|  |  |
| --- | --- |
| Project | **IEEE 802.21 MIHS****<**[**http://www.ieee802.org/21/**](http://www.ieee802.org/21/)**>** |
| Title | **Proposed remedy for Comment 102** |
| DCN | **21-14-0097-01-MuGM** |
| Date Submitted | **May 27, 2014** |
| Source(s) | Yoshikazu Hanatani (Toshiba) |  |
| Re: | IEEE 802.21 Session #62 in Waikoloa |
| Abstract | This document describes a proposed remedy for LB7c Comments #102 about group manipulation commands and group addressed commands protection. |
| Purpose | For LB7c Comment Resolution |
| Notice | This document has been prepared to assist the IEEE 802.21 Working Group. It is offered as a basis for discussion and is not binding on the contributing individual(s) or organization(s). The material in this document is subject to change in form and content after further study. The contributor(s) reserve(s) the right to add, amend or withdraw material contained herein. |
| Release | The contributor grants a free, irrevocable license to the IEEE to incorporate material contained in this contribution, and any modifications thereof, in the creation of an IEEE Standards publication; to copyright in the IEEE’s name any IEEE Standards publication even though it may include portions of this contribution; and at the IEEE’s sole discretion to permit others to reproduce in whole or in part the resulting IEEE Standards publication. The contributor also acknowledges and accepts that IEEE 802.21 may make this contribution public. |
| Patent Policy | The contributor is familiar with IEEE patent policy, as stated in [Section 6 of the IEEE-SA Standards Board bylaws](http://standards.ieee.org/guides/opman/sect6.html#6.3) <[http://standards.ieee.org/guides/bylaws/sect6-7.html#6](http://127.0.0.1:4664/cache?event_id=757737&schema_id=1&s=5X0vID10lu_E6yrIkWkNd4Wz2H8&q=hancock)> and in *Understanding Patent Issues During IEEE Standards Development* <http://standards.ieee.org/board/pat/faq.pdf> |

# Comments

* Cmt #102: It has never been clear that the MIHF in GMCS is also a recipient of group manipulate command. However, it "stores the pairs of a Node Index and a corresponding Node Key (i.e., device keys) to retrieve a group key from a GKB," This is very confusing! What device keys the MIHF of GMCS should have? Its own device keys or device keys for all the potential recipients? If these are its own device keys as a recipient of group manipulation command, then the purpose of "retrieve a group key from a GKB" does not sound right, because "MGK" is also an item in this Recipient Information Base. It does not need to retrieve it from GKB. If these are device keys for all the potential recipients, then they have already stored in the MIH User of GMCS (see 9.5.3.1.1). .

# Discussion:

1. Remove device keys from the MIHF in GMCS.

Change the definition of Information bases as follows.

|  |  |  |
| --- | --- | --- |
| MGK | OCTET(16) | This is the base data type of an master group key. |
| GROUP\_MEMBERSHIP\_BASE | LIST(SEQUENCE(MIHF\_ID,TRANSPORT\_ADDR,CHOICE(MGK,NULL),CHOICE(SEQUENCE\_NUMBER,NULL),CHOICE(SAID,NULL))) | The information base stores group membership information to manage a belonged group. |
| GROUP\_RECIPIENT\_INFO~~MIHF\_BASE~~ | SEQUENCE(LIST(GRP\_MGMT\_TREE\_NODE),LIST(CERTIFICATE),~~LIST(GROUP\_MEMBERSHIP\_BAS~~~~E)~~) | The information base stores an individual device keys to decrypt GKB, Certificates to verify digital signatures~~, and Group\_Membership\_Base to manage belonged groups~~. |

GMCS uses GROUP\_MEMBERSHIP\_BASE.

GMCR uses GROUP\_MEMBERSHIP\_BASE and GROUP\_RECIPIENT\_INFO.

So, we can remove the device keys from the MIHF in GMCS.

1. Transporting method of MGK to the local MIHF in GMCS.

The local MIHF in GMCS directly receives an MGK from the primitive of the group manipulation commands. To realize this procedure, we need to amend two primitives for the group manipulation commands.

We need explanation texts for this procedure.

# Suggested Remedy: (Alternative 2 in **21-14-0097-01-MuGM**)

 [1] Change 7.4.31.3.2 as follows:

**7.4.31.3.2 Semantics of service primitive**

MIH\_MN\_Group\_Manipulate.response (

DestinationIdentifier,

TargetIdentifier,

MulticastAddress,

MasterGroupKey,

SubgroupRange,

UserSpecificData,

CompleteSubtree,

GroupKeyData,

GroupStatus

)

Parameters:

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Description |
| DestinationIdentifier | MIHF\_ID | Specifies the MIHF ID of the destination of the primitive. |
| TargetIdentifier | MIHF\_ID | The target MIHF group identifier for the group operation. |
| MulticastAddress | TRANSPORT\_ADDR | (Optional) Multicast address corresponding with the target group identifier. |
| MasterGroupKey | MGK | (Optional) The master group key associated with the target MIHF group identifier. |
| SubgroupRange | SUBGROUP\_RANGE | (Optional) Subgroup to process the command.a |
| UserSpecificDatab | OCTET\_STRING | (Optional) Auxiliary data. |
| CompleteSubtree | COMPLETE\_SUBTREE | (Optional) Complete Subtree data. |
| GroupKeyData | GROUP\_KEY\_DATA | (Optional )Encrypted group key. |
| GroupStatus | GROUP\_STATUS | Status of the group operation. |

[2] Change 7.4.32.1.2 as follows:

**7.4.32.1.2 Semantics of service primitive**

MIH\_Net\_Group\_Manipulate.request (

DestinationIdentifier,

ResponseFlag,

GroupKeyUpdateFlag,

TargetIdentifier,

MulticastAddress,

MasterGroupKey,

SubgroupRange,

UserSpecificData,

CompleteSubtree,

GroupKeyData

)

Parameters:

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Description |
| DestinationIdentifier | MIHF\_ID | Specifies group MIHF-ID of the remote MIHF peers. DestinationIdentifier may be different from TargetIdentifier. |
| ResponseFlaga | RESPONSE\_FLAG | (Optional) Flag that represents whether or not a response is needed. |
| GroupKeyUpdateFlag | GROUP\_KEY\_UPDATE\_FLAG | Flag that represents whether or not a group key in GroupKeyData is updated. |
| TargetIdentifier | MIHF\_ID | The target MIHF group identifier for the group operation. |
| MulticastAddress | TRANSPORT\_ADDR | (Optional) Multicast address corresponding with the target group identifier. |
| MasterGroupKey | MGK | (Optional) The master group key associated with the target MIHF group identifier. |
| SubgroupRange | SUBGROUP\_RANGE | (Optional) Subgroup to process the command |
| UserSpecificData | OCTET\_STRING | (Optional) Auxiliary data. |
| CompleteSubtree | COMPLETE\_SUBTREE | Complete Subtree data. |
| GroupKeyData | GROUP\_KEY\_DATA | (Optional) Encrypted group key. |

[3] Change 9.5.3.1.1 as follows:

**9.5.3.1.1 MIH user of a GMCS**

~~Required components in an MIH User of a GMCS in a PoS relevant to group manipulation and group addressed commands are listed as follows:~~

An MIH User, which is a command center, of a GMCS is shown in figure 38. MIHF needs the MGK to encrypt service specific TLVs in group manipulation and group addressed commands. MIHF obtains the MGK via MIH\_Net\_Group\_Manipulate.request primitive. Required components in an MIH User of GMCS relevant to group manipulation and group addressed commands are listed as follows:

**Figure 38: GMCS functional components**

[4] Change Step e) in 9.5.3.1.1 as follows:

e) (Optional) Define GroupKeyData:

 i. When MGK is not used, this process is skipped.

 ii. Send the MGK and the CompleteSubtree to the MasterGroupKeyWrapping procedure, and receive GroupKeyData. The procedure accesses the *Group Management Tree Information Base* to refer all the pairs of a Node Index and a corresponding Node Key.

 iii. Set MGK to MasterGroupKey.

[5] Change 9.5.3.1.2 as follows:

**9.5.3.1.2 MIHF of a GMCS**

 Required components relevant to group manipulation and group commands are listed as follows:

A signing key (of type SIGNING\_KEY as defined in Table F.25). The key is for creation of a signature at the GMCS.

~~A~~ *~~Recipient Information Base~~* ~~(of type RECIPIENT\_MIHF\_BASE as defined in Table F.25)~~ A *Group Membership Information Base* (of type GROUP\_MEMBERSHIP\_BASE as defined in Table F.25) stores ~~the pairs of a Node Index and a corresponding Node Key (i.e., device keys) to retrieve a group key from a GKB, the certificate used to verify digital signatures, and~~ the information required to send commands to the group, i.e., the MIHF Group ID, the transport address used. If the service specific TLVs carried in group addressed commands and group manipulation commands addressed to the group are encrypted, the *Group Membership Information* Base also stores the MGK, the sequence number and the SAID associated with ~~to~~ the group.

It is assumed that the MIHF is able to obtain in some way a multicast address associated with an MIHF Group ID. The multicast address may be contained in the MIH\_Net\_Group\_Manipulate.request received from the MIH User. ~~In this case, if the TargetGroupIdentifier in the received request is not registered in the~~ *~~Recipient Information Base~~*~~, obtain the multicast address associated with the TargetGroupIdentifier and update the~~ *~~Recipient Information Base~~* ~~with the DestinationIdentifier and the associated multicast address.~~ The MIHF ~~of the Command center~~ receives an MIH\_Net\_Group\_Manipulate.request, which is generated by the MIH User, the MIHF generates and sends an MIH\_Net\_Group\_Manipulate indication/request message to a multicast group. Note that this behavior depends on the ResponseFlag parameter. When “ResponseFlag=1”, the MIHF will generate MIH\_Net\_Group\_Manipulate request message. When “ResponseFlag=0”, the MIHF will generate MIH\_Net\_Group\_Manipulate indication message.

In the following we detail the steps performed to generate the message:

 a) Generate a Source MIHF ID TLV using its own MIHF ID.

 b) Generate a Destination MIHF ID TLV from the DestinationIdentifier in the received MIH\_Group\_Manipulate.request.

c) If GroupKeyData is accompanied, generate a Sequence Number TLV.

 d) The MIHF generates optionally a Multicast Address TLV. If the MIH\_Net\_Group\_Manipulate.request contains a MulticastAddress parameter, the parameter is contained in the Multicast Address TLV. Else if the MIH\_Net\_Group\_Manipulate.request does not contain a MulticastAddress parameter, the MIHF decides a multicast address parameter.

 e) If the MIH\_Net\_Group\_Manipulate.request contains a SubgroupRange, it generates a SubgroupRange TLV from the SubgroupRange.

 f) If the MIH\_Net\_Group\_Manipulate.request contains a UserSpecificData, it generates an Aux Data TLV from the UserSpecificData.

 g) Generate a SubtreeFlag TLV from the SubtreeFlag in the received MIH\_Net\_Group\_Manipulate.request.

 h) Generate a Complete Subtree TLV from the CompleteSubtree in the received MIH\_Net\_Group\_Manipulate.request.

 i) If the MIH\_Net\_Group\_Manipulate.request contains a GroupKeyData, it generates a Group Key Data TLV from the GroupKeyData.

 j) If GroupKeyData is accompanied, generate an SAID Notification TLV. If KeyUpdateFlag=0, the TLV contains the security association identifier associated with the GroupKeyData. Otherwise, the TLV contains a newly allocated security association ID for the GroupKeyData. The security association identifier contained in the TLV is later on contained in an SAID TLV used for MIH messages encrypted by the group key corresponding to the GroupKeyData .

 k) If a security association ID with respect to the DestinationIdentifier is stored in its own *Recipient Information Base*, it encrypts Service Specific TLVs of this group manipulation command as shown in 9.6.4.

 l) Generate a Signature TLV as shown in 9.6.4 using the signing key of the MIHF.

 m) If ResponseFlag=0, generate an MIH\_Net\_Group\_Manipulate indication using the preceding TLVs, else generate an MIH\_Net\_Group\_Manipulate request using the preceding TLVs.

 n) Update the *Group Membership Information* Base with TargetGroupIdentifier, the multicast address parameter, the sequence number and the SAID. If a MasterGroupKey is contained in the

MIH\_Net\_Group\_Manipulate.request, also update the *Group Membership Information* Base with the MasterGroupKey as the MGK.

Figure 40, shows a flow diagram summarizing the steps performed by the MIHF at a PoS, described in this Clause.

[6] Change 9.5.3.2 as follows.

**9.5.3.2 Procedures for group manipulation command recipients (GMCR)**

 Required components relevant to group manipulation and group commands are listed as follows:

A *~~Recipient Information Base~~ Group Recipient Information* (of type ~~RECIPIENT\_MIHF\_BASE~~ GROUP\_RECIPIENT\_INFO as defined in Table F.25) containing the pairs of a Node Index and a corresponding Node Key (i.e., device keys) to retrieve an MGK from a GKB, the certificate used to verify digital signatures~~, and the information required to send commands to the group, i.e., the MIHF Group ID, the transport address used, the MGK, the sequence number and the SAID associated to the group~~. When a client MN receives a group manipulation command, i.e., an MIH\_Net\_Group\_Manipulate indication/request message, issued by a GMCS, the MIHF of the GMCR processes the command.

*Group Membership Information Base* (of type GROUP\_MEMBERSHIP\_BASE as defined in Table F.25) stores the information required to receive commands to the group, i.e., the MIHF Group ID, the transport address used. If the service specific TLVs carried in group addressed commands and group manipulation commands addressed to the group are encrypted, the *Group Membership Information* Base also stores the MGK, the sequence number and the SAID associated with the group.

 a) The MIHF obtains a Source Identifier from the Source MIHF ID TLV.

 b) The MIHF verifies the Signature TLV using a verification key in the certificate corresponding to the obtained SourceIdentifier stored in the *~~Recipient Information Base~~ Group Recipient Information*. If the verification fails, the MIHF shall cancel the following steps and stop processing the command.

 c) The MIHF checks the DestinationIdentifier in the Destination MIHF ID TLV. If the DestinationIdentifier does not match one of the following MIHF IDs, the MIHF shall cancel the following steps and stop processing the command: (i) An MIHF Group ID corresponding to a broadcast address, (ii) an MIHF Group ID which is registered with a multicast address in the *~~Recipient Information~~ Group Membership Information Base*, or (iii) the MN's own MIHF ID.

 d) The MIHF decrypts the payload if it is encrypted, i.e., if it is a Security TLV. The decryption key is derived from the MGK associated with the DestinationIdentifier in the *~~Recipient Information~~ Group Membership Information Base*.

 1) In case an MN cannot decrypt the Security TLV, the message will be silently discarded.

 e) If a SubgroupRange TLV exists in the indication, the MIHF obtains a SubgroupRange and checks whether its own Leaf Number is contained in the SubgroupRange or not. If it is not, the MIHF shall cancel the following steps and stop processing.

 f) The MIHF obtains a TargetIdentifier in the Target Identifier TLV, a SubtreeFlag in the SubtreeFlag TLV, and a CompleteSubtree in the Complete Subtree TLV.

k) If a MulticastAddress TLV exists in the indication, the MIHF obtains a MulticastAddress. Otherwise, the MIHF obtains a multicast address with respect to the TargetIdentifier from a server (Note that this operation is out of the scope of this specification).

 l) If a GroupKeyData TLV exists in the indication, the MIHF obtains a GroupKeyData and derives ~~a group key~~ an MGK by processing the GroupKeyData using a Node Key corresponding with the Node Index as described in 9.5.2.2.

 m) If a SAID TLV exists in the indication, the MIHF obtains a SAID.

 n) If a Sequence Number TLV exists in the indication, the MIHF obtains a SequenceNumber. If the GroupKeyUpdateFlag is “1,” the MIHF resets the SequenceNumber to an initial value.

 o) The MIHF checks whether the TargetIdentifier obtained in Step f) has already been registered or not in the *~~Recipient Information Base~~ Group Recipient Information*. If it has been, go to Step p) [Stay]. Otherwise, go to Step r) [Join].

 p) [Stay] The MIHF updates the multicast address, the group key and the SAID, and the SequenceNumber, with respect to the TargetIdentifier, in the *~~Recipient Information~~ Group Membership Information Base*.

 q) The MIHF throws an MIH\_Net\_Group\_Manipulate.indication described in 7.4.32.2 to the MIH User. The GroupStatus field of the indication shall be “Unchanged successful” (5). The procedure of command processing terminates.

 r) [Join] The MIHF starts listening to the multicast address associated with the TargetIdentifier. The MIHF saves in the *~~Recipient Information~~ Group Membership Information Base* the TargetIdentifier, the associated multicast address, the group key (Option), the SequenceNumber (Option), and the SAID (Option).

 s) The MIHF issues an MIH\_Net\_Group\_Manipulate.indication described in 7.4.32.2 to the MIH User. The GroupStatus field must be “Join operation successful” (0). The procedure of command processing terminates.

 t) The MIHF checks whether the TargetIdentifier has already been registered or not in the *~~Recipient Information~~ Group Membership Information Base*. If it has been, go to Step p) [Leave]. Otherwise, the MIHF terminates the procedure of command processing.

 u) [Leave] The MIHF finds the multicast address recorded on the same row as the TargetIdentifier obtained in Step k) and the MIHF stops listening to it. The MIHF removes the row that has the TargetIdentifier.

 v) The MIHF throws an MIH\_Net\_Group\_Manipulate.indication described in 7.4.32.2 to the MIH User. The GroupStatus field must be “Leave operation successful” (3). The procedure of command processing terminates.

 Figure 42 summarizes the steps followed by the MIHF on the MN upon reception of an MIH\_Net\_Group\_Manipulation.indication.

Subclause 7.4.31 introduces a mechanism enabling the recipient to trigger the Join/Leave operations controlled by the command center. In order to do so, the MIH User located at the recipient notifies the command center of its desire to Join or Leave a group through the use of the MIH\_MN\_Group\_Manipulate primitive. The MIHF of the PoS with command center, upon receiving the associated request message, performs the same process as defined in this Clause, for the use of the MIH\_Net\_Group\_Manipulate, although in this case, the group to be manipulated is provided by the recipient. The resulting GKB parameters are returned to the recipient in the MIH\_MN\_Group\_Manipulate response message.

[7] Change Table F.25 as follows.

|  |  |  |
| --- | --- | --- |
| MGK | OCTET(16) | This is the base data type of an master group key. |
| GROUP\_MEMBERSHIP\_BASE | LIST(SEQUENCE(MIHF\_ID,TRANSPORT\_ADDR,CHOICE(MGK,NULL),CHOICE(SEQUENCE\_NUMBER,NULL),CHOICE(SAID,NULL))) | The information base stores group membership information to manage a belonged group. |
| GROUP\_RECIPIENT\_INFO~~MIHF\_BASE~~ | SEQUENCE(LIST(GRP\_MGMT\_TREE\_NODE),LIST(CERTIFICATE),~~LIST(GROUP\_MEMBERSHIP\_BAS~~~~E)~~) | The information base stores an individual device keys to decrypt GKB, Certificates to verify digital signatures~~, and Group\_Membership\_Base to manage belonged groups~~. |