Therefore it is out of the scope of current key management description.

1. SM in BM-SC and the ME are provisioned with Key encryption key (KEK). The KEK is used to protect the MIKEY message delivery. KEK may be derived e.g. from the GBA keys.

2. SM sends MIKEY the following parameters:

   - MBMS service Identifier MBMS-ID
   - KEK
   - TGK
   - TGK identifier TGK-ID
   - TGK lifetime TGK_LIFE
   - TEK_RAND

3. MIKEY in BM-SC stores TEK-RAND and encrypts TKG, TGK-ID, TGK_LIFE and MBMS-ID with KEK

4. MIKEY in BM-SC sends the encrypted key material with TEK_RAND to UE in PTP manner.

5. MIKEY in the ME stores the TEK_RAND

6. MIKEY in the ME decrypts the encrypted key material with KEK

6.2.3 PTM re-keying and data transmission procedure

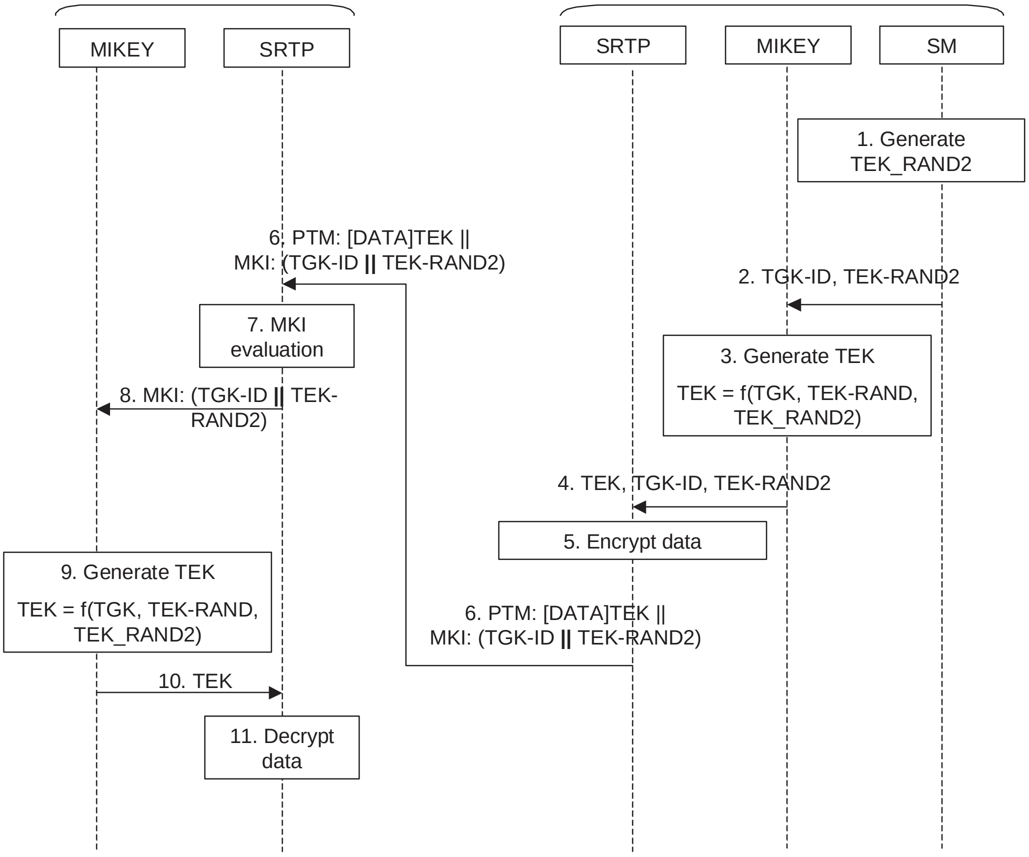
MIKEY
SRTP
SRTP
MIKEY
SM
1. Generate TEK_RAND2
2. TGK-ID, TEK-RAND2
3. Generate TEK
TEK = f(TGK, TEK-RAND, TEK_RAND2)
4. TEK, TGK-ID, TEK-RAND2
5. Encrypt data
6. PTM: [DATA]TEK || MKI: (TGK-ID || TEK-RAND2)
6. PTM: [DATA]TEK || MKI: (TGK-ID || TEK-RAND2)
7. MKI evaluation
8. MKI: (TGK-ID || TEK-RAND2)
9. Generate TEK
TEK = f(TGK, TEK-RAND, TEK_RAND2)
10. TEK
11. Decrypt data

Figure 4: PTM re-keying and data transmission procedure

The initial keying is assumed to have happened.

1. The SM generates TEK_RAND2. Note that this can be generated frequently based on operator policy, e.g. every ten minutes.

2. SM passes the TEK_RAND2, TGK-ID and TGK_LIFE to MIKEY in BM-SC.

3. MIKEY in BM-SC derives TEK using TGK, TEK_RAND and TEK_RAND2. Note that the TEK-RAND is stored from the initial key exchange.

4. MIKEY in BM-SC sends TEK with TGK_ID and TEK_RAND2 to the security protocol (i.e. SRTP). Note that TGK_ID and TEK_RAND2 constitute the MKI field that is carried in SRTP header.

5. SRTP encrypts MBMS data with TEK.

6. SRTP sends the encrypted MBMS data to the UE. SRTP also includes the MKI field, i.e. the TGK_ID and TEK_RAND2.

7. SRTP in ME receives the encrypted content and takes the following actions:

   - If the MKI field is unchanged from the previous received content, processing continues in step 11.

- If the MKI field is changed, processing continues in step 8.

8. SRTP passes the MKI (= TGK_ID and TEK_RAND2) to MIKEY IN ME and requests for a new TEK.

9. MIKEY in ME calculates a new TEK using TGK, TEK_RAND and TEK_RAND2.

10. MIKEY in ME passes the new TEK to SRTP.

11. SRTP decrypts the content using the TEK assigned for the content and passes the unencrypted content to the user application.

6.2.4 PTP re-keying procedure

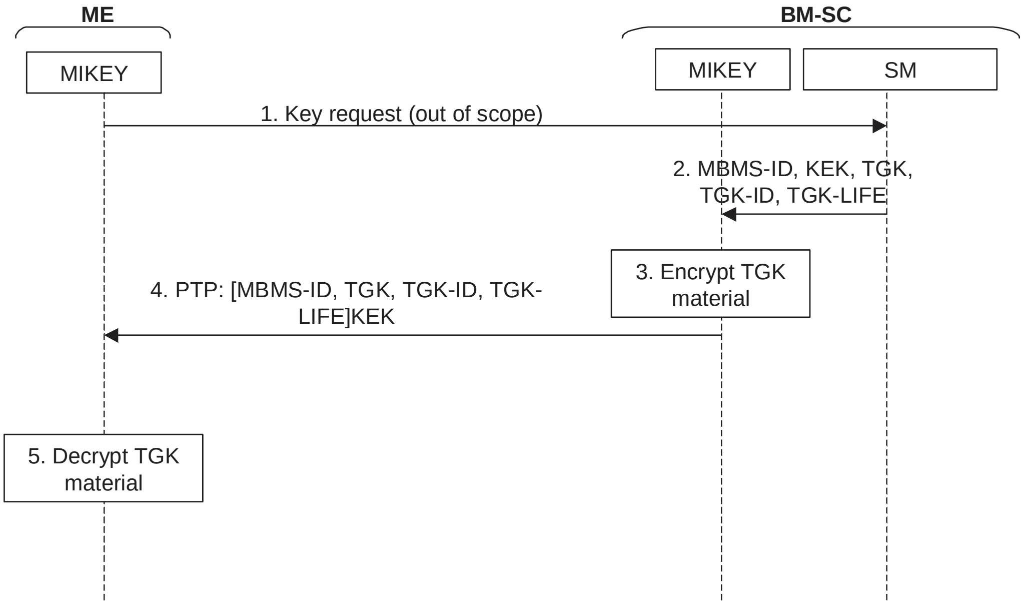
ME
BM-SC
MIKEY
MIKEY
SM
1. Key request (out of scope)
2. MBMS-ID, KEK, TGK, TGK-ID, TGK-LIFE
3. Encrypt TGK material
4. PTP: [MBMS-ID, TGK, TGK-ID, TGK-LIFE]KEK
5. Decrypt TGK material

Figure 5: PTP re-keying procedure

1. ME detects the need for a new TGK for example from TGK_LIFE or from MKI field when a TGK corresponding a TGK_ID is not available. ME sends a re-key request to SM in BM-SC.
   Note that the exact message is out of the scope of key management and it may be e.g. an HTTP request. The parameters needed are also FFS.

2. The SM in BM-SC receives the request and sends to MIKEY in BM-SC the following parameters (It is assumed that in this point the request message has been authenticated and authorised):

- MBMS service Identifier MBMS-ID
- KEK
- TGK
- TGK identifier TGK-ID
- TGK lifetime TGK_LIFE

3. MIKEY in BM-SC encrypts TKG, TGK-ID, TGK_LIFE and MBMS-ID with KEK

4. MIKEY in BM-SC sends the encrypted key material to UE in PTP manner.

5. MIKEY in the ME decrypts the encrypted key material with KEK.

6.2.5 TGK lifetime

The TGK lifetime is defined by operator policy. Frequent TGK updates provide more security and allow more flexible subscription management, but can also cause more signalling overhead since UEs retrieve new TGK values in point-to-point manner.

6.2.6 Updating the TGK before use

A new TGK should be provided to ME before the new TGK value is used to derive SK-TGK values. The UE retrieves the new TGK from the SM. If many UEs try to retrieve the key simultaneously, there will be a burst of requests. This is so called implosion problem, which may be mitigated for example in the following ways:

- Many TGKs may be sent in one MIKEY message
- The UEs may be scheduled to retrieve the TGK at different times, e.g.:
    - UEs may determine a random point in time before the TGK lifetime expires
    - BM-SC may determine the retrieve time or time interval
    - A combination of the above

6.3 Protection of the transmitted traffic

Editor’s note: this section will contain the details of how traffic is protected

Editor’s note: this section may contain several protection methods.

Editor’s note: if SRTP is chosen, the master key identifier can be used to indicate the current MBMS key whichever key management method is chosen

3GPP TSG-SA3 Meeting #31
Munich, Germany, 18-21 November 2003

Tdoc ⌘S3-03xxxx

CR-Form-v7

CHANGE REQUEST

⌘	**TS 33.246**	**CR**	**CRNum**	⌘**rev**	**-**	⌘	Current version:	0.2.1 ⌘

*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⌘ □ ME ☒ Radio Access Network □ Core Network ☒

Title:	⌘	UICC based MBMS Key management with MIKEY		
Source:	⌘	Ericsson		
Work item code:	⌘	MBMS	***Date:*** ⌘	11/11/2003
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:	⌘	Key management has not been specified
Summary of change:	⌘	UICC based MBMS key management is performed with MIKEY protocol
Consequences if not approved:	⌘	

Clauses affected:	⌘			6	
		Y	**N**		
Other specs Affected:	⌘		X	Other core specifications	⌘
			X	Test specifications	
			X	O&M Specifications	
Other comments:	⌘				

How to create CRs using this form:

Comprehensive information and tips about how to create CRs can be found at http://www.3gpp.org/specs/CR.htm. Below is a brief summary:

1) Fill out the above form. The symbols above marked ⌘ contain pop-up help information about the field that they are closest to.

2) Obtain the latest version for the release of the specification to which the change is proposed. Use the MS Word "revision marks" feature (also known as "track changes") when making the changes. All 3GPP specifications can be downloaded from the 3GPP server under ftp://ftp.3gpp.org/specs/ For the latest version, look for the directory name with the latest date e.g. 2001-03 contains the specifications resulting from the March 2001 TSG meetings.

3) With "track changes" disabled, paste the entire CR form (use CTRL-A to select it) into the specification just in front of the clause containing the first piece of changed text. Delete those parts of the specification which are not relevant to the change request.

6 Security mechanisms

6.1 Authentication and authorisation of a user

Editor's note: this section will contain the details of how a user joins a particular Multicast Service

6.2 Key management

Editor's note: this section will contain the details of how the keys are updated in a Multicast Service

Editor's note: MIKEY is being considered as the method for carrying keys. Possible optimisations were proposed at the ad-hoc in Antwerp (S3z030010). One identified issue was the possible need to terminate MIKEY in the UICC and/or terminal in the combined method. The use of MIKEY relates to the PTP delivery of a key

6.2.1 Overview

The overview of MBMS security functional architecture is depicted in figure

- The SM is general session manager in BM-SC that uses key management services of MIKEY protocol.

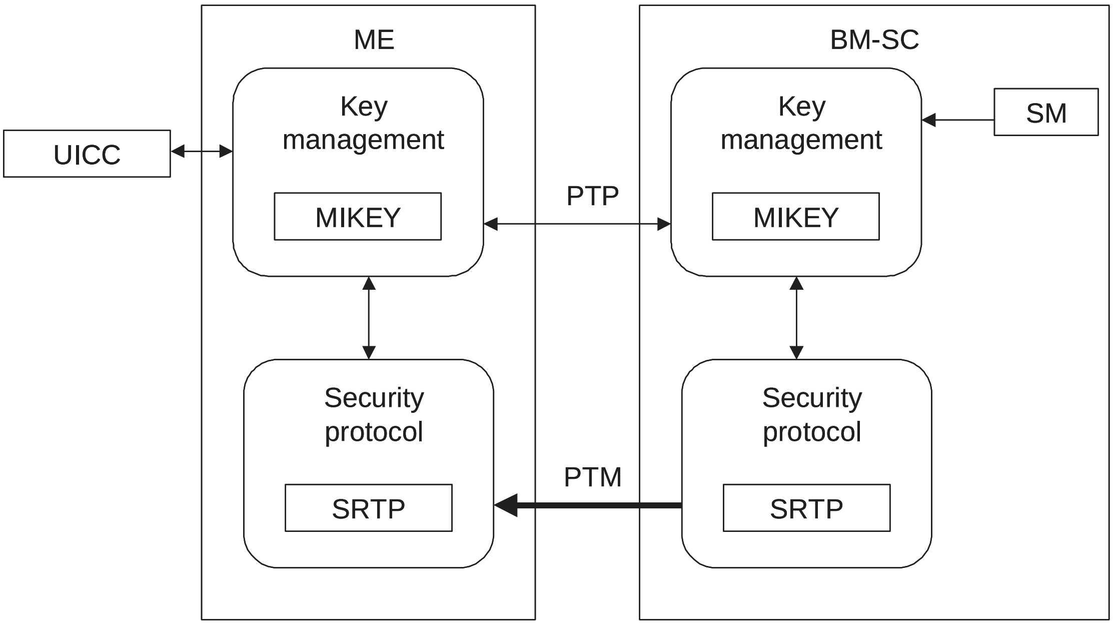

Figure 2: MBMS security functional architecture

6.2.3 Provisioning of KEK

SM in BM-SC and the UICC are provisioned with Key encryption key (KEK). The KEK is used to protect the MIKEY message delivery. Note that the KEK may provisioned on the UICC in the card manufacture phase or operator may provision it with OTA mechanisms.

6.2.2 TGK generation

The SM generates a value for Tek Generation Key (TGK) and a value for TEK_RAND. These actions are independent of the actions of individual UEs.

6.2.3 PTP initial keying procedure

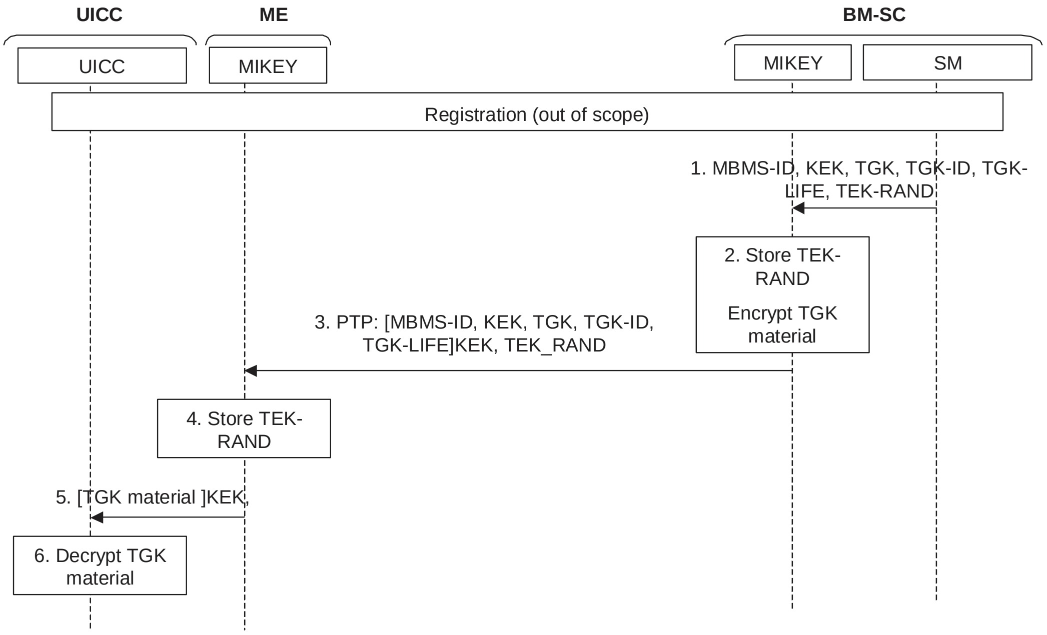
UICC
ME
BM-SC
UICC
MIKEY
MIKEY
SM
Registration (out of scope)
1. MBMS-ID, KEK, TGK, TGK-ID, TGK-LIFE, TEK-RAND
2. Store TEK-RAND
Encrypt TGK material
3. PTP: [MBMS-ID, KEK, TGK, TGK-ID, TGK-LIFE]KEK, TEK_RAND
4. Store TEK-RAND
5. [TGK material]KEK,
6. Decrypt TGK material

Figure 3: Initial keying procedure

Before the initial key exchange is performed, the user has been registered to the service and she has been authenticated and authorised.

The authentication of the user is FFS and it could be done e.g. with GBA. Therefore it is out of the scope of current key management description.

1. SM sends MIKEY the following parameters:

- MBMS service Identifier MBMS-ID
- KEK
- TGK
- TGK identifier TGK-ID
- TGK lifetime TGK_LIFE
- TEK_RAND

2. MIKEY stores TEK-RAND and encrypts TKG, TGK-ID, TGK_LIFE and MBMS-ID with KEK

3. MIKEY sends the encrypted key material with TEK_RAND to UE in PTP manner.

4. MIKEY in the ME stores the TEK_RAND

5. MIKEY in the ME passes the encrypted key material (TKG, TGK-ID, TGK_LIFE and MBMS-ID) to the UICC Note that it is FFS what key related material e.g. TGK_ID or TGK_LIFE the ME needs from the UICC in order to control e.g. the triggering of re-keying and how ME receives this material.

6. UICC decrypts the encrypted key material with KEK

6.2.3 PTM re-keying and data transmission procedure

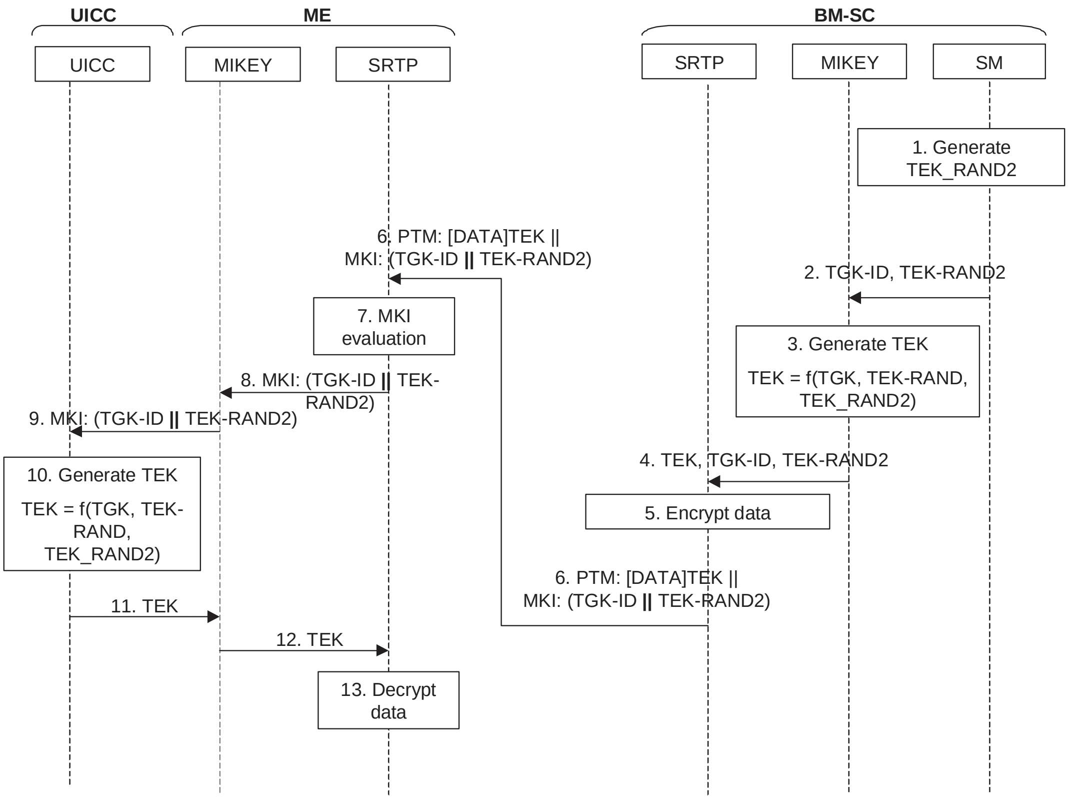
UICC
ME
BM-SC
UICC
MIKEY
SRTP
SRTP
MIKEY
SM
1. Generate TEK_RAND2
2. TGK-ID, TEK-RAND2
3. Generate TEK
TEK = f(TGK, TEK-RAND, TEK_RAND2)
4. TEK, TGK-ID, TEK-RAND2
5. Encrypt data
6. PTM: [DATA]TEK || MKI: (TGK-ID || TEK-RAND2)
6. PTM: [DATA]TEK || MKI: (TGK-ID || TEK-RAND2)
7. MKI evaluation
8. MKI: (TGK-ID || TEK-RAND2)
9. MKI: (TGK-ID || TEK-RAND2)
10. Generate TEK
TEK = f(TGK, TEK-RAND, TEK_RAND2)
11. TEK
12. TEK
13. Decrypt data

Figure 4: PTM re-keying and data transmission procedure

The initial keying is assumed to have happened.

1. The SM generates TEK_RAND2. Note that this can be generated frequently based on operator policy, e.g. every ten minutes.

2. SM passes the TEK_RAND2, TGK-ID and TGK_LIFE to MIKEY in BM-SC.

3. MIKEY in BM-SC derives TEK using TGK, TEK_RAND and TEK_RAND2. Note that the TEK-RAND is stored from the initial key exchange.

4. MIKEY in BM-SC sends TEK with TGK_ID and TEK_RAND2 to the security protocol (i.e. SRTP). Note that TGK_ID and TEK_RAND2 constitute the MKI field that is carried in SRTP header.

5. SRTP encrypts MBMS data with TEK.

6. SRTP sends the encrypted MBMS data to the UE. SRTP also includes the MKI field, i.e. the TGK_ID and TEK_RAND2.

7. SRTP in ME receives the encrypted content and takes the following actions:

- If the MKI field is unchanged from the previous received content, processing continues in step 13.
- If the MKI field is changed, processing continues in step 8.

8. SRTP passes the MKI (= TGK_ID and TEK_RAND2) to MIKEY IN ME and requests for a new TEK.
9. MIKEY passes the MKI (= TGK_ID and TEK_RAND, TEK_RAND2) to UICC to request a new TEK.
10. UICC calculates a new TEK using TGK and TEK_RAND and TEK_RAND2.
11. UICC passes the new TEK to MIKEY
12. MIKEY passes the new TEK to SRTP.
13. SRTP decrypts the content using the TEK assigned for the content and passes the unencrypted content to the user application.

6.2.4 PTP re-keying procedure

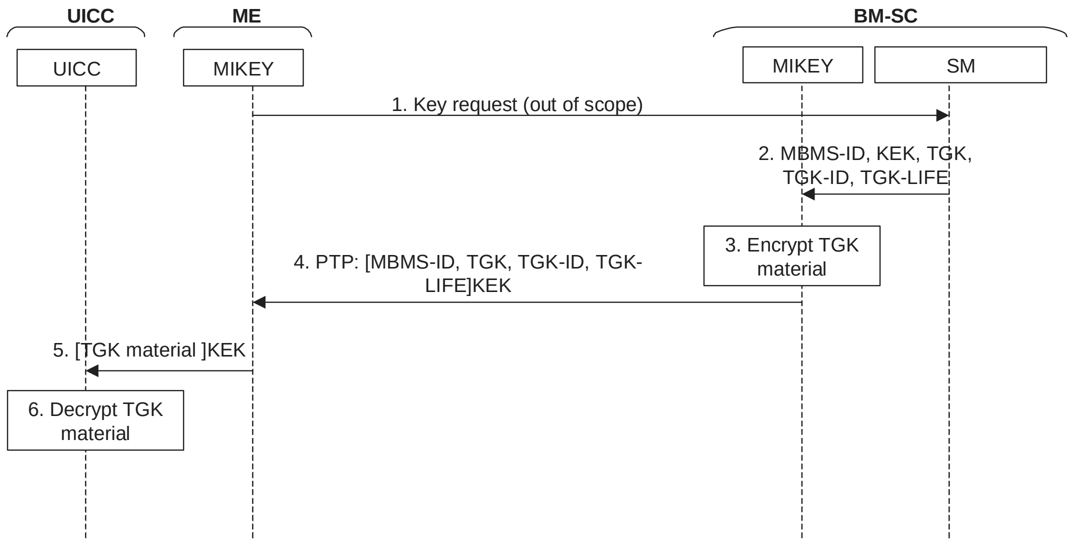
UICC
ME
BM-SC
UICC
MIKEY
MIKEY
SM
1. Key request (out of scope)
2. MBMS-ID, KEK, TGK, TGK-ID, TGK-LIFE
3. Encrypt TGK material
4. PTP: [MBMS-ID, TGK, TGK-ID, TGK-LIFE]KEK
5. [TGK material]KEK
6. Decrypt TGK material

Figure 5: PTP re-keying procedure

1. ME detects the need for a new TGK for example from TGK_LIFE or from MKI field when a TGK corresponding a TGK_ID is not available. ME sends a re-key request to SM in BM-SC.
   Note that the exact message is out of the scope of key management and it may be e.g. an HTTP request. The parameters needed are also FFS.
   Note that it is FFS what key related material e.g. TGK_ID or TGK_LIFE the ME needs from the UICC in order to control e.g. the triggering of re-keying and how ME receives this material.
2. The SM in BM-SC receives the request and sends to MIKEY in BM-SC the following parameters (It is assumed that in this point the request message has been authenticated and authorised):

- MBMS service Identifier MBMS-ID
- KEK
- TGK
- TGK identifier TGK-ID

- TGK lifetime TGK_LIFE

3. MIKEY in BM-SC encrypts TKG, TGK-ID, TGK_LIFE and MBMS-ID with KEK

4. MIKEY in BM-SC sends the encrypted key material to UE in PTP manner.

5. MIKEY in the ME passes the encrypted key material (TKG, TGK-ID, TGK_LIFE and MBMS-ID) to the UICC

6. UICC decrypts the encrypted key material with KEK.

6.2.5 TGK lifetime

The TGK lifetime is defined by operator policy. Frequent TGK updates provide more security and allow more flexible subscription management, but can also cause more signalling overhead since UEs retrieve new TGK values in point-to-point manner.

6.2.6 Updating the TGK before use

A new TGK should be provided to UICC before the new TGK value is used to derive SK-TGK values. The UE retrieves the new TGK from the SM. If many UEs try to retrieve the key simultaneously, there will be a burst of requests. This is so called implosion problem, which may be mitigated for example in the following ways:

- Many TGKs may be sent in one MIKEY message
- The UEs may be scheduled to retrieve the TGK at different times, e.g.:
    - UEs may determine a random point in time before the TGK lifetime expires
    - BM-SC may determine the retrieve time or time interval
    - A combination of the above

6.3 Protection of the transmitted traffic

Editor’s note: this section will contain the details of how traffic is protected

Editor’s note: this section may contain several protection methods.

Editor’s note: if SRTP is chosen, the master key identifier can be used to indicate the current MBMS key whichever key management method is chosen