3GPP TSG SA WG3 Security — S3#30 **S3-030511**
06 - 10 October 2003
Povoa de Varzim, Portugal

3GPP TSG-T3 Meeting #27 **Tdoc T3-030697**
Marseilles, France, 19-22 august 2003.

Title:	LS on potential USIM impact of the MBMS security framework
Release:	**Rel-6**
Source:	3GPP TSG T WG3
To:	TSG SA WG1, TSG SA WG3
Cc:	EP SCP, RAN2

Contact Person:

Name:	François Ennesser
Tel. Number:	(+33-1) 46 00 45 26
E-mail Address:	francois.ennesser@slb.com

Attachments: T3-030599

1. Overall Description:

T3 has become aware of undergoing discussions in SA3 to define a suitable key management scheme to handle the security and charging aspects of the MBMS services in release 6. In particular, it has been brought to T3 attention that some proposals under discussion, such as the one based on the 3GPP2 BCMCS model, could significantly involve the USIM capabilities for secure storage of keys and authentication.

A T3 contributing company has already started investigations on how the T3 specification could be enhanced to support the above key management scheme, if it becomes finally endorsed by SA3. It is the feeling of the contributing company that the proposed key hierarchy, where short term cipher keys are derived in a broadcast manner from medium term keys stored in the USIM, provides sizeable advantages over point-to-point distribution schemes of short term keys. They also conclude that the USIM can provide adequate functionalities to enhance the capabilities of this model, as shown in their technical contribution (attached for information).

Such enhancements imply a significant evolution of the USIM, including the definition of new commands, the proper development of which would demand some amount of time, particularly since EP SCP should preferably be involved in these definitions. Therefore, T3 will warmly welcome early guidance on the work that it can undertake at this point to ensure delivery of the desired MBMS functionalities within the Release 6 timeframe.

2. Actions:

To SA3.

ACTION: T3 asks SA3 to consider the attached contribution in their discussions and to quickly inform T3 and SA1 of their decision for the MBMS key management scheme and its potential impact on the USIM specifications.

To SA1.

ACTION: T3 asks SA1 to provide guidance on the requirements it will need to satisfy for MBMS Release 6, in order to facilitate the definition of an appropriate work item if necessary.

3. Date of Next 3GPP TSG-T WG3 Meetings:

TSG-T3 Meeting #29	18-21 November 2003	New York, USA.
TSG-T3 Meeting #30	10-13 February 2004	Sophia Antipolis, France.

Marseille, France, 19 – 22 August

Source: **Schlumberger**

Title: **USIM enhancements for MBMS support**

Document for: **Information**

1. Introduction

At SA3#29 a broad set of companies proposed a joint solution for MBMS security based on the 3GPP2 model and adapted to 3GPP. The intention of this contribution is to inform T3 of these developments as they might impact their specifications before the closing of release 6. This document also provides a framework for changes that might affect the T3 specifications, This could be used as a basis for further T3 work if the associated security architecture is finally endorsed by SA3.

The principles of this model are based on a point-to-multipoint system for distribution of short-term keys (SK), which are locally derived from a medium-term key (BAK) that is securely stored in the UICC [4].

This contribution explores what are the enhancements needed on the USIM-ME interface in order to support the MBMS re-keying mechanisms for MBMS security. It does not intend to address all the issues nor to provide the best solution, but rather to provide a starting point for more thorough work. We would like to encourage T3 delegates to study these issues in depth and make further contribution on this topic.

2. Content

3. Key renewal principles

3.1. SK Renewal.

The ME may ask SK renewal whether it finds that a new or unknown pair (BAK_ID, SK_RAND) has been distributed in the MBMS broadcasted data flow corresponding to a particular Broadcast/Multicast service. To obtain the corresponding SK, the ME has to send an **MBMSRetrieveSK** command to the USIM.

The USIM will then derive from the BAK (corresponding to the given BAK_ID) and from the SK_RAND, the subsequent SK value. The USIM will return this value to the ME.

More details of this procedure are given in section 4.1

3.2. BAK Update

BAK distribution may be performed in a point-to-point or in a point to multipoint basis. The same mechanisms apply in the USIM-ME interface for both cases.

BAK_UPDATE message is received by the mobile equipment inside a MBMSManagementRequest message. The ME is responsible to send this message to the USIM through the APDU **MBMSManagementOperation** command.

In the content of the BAK_UPDATE message, the new BAK value, and all data associated to the BAK_UPDATE request, is encrypted with a key (TK) derived form the RK. RK is provisioned to the USIM with mechanisms that are out of the scope of this document.

Once this update message is received, the USIM will be responsible to decrypt the message and perform the corresponding update in the stored BAK.

More details of this procedure are given in section 4.2 and following.

4. Additional MBMS files and data

The following additional files are needed for MBMS key management

4.1. EF_{MBMSId} (MBMS Identifiers)

This EF contains the identifiers of the MBM Services to which the user is subscribed.

<table>
<tr><td colspan="2">Identifier: TBD</td><td>Structure: linear fixed</td><td>Mandatory</td></tr>
<tr><td colspan="2">Record length: Y bytes</td><td colspan="2">Update activity: low</td></tr>
<tr><td colspan="4">Access Conditions:
READ PIN
UPDATE ADM
DEACTIVATE ADM
ACTIVATE ADM</td></tr>
<tr><td>Bytes</td><td>Description</td><td>M/O</td><td>Length</td></tr>
<tr><td>1 to Y</td><td>MBMS Identifier</td><td>M</td><td>Y bytes</td></tr>
</table>

Coding:

- Empty records shall be coded with 'FF'

4.2. EF$_{\text{MBMSBAKId}}$ (MBMS BAK Identifiers)

This EF contains the BAK Identifiers currently used for each of the MBM Service defined associated in EF$_{\text{MBMSId}}$ There is a one-to-one relationship between the records of this file and the records in EF$_{\text{MBMSId}}$

<table>
<tr><td colspan="2">Identifier: TBD</td><td>Structure: linear fixed</td><td>Mandatory</td></tr>
<tr><td colspan="2">Record length: Y bytes</td><td colspan="2">Update activity: high</td></tr>
<tr><td colspan="4">Access Conditions:
READ PIN
UPDATE ADM
DEACTIVATE ADM
ACTIVATE ADM</td></tr>
<tr><td>Bytes</td><td>Description</td><td>M/O</td><td>Length</td></tr>
<tr><td>1 to Y</td><td>BAK Identifier</td><td>M</td><td>Y bytes</td></tr>
</table>

Coding:

Empty records shall be coded with 'FF'

4.3. EF$_{\text{MBMSBAKExp}}$ (MBMS BAK Expiration)

This EF contains the BAK Expiration Date for each of the BAK keys Identified by the BAK_ID associated in EF$_{\text{MBMSBAKId}}$ There is a one-to-one relationship between the records of this file and the records in EF$_{\text{MBMSBAKId}}$

<table>
<tr><td colspan="2">Identifier: TBD</td><td>Structure: linear fixed</td><td>Mandatory</td></tr>
<tr><td colspan="2">Record length: Y bytes</td><td colspan="2">Update activity: high</td></tr>
<tr><td colspan="4">Access Conditions:
READ PIN
UPDATE ADM
DEACTIVATE ADM
ACTIVATE ADM</td></tr>
<tr><td>Bytes</td><td>Description</td><td>M/O</td><td>Length</td></tr>
<tr><td>1 to Y</td><td>BAK Expiration Date</td><td>M</td><td>Y bytes</td></tr>
</table>

Coding:

Empty records shall be coded with 'FF'

4.4. EF$_{\text{MBMSBAK}}$ (MBMS BAK Values)

This EF contains the BAK values currently used for each of the BAK keys Identified by the BAK_ID associated in $EF_{MBMSBAKId}$ There is a one-to-one relationship between the records of this file and the records in $EF_{MBMSBAKId}$

Identifier: TBD	Structure: linear fixed	Mandatory
Record length: 16 bytes	Update activity: high	

Access Conditions:

READ	ADM
UPDATE	ADM
DEACTIVATE	ADM
ACTIVATE	ADM

Bytes	Description	M/O	Length
1 to 16	BAK value	M	16 bytes

Coding:

Empty records shall be coded with 'FF'

4.5. $EF_{MBMSUpBAK}$ (Updated BAKs)

This EF contains the quadruples (MBMS Identifier, BAK Identifier , BAK expiration, BAK value) corresponding to BAK keys that have been delivered to the USIM but have not yet been used.

Identifier: TBD	Structure: cyclic	Optional
Record length: X+Y+Z+W+7 bytes	Update activity: high	

Access Conditions:

READ	ADM
UPDATE	ADM
DEACTIVATE	ADM
ACTIVATE	ADM

Bytes	Description	M/O	Length
1	Length of MBMS Identifier	M	1 byte
2 to X +1	MBMS Identifier	M	X bytes
X+2	Length of BAK Identifier	M	1 byte
X+3 to X+3+Y	BAK Identifier	M	Y bytes
X+Y+4	Length of BAK Expiration	M	1 byte
X+Y+5 to X+Y+Z+5	BAK Expiration	M	Z bytes
X+Y+Z+6	Length of BAK Value	M	1 byte
X+Y+Z+7 to X+Y+Z+W+7	BAK Value		W bytes

Coding:

Empty records shall be coded with 'FF'

4.6. KEYS

Registration key (RK)

5. APDUS

Two new APDUs are defined in the USIM-ME interface:

5.1. MBMSRetrieveSK

This command asks the USIM to generate the corresponding SK from the SK_RAND that is sent in the command and the BAK value that is referenced by the MBMS and the BAK Identifiers (also sent in the command).

The USIM shall first search if the MBMS and BAK Identifiers correspond to a stored BAK in the EF_{MBMSId} and $EF_{MBMSBAKId}$ files. If that is the case, it uses the SK_RAND value and the corresponding BAK value with the appropriate decryption algorithm to retrieve the SK value. This value is then sent to the ME in the command response.

If the (MBMS Identifier, BAK Identifier) pair does not correspond to any of the stored in EF_{MBMSId} and $EF_{MBMSBAKId}$ files, the USIM shall look for any record in the $EF_{MBMSUpBAK}$ file containing this MBMS and BAK identifier. If this record is found, The USIM will replace the BAK Identifier and BAK value contained in EF_{MBMSId} and $EF_{MBMSBAKId}$ corresponding to the MBMS Identifier by the new values. Then, it will use the BAK value with the appropriate decryption algorithm to retrieve the SK value. This value is then sent to the ME in the command response.

If none of precedent procedures apply (MBMS Identifier and BAK Identifier unavailable), the USIM will reply with an error status word or a preformatted MBMSManagementResponse (see 6.1) asking for a BAK update procedure.

Input:

- MBMS_ID, BAK_ID, SK_RAND.

Output:

- empty | MBMSManagementResponse

Command parameters

Code	Value
CLA	As specified in ETSI TS 102 221[7]
INS	FFS
P1	00
P2	00
Lc	Length of subsequent data field or empty
Data	MBMS_ID, BAK_ID, SK_RAND.
Le	Not present | Length of the response data

5.2. MBMSManagementOperation

This APDU asks the USIM to perform a Management Operation related to a Multicast/ Broadcast service.

The APDU contains an MBMSManagementRequest as input.
The command may return an MBMSManagementResponse as an output.

MBMSManagementRequest

The MBMSManagementRequest is described by an operation code (OP_Code), a counter for replay protection (OP_Counter) and an operation Body (OP_Body) that may include some padding.

The message (OP_Code, OP_Counter, OP_Body) is encrypted with a Temporary Key (TK). TK is derived from the Registration Key (RK) (stored in the USIM) and the TK_RAND value, which is sent within the command.

Additionally, the message may be integrity protected with a digest which is calculated using the unencrypted data OP_Code | OP_Counter | OP_Body

Previous to any further processing, the USIM will be on charge of decrypting the encrypted part of the MBMSManagementRequest, and perform the corresponding anti-replay and integrity test (if available).

The following values are defined for the operation code:

OP_Code :

-UPDATE_BAK
-DELETE BAK
-SUBSCRIBE
-UNSUBSCRIBE
-Others (FFS)

The procedures to manage MBMSManagementRequest for the different OP_Code are described in section 5.

<u>MBMSManagementResponse</u>

The same principles apply for MBMSManagementResponse. The following fields are present: a response code (RES_Code), a counter for replay protection (RES_Counter), which shall have the same value as the precedent (OP_Counter), and a response body (RES_Body) that may include some padding.

The message (RES_Code, RES_Counter, RES_Body) is encrypted with a Temporary Key (TK). TK is derived from the Registration Key (RK) (stored in the USIM) and the TK_RAND value, which may reuse the same value as the corresponding MBMSManagementRequest or may be generated by the USIM.

Additionally, the message may be integrity protected with a digest with is calculated using the unencrypted message (RES_Code, RES_Counter, RES_Body).

RES_Code :

-UPDATE_BAK
-DELETE BAK
-SUBSCRIBE
-UNSUBSCRIBE
-Others (FFS)

Input:

-_MBMSManagementRequest = TK_RAND, OP_Digest, (OP_Code, OP_Counter, OP_Body)*

*Encrypted with TK

Output:

- None | MBMSManagementResponse =TK_RAND, RES_Digest (RES_Code, RES_Counter, RES_Body)*

*Encrypted with TK

Command parameters

Code	Value
CLA	As specified in ETSI TS 102 221 [7]
INS	TBD
P1	00
P2	00
Lc	Length of subsequent data field or empty
Data	TK_RAND, OP_Digest, (OP_Code, OP_Counter, OP_Body)*
Le	Length of the response data

6. MBMSManagementOperations

6.1. UPDATE_BAK

Procedure:

The USIM shall search for the given MBMS_ID in the EF$_{MBMSId}$. If this record exists it shall then create a new entry in the EF$_{MBMSUpBAK}$ using the MBMS_ID, BAK_ID, BAK_Expiration and BAK_Value given values.

The MBMSManagementResponse will be returned whether it is asked in the Acknowledge_Needed data field of the corresponding MBMSManagementRequest. MBMSManagementResponse will contain the MBMS_ID and the BAK_ID as acknowledge information and a MBMS_Result field describing the results of the Update BAK procedure (see coding in Annexe 1).

Additionally, the MBMSManagementResponse with UPDATE_BAK as OP_Code may be used as response message of the MBMSRetrieveSK command when the BAK_ID given is unavailable. In this case, the USIM shall include MBMS_REQUEST field in the RES_Body and a new RES_Counter shall be generated.

MBMSManagementRequest encrypted content

OP_Code:

-UPDATE_BAK

OP_Counter:

OP_Body:

-MBMS_ID
-BAK_ID
-BAK_Expiration
-BAK_Value

-Acknowledge_Needed

<u>MBMSManagementResponse encrypted content</u>

RES_Code:

-UPDATE_BAK

RES_Counter

RES_Body

-MBMS_ID

-BAK_ID

-MBMS_RESULT | MBMS_REQUEST

6.2. DELETE BAK

<u>Procedure:</u>

The USIM shall search for the given (MBMS_ID, BAK_ID) pair in the EF_{MBMSId} and $EF_{MBMSBAKId}$ files. If the record is found, it shall erase (fill up with ‘FF’) the records corresponding to this BAK in $EF_{MBMSBAKId}$ and $F_{MBMSBAK.}$

If this research was unsuccessful, the USIM shall look for the (MBMS_ID, BAK_ID) pair in the $EF_{MBMSUpBAK}$ file. If this record is found, The USIM shall remove it.

The MBMSManagementResponse will be returned whether it is asked in the Acknowledge_Needed data field of the coresponding MBMSManagementRequest. The MBMSManagementResponse will contain the MBMS_ID and the BAK_ID as acknowledge information and a MBMS_Result field describing the results of the BAK deletion procedure (see coding in Annexe 1).

<u>MBMSManagementRequest encrypted content</u>

OP_Code :

-DELETE BAK

OP_Counter:

OP_Body:

-MBMS_ID

-BAK_ID

-Acknowledge_Needed

<u>MBMSManagementResponse encrypted content</u>

RES_Code:

- DELETE BAK

RES_Counter

RES_Body

-MBMS_ID
-BAK_ID
-MBMS_RESULT

6.3. SUBSCRIBE

Procedure:

The USIM shall create a new entry using an empty record in the EF_{MBMSId} and fill it with the new MBMS_ID. The BAK_ID, BAK_Expiration and BAK_Value values are inserted in the corresponding records of $EF_{MBMSBAKId}$, $EF_{MBMSBAK}$ and $EF_{MBMSBAKExp}$ files.

The MBMSManagementResponse will be returned whether it is asked in the Acknowledge_Needed data field of the coresponding MBMSManagementRequest. The MBMSManagementResponse will contain the MBMS_ID and the BAK_ID as acknowledge information and a MBMS_Result field describing the results of the MBMS subscription procedure (see coding in Annexe 1).

MBMSManagementRequest encrypted content

OP_Code :

- SUBSCRIBE

OP_Counter:

OP_Body:

-MBMS_ID
-BAK_ID
-BAK_Expiration
-BAK_Value
-Acknowledge_Needed

MBMSManagementResponse encrypted content

RES_Code:

-SUBSCRIBE

RES_Counter

RES_Body

-MBMS_ID
-BAK_ID

-MBMS_RESULT

6.4. UNSUBSCRIBE

Procedure:

The USIM shall search for the given MBMS_ID in the EF_{MBMSId}. The USIM shall then delete the entry corresponding to the given MBMS_ID in the EF_{MBMSId}, $EF_{MBMSBAKId}$, $EF_{MBMSBAKExp}$ and $EF_{MBMSBAK}$ files.

The MBMSManagementResponse will be returned whether it is asked in the Acknowledge_Needed data field of the coresponding MBMSManagementRequest. The MBMSManagementResponse will contain the MBMS_ID as acknowledge information and a MBMS_Result field describing the results of the MBMS unsubscription procedure (see coding in Annexe 1).

MBMSManagementRequest encrypted content

OP_Code :

- UNSUBSCRIBE

OP_Counter:

OP_Body:

-MBMS_ID
-Acknowledge_Needed

MBMSManagementResponse encrypted content

RES_Code:

-UNSUBSCRIBE

RES_Counter

RES_Body

-MBMS_ID

7. Extensions and additional comments

7.1. Broadcast of BAK Update

In order to support BAK renewal via broadcast (and eventually, any other management operation), it can be possible to provide different RKs to the same subscriber. Some of these new keys, hereafter referred as Registration Broadcast Key (RBK), could be common to all (or to a subset of) the subscribers of particular Multicast/Broadcast services.

A new data field is needed in the MBMSManagementRequest and MBMSManagementResponse messages to identify the RBK that is used to encrypt the MBMSManagementOperation command.

Note1: This procedure is FFS
Note2: LKH model may be used to support different hierarchies and groups of RK keys.

7.2. Charging issues. Counters

In order to provide additional control of the services being used by the subscriber, the following enhancements may be provided.

At least two new files:

<u>SK counter (SK_Counter):</u> Defining, for each Broadcast/Multicast Service, the number of times that the MBMSRetrieveSK command has been performed.

<u>Maximum SK counter value (Max_SK_Counter):</u> Defining the maximum value that the corresponding SK_Counter may have.

Note1: MBMSRetrieveSK needs to update the SK_counter(s), corresponding to the MBMS_ID, when the command is using a SK_RAND different from the previous one.

Note2: UPDATE_BAK procedure resets the SK_Counter(s) corresponding to the MBMS_ID

Note3: SUBSCRIBE procedure will describe the number of SK counters and Maximum SK counter values associated to the Broadcast/Multicast Service.

Note4: An Additional MBMSManagementOperation may be needed to retrieve the additional information of a subscription (RETRIEVE_SUBSCRIPTION_INFO) e.g. counter values.

Note5: An Additional MBMSManagementOperation may be needed to update the counter values (UPDATE_COUTERS).

With this minimum set of changes, operators providing MBMS, could take benefits from a suitable granularity in the control of the usage of services by their subscribers. For instance, performing a regular (and frequent) SK renewal, it is possible to know the exact amount of time than the user has been accessing the Broadcast/Multicast service. Additionally, maximum values in the counters may enable flexible subscriptions (e.g. a user subscribed to a particular MBMS service X hour a month with X that can be updated remotely). Consequently, It is then possible charging methods based in the amount of data or time.

These new features are FFS.

7.3. MBMS Service activated in USIM service table

A new service number is needed in the EF_{UST} (USIM Service Table) to indicate that the USIM is MBMS capable [7]

-Services
Contents: Service n°1: Local Phone Book
…

Service n°xx: MBMS

7.4. OTA Management

Taking advantages from the existing infrastructure, OTA security mechanisms [4] could be used to perform MBMS management operations by:

1- Sending an MBMSManagementOperation command by OTA
2- Accessing and updating MBMS related files (e.g. BAK files, counters...)
3- Accessing and updating MBMS registration keys (RK, RBKs,...)

7.5. OTA and PreRelease 6 USIMs

In order to support MBMS capabilities in pre-Release 6 USIMs, it could be possible to implement these MBMS new features by a javacard application that can be downloaded by OTA using the existing procedures [4]

When the ME detects that the MBMS capabilities are not present in the USIM Service table, it could try to select this MBMS application (e.g. by AID). In that case, the two needed commands (MBMSRetrieveSK and MBMSManagementOperation) would be managed by the MBMS application itself and not by the USIM.

7.6. 3GPP2

The same principles and commands may be applied to 3GPP2 BCMCS Security in the RUIM context.

8. ANNEXE 1. DATA Coding example

The following section describes the coding of the data fields defined in this document:

OP_Code or RES_Code values:

DATA Field	Length	Value
UPDATE_BAK	1	‘01’
DELETE BAK	1	‘02’
SUBSCRIBE	1	‘03’
UNSUBSCRIBE	1	‘04’

OP_Counter or RES_Counter

DATA Field	Length	Value
OP_Counter	5	
RES_Counter	5	

OP_Body or RES_BODY TLVs:

DATA Field	Length of tag	Tag	Length
MBMS_ID	1	0x01	
BAK_ID	1	0x02	
BAK_Expiration	1	0x03	
BAK_Value	1	0x04	16
MBMS_RESULT	1	0x05	
MBMS_REQUEST	1	0x06	
Acknowledge_Needed	1	0x07	

MBMS_RESULT | MBMS_REQUEST coding

TBD

Other TLVs:

DATA Field	Length of tag	Tag	Length
SK_RAND	1	0x01	4
TK_RAND	1	0x02	8
OP_Digest	1	0x03	8
RES_Digest	1	0x04	8

9. References

[1] 3GPP TS 22.146: "Multimedia Broadcast/Multicast Service; Stage 1".

[2] 3GPP TR 23.846: " Multimedia Broadcast/Multicast Service; Architecture and Functional Description".

[3] 3GPP TS 33.246, Technical Specification Group Services and System Aspects; Security; Security of Multimedia Broadcast/Multicast Service;

[4] 3GPP TS 23.048: "Security mechanisms for the (Universal) Subscriber Interface Module (U)SIM Application Toolkit; Stage 2".

[5] S3-030040 MBMS Security Framework and Pseudo-CR to 33.246. Qualcomm

[6] ETSI TS 102 221. Physical and logical characteristics

[7] 3GPP TS 31.102. Characteristics of the USIM Application