# IEEE 802.21 MEDIA INDEPENDENT HANDOVER

DCN: 21-12-0157-00-MuGM

Title: **Proposal to IEEE 802.21d based on MKB**

Date Submitted: November, 4th, 2012

Presented at IEEE 802.21 session #53 in San Antonio

Authors or Source(s):

**Yoshikazu Hanatani, Toru Kambayashi (Toshiba)**

Abstract: This proposal is a contribution for the 802.21d in response to 802.21-12-0091-06-MuGM-requirements-document. This proposal has two procedures: a group manipulation procedure based on MKB and a group command procedure.

# IEEE 802.21 presentation release statements

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.

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 this contribution may be made public by IEEE 802.21.

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/bylaws/sect6-7.html#6> and in *Understanding Patent Issues During IEEE Standards Development* http://standards.ieee.org/board/pat/faq.pdf>

# Outline

- Introduction
- Definitions
- Architecture and Concept
  - System model
  - Concept of our solution
  - MKB Basics
  - Mapping to MIH framework
- Prerequisites for System
- Proposal
  - Group ID
  - Group manipulation
  - Group command
  - MIH Primitives and MIH Messages
  - Group key hierarchy
- Conformance to the requirements
- Summary

# Introduction

- This proposal is a contribution to the 802.21d in response to 802.21-12-0091-06-MuGM-requirements-document.
- This proposal provides a secure management method of multicast groups using Media Key Block mechanism.

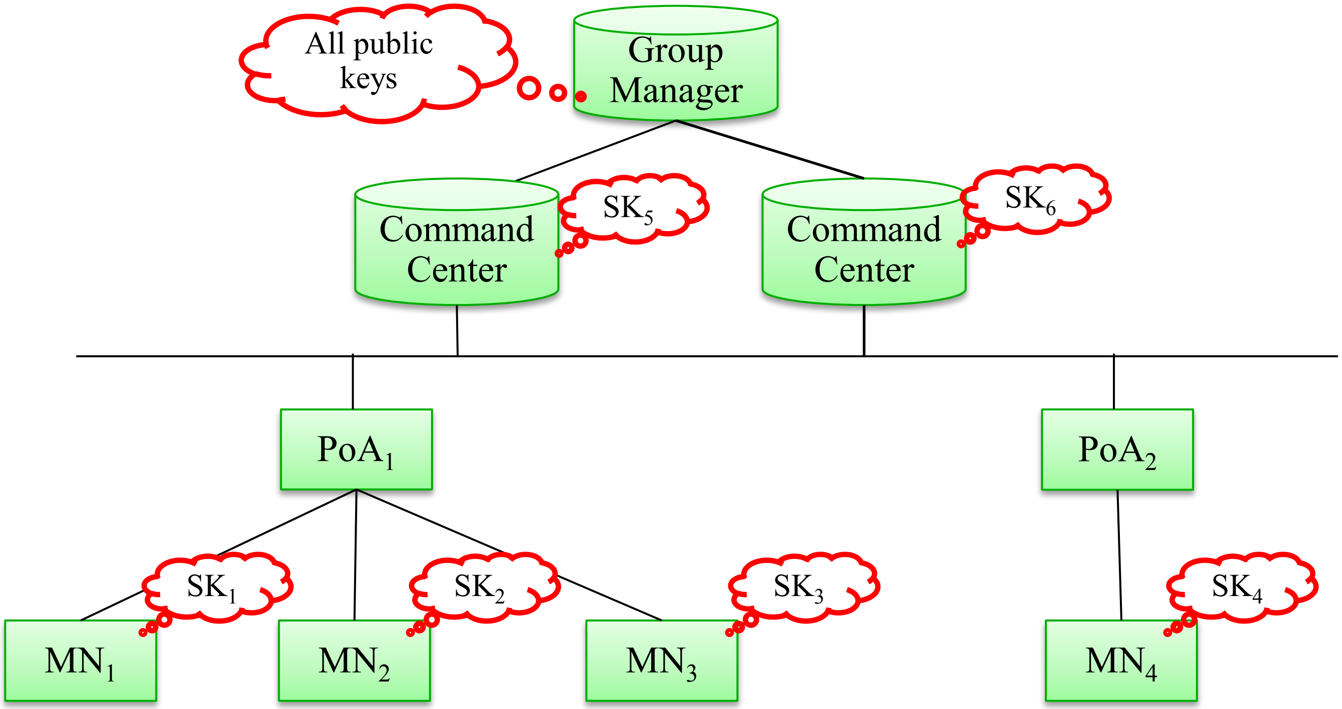
# Definitions

- MKB: Media Key Block
- MIHF ID: Should be redefined as Individual MIHF ID and Group MIHF ID
- Individual MIHF ID: current MIHF ID
- Group MIHF ID: Should newly defined
- Group manipulation command: A command to make members join in a group or leave from the group.
- Group command: A command issued to members which belongs to a group.
- Group Manager: A server which issues a group manipulation command.
- Command Center: A server which issues a group command.

# Outline

- Introduction
- Definitions
- Architecture and Concept
  - System model
  - Concept of our solution
  - MKB Basics
  - Mapping to MIH framework
- Prerequisites for System
- Proposal
  - Group ID
  - Group manipulation
  - Group command
  - MIH Primitives
  - MIH Messages
  - Group key hierarchy
- Conformance to the requirements
- Summary

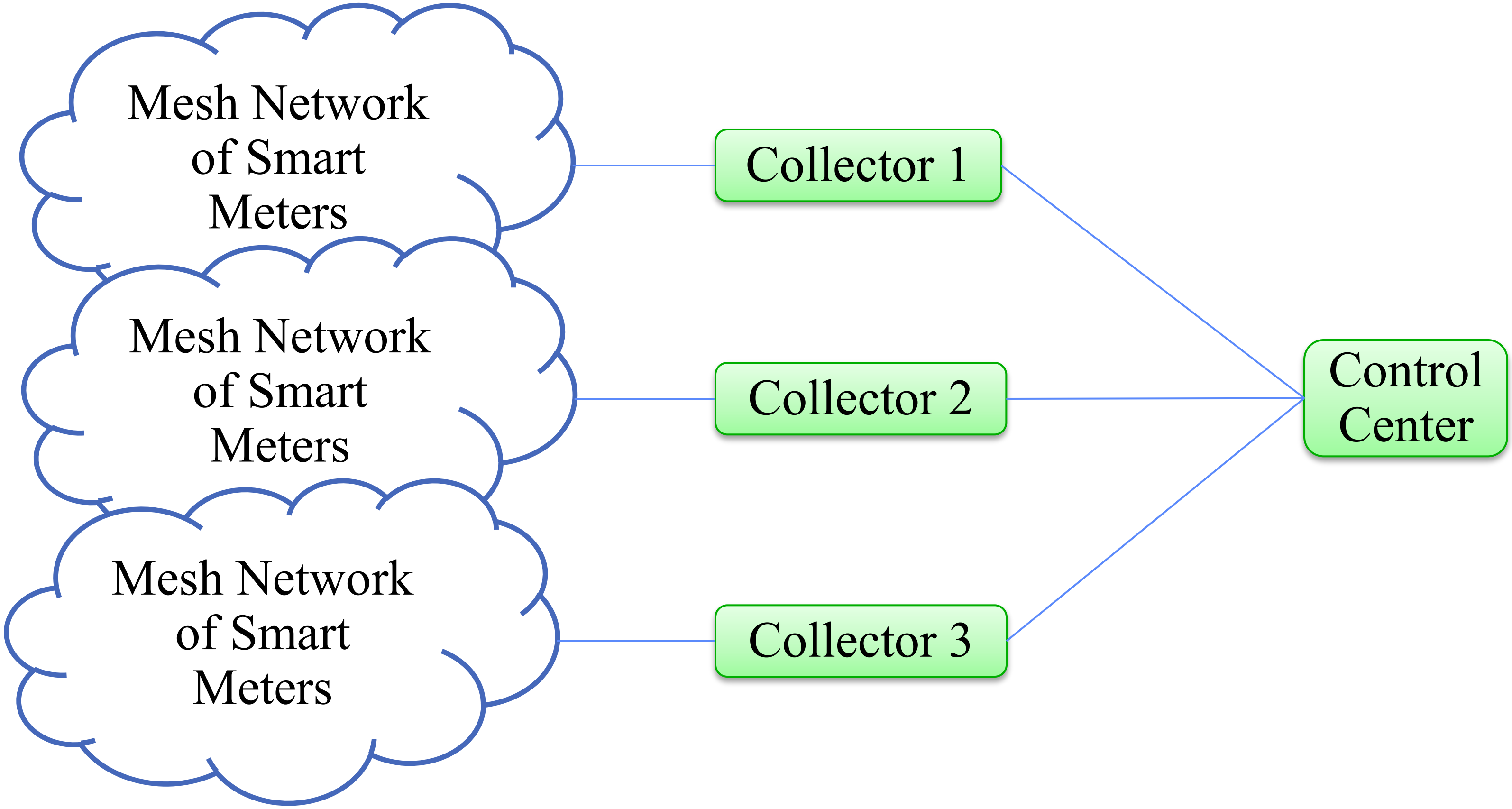
# System model



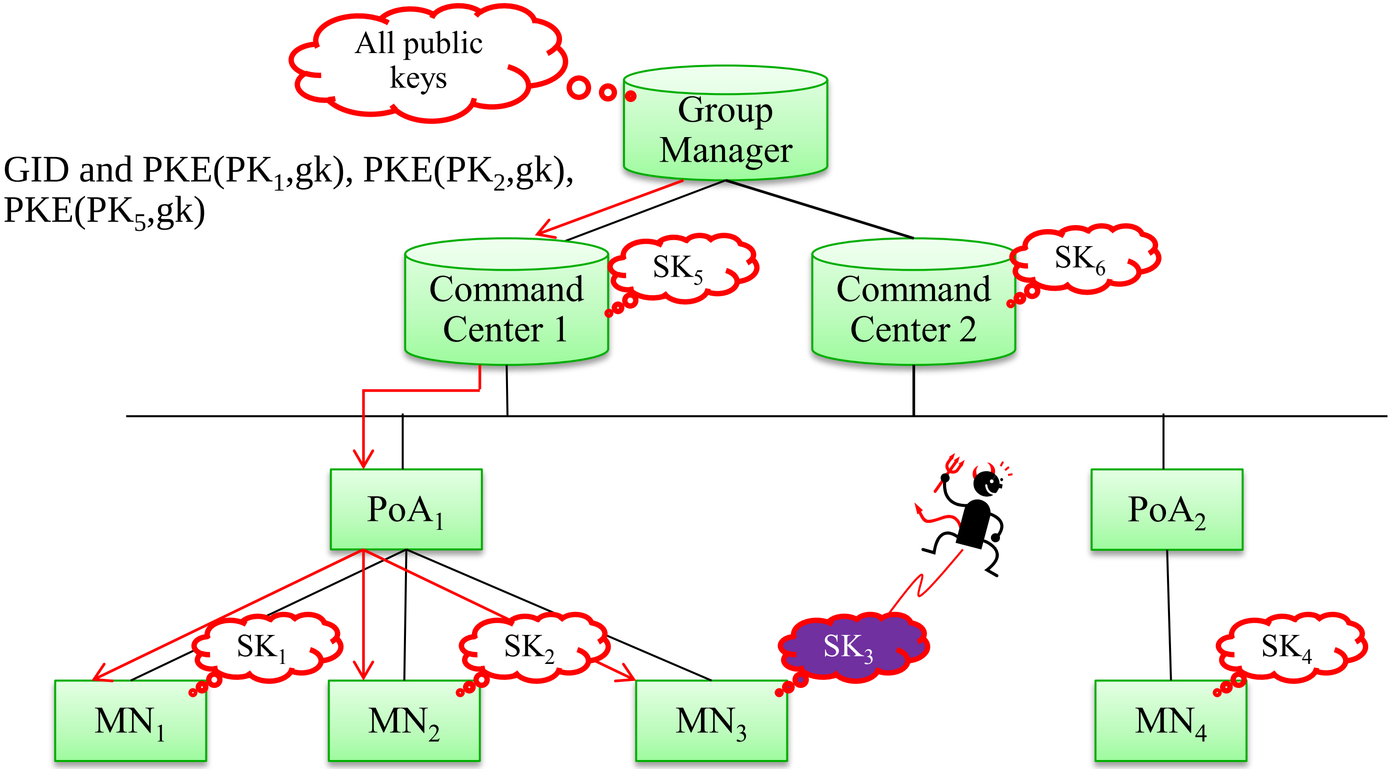
- **Assumption**: Group Manager, Command Center and MNs have long-term keys (e.g., SKs, all public keys).
- **System description**:
  - Group Manager issues a group manipulation command to distribute a group ID and a group key to the MNs of a group via Command Center. Arguments of a group manipulation command can optionally be encrypted by the long-term key.
  - Command Center issues a group command to the MNs of the group designated by a group ID. A group command can optionally be encrypted by the group key assigned to the group.

# Use Cases Architecture

- Applications
  - Handover for load balance
  - Handover in case of a system failure (failover and restoration), etc.
  - F/W Update
  - Configuration Parameter Update


Mesh Network of Smart Meters
Mesh Network of Smart Meters
Mesh Network of Smart Meters
Collector 1
Collector 2
Collector 3
Control Center

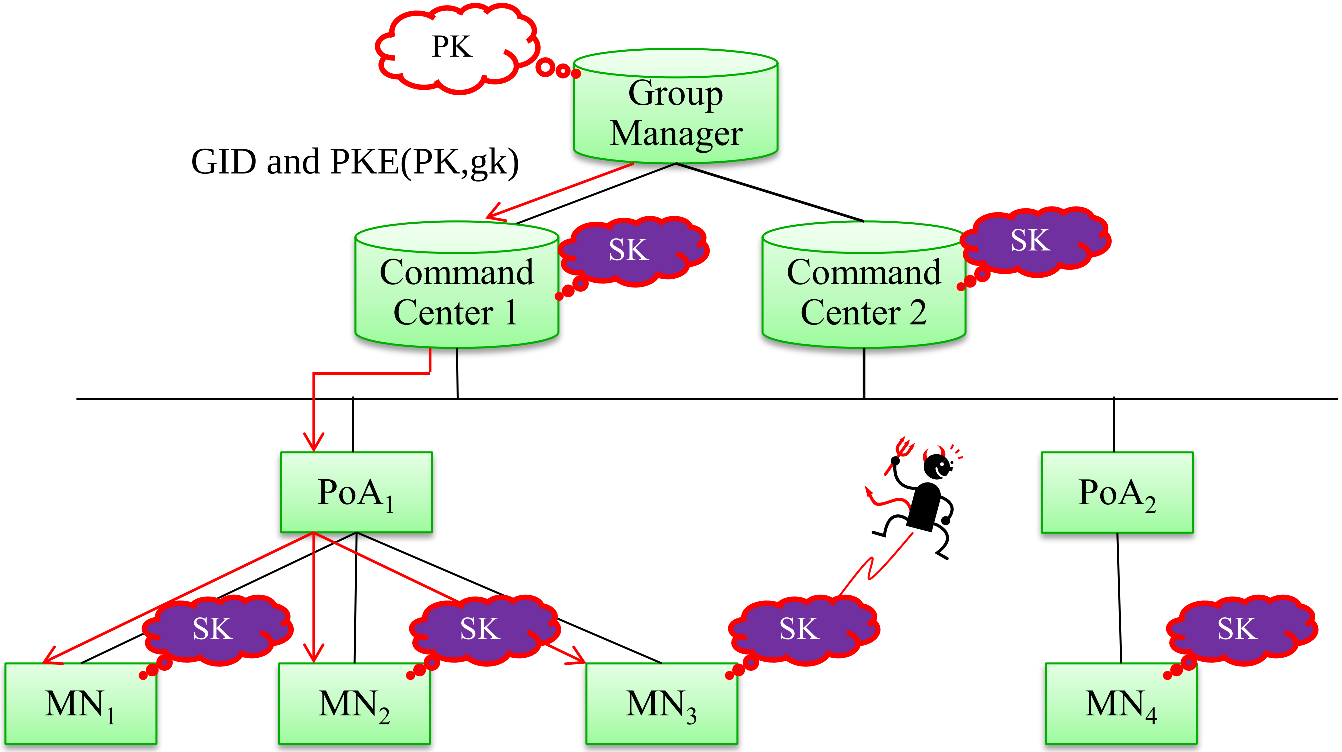
# Solution 1: Unique Keys



- **Advantage**: Resilience to leak of a unique key (No damage to the other MNs).
- **Disadvantage**: Group Manager need to manage a large number of keys. Group Manager should send a large size of data.

PKE: Public Key Encryption

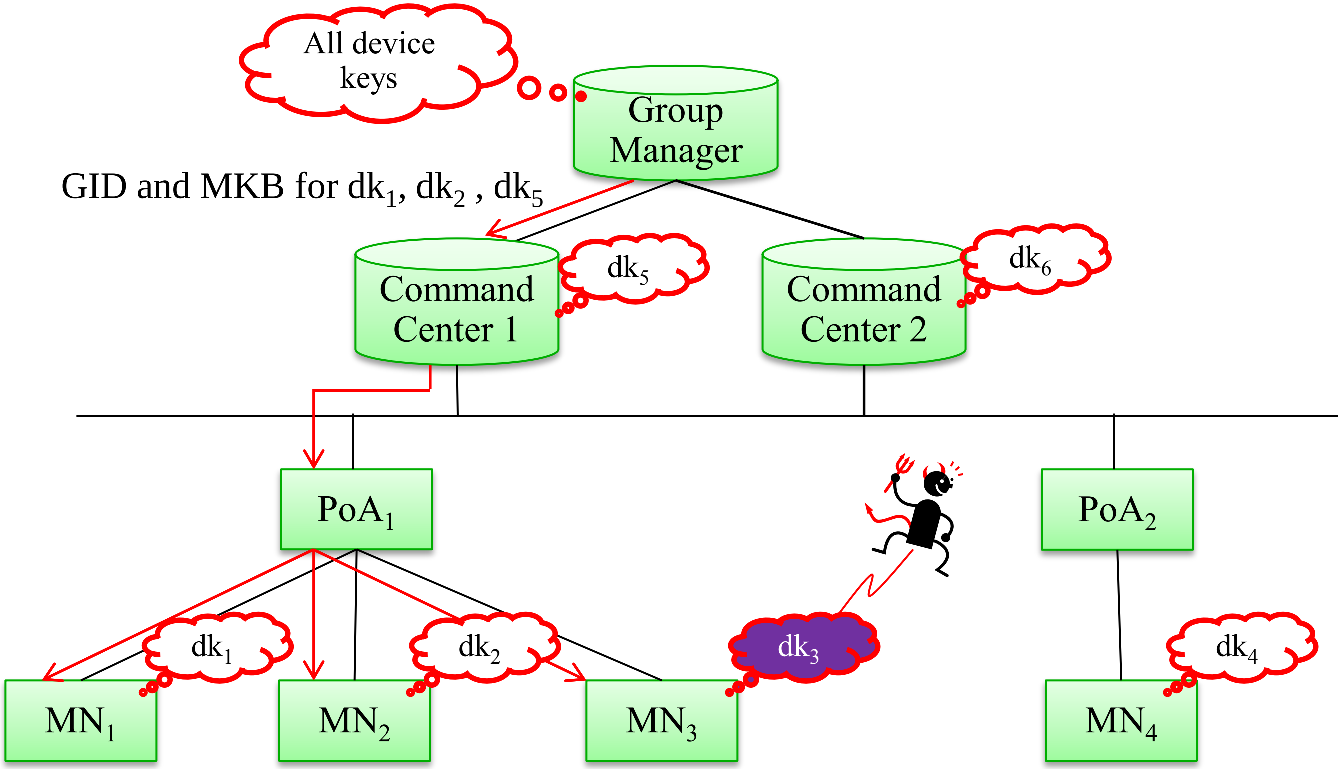
# Solution 2: A Group Key



- **Advantage:** Only one public key pair is necessary. Very easy to manage.
- **Disadvantage:** No resilience to key leak at an MN: No way to securely update the group key once one of the MNs is compromised.

PKE: Public Key Encryption

# Solution 3: MKB



- **Advantage**:
  - Group Manager sends a smaller size of data than Solution 1.
  - Resilience to leak of a long-term key: There exist a way to update a group key excluding the compromised MNs if they are detected.
- **Disadvantage**: The number of long-term keys is larger than a unique key or a group key approaches. (Each device key, which is the long-term keys, consists of plural symmetric keys.)

# Quantitative Comparison

| | Number of Key (GM) | Total Key size (GM) | Number of Key (MN) | Total Key size (MN) | Group Key Data size (GM) |
|---|---|---|---|---|---|
| Unique Keys (m = $2^{32}$) | $2^{32}$ | 239 GB | 1 | 28 B | 360 GB |
| Unique Keys (m = $2^{20}$) | $2^{20}$ | 59 MB | 1 | 28 B | 88 MB |
| Unique Keys (m = $2^{16}$) | $2^{16}$ | 3.6 MB | 1 | 28 B | 5.5 MB |
| MKB (m=$2^{32}$) | $2^{33}$-1 | 137 GB | 32 | 512 B | 34GB |
| MKB (m = $2^{20}$) | $2^{21}$-1 | 34 MB | 20 | 200 B | 8.3 MB |
| MKB (m=$2^{16}$) | $2^{17}$-1 | 2 MB | 16 | 160 B | 528KB |
| A Group Key | 1 | 56 B | 1 | 28 B | 84 B |

- Number of key (X): The number of keys which X holds.
- Total Key size (X): The total size of keys which X holds.
- Group Key Data size (GM): The maximum size of data which GM need to send to distribute a group key. The worst case.
- The group key approach has no resilience to key leakage.

- m: The number of the concerned MNs.
- Public key encryption: ECIES (112-bit security)
- Symmetric key encryption: AES-128 (128-bit security)
- MKB: Complete Subtree method using AES-128
- GM: Group Manager, MN: Mobile Node

- $2^{32}$ ≈43 billion, $2^{20}$ ≈1 million, $2^{16}$ ≈65 thousand

# Our Proposal

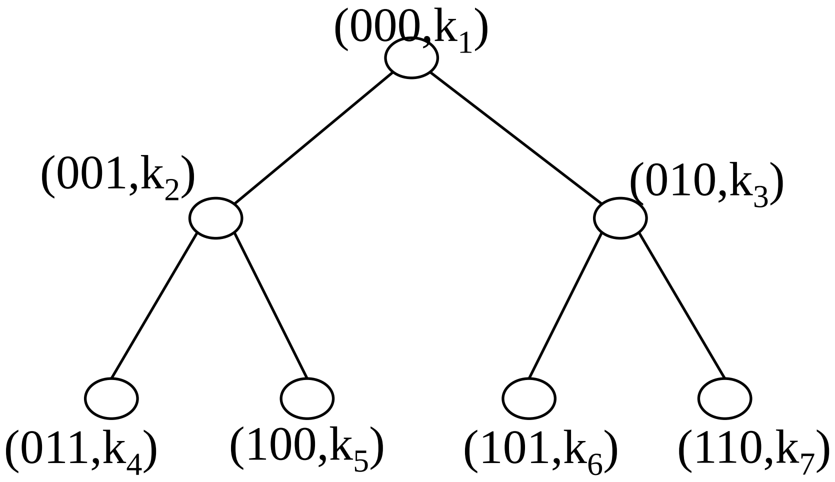
- We propose to make use of an MKB technology to deliver a group key (gk) to a target group of MNs via a multicast/broadcast channel.
- The group key (gk) is distributed in an encrypted manner: Only the MNs that belong to the target group can decrypt it.
- We will introduce some examples to show the basics of MKB.

# Toy example of MKB (1/3)

**Complete Subtree method**

The number of controlled MNs is 4.

1. GM generates a binary tree which has 4 leaves.
2. GM generates 7 device keys and 7 device key IDs, and virtually assigns them to the nodes of the binary tree.
3. The device key for a leaf node is defined as the set of the IDs and the keys which are picked up along the path from the leaf node ascending to the root.
4. An MN is assigned to a leaf node of the tree, and the long-term key for the MN is the device key defined as above.



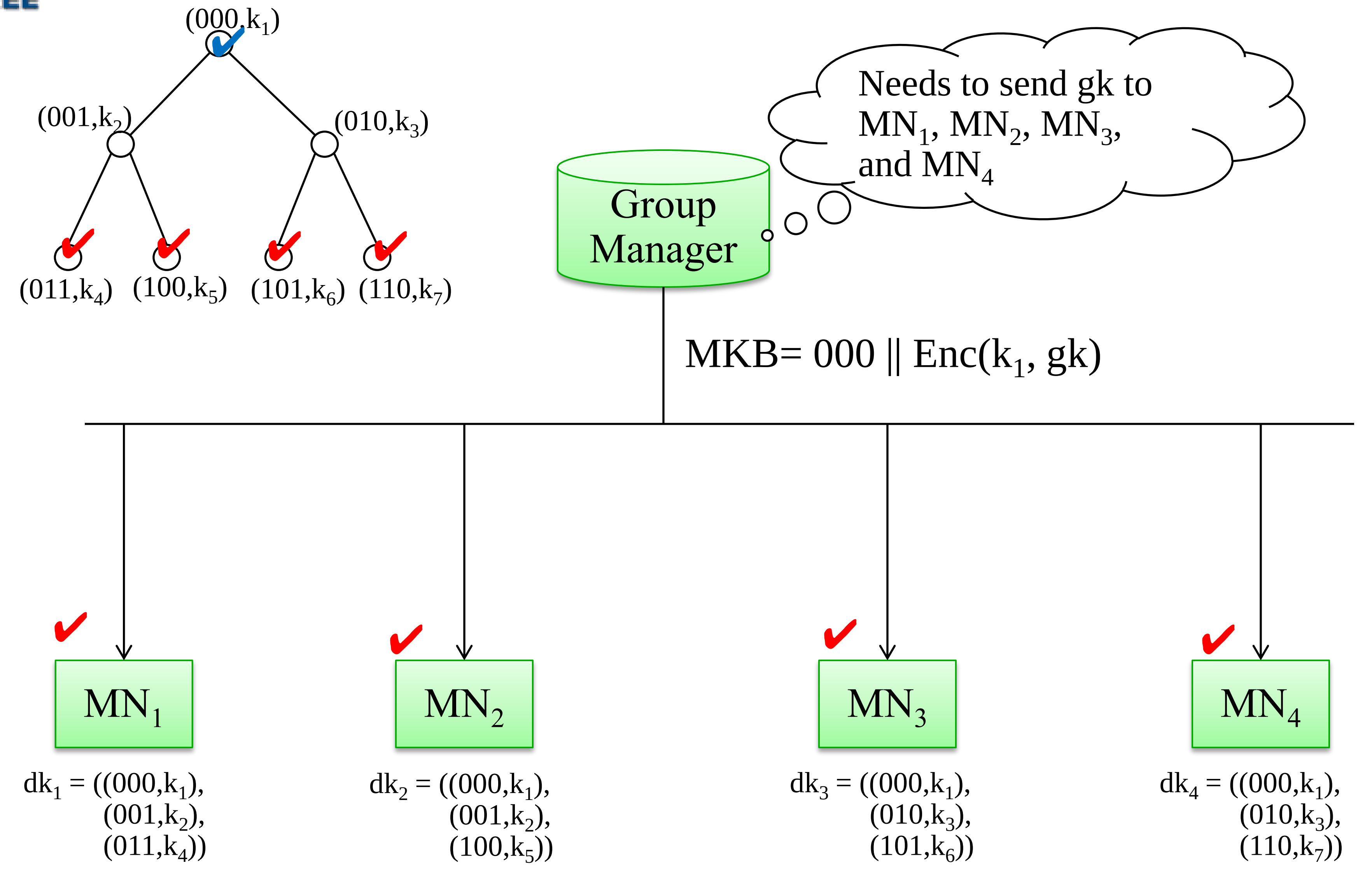
$dk_1 = ((000,k_1), (001,k_2), (011,k_4))$

$dk_2 = ((000,k_1), (001,k_2), (100,k_5))$

$dk_3 = ((000,k_1), (010,k_3), (101,k_6))$

$dk_4 = ((000,k_1), (010,k_3), (110,k_7))$

# Toy example of MKB (2/3)



Needs to send gk to $MN_1$, $MN_2$, $MN_3$, and $MN_4$

Group Manager

MKB= 000 || Enc($k_1$, gk)

$MN_1$

$MN_2$

$MN_3$

$MN_4$

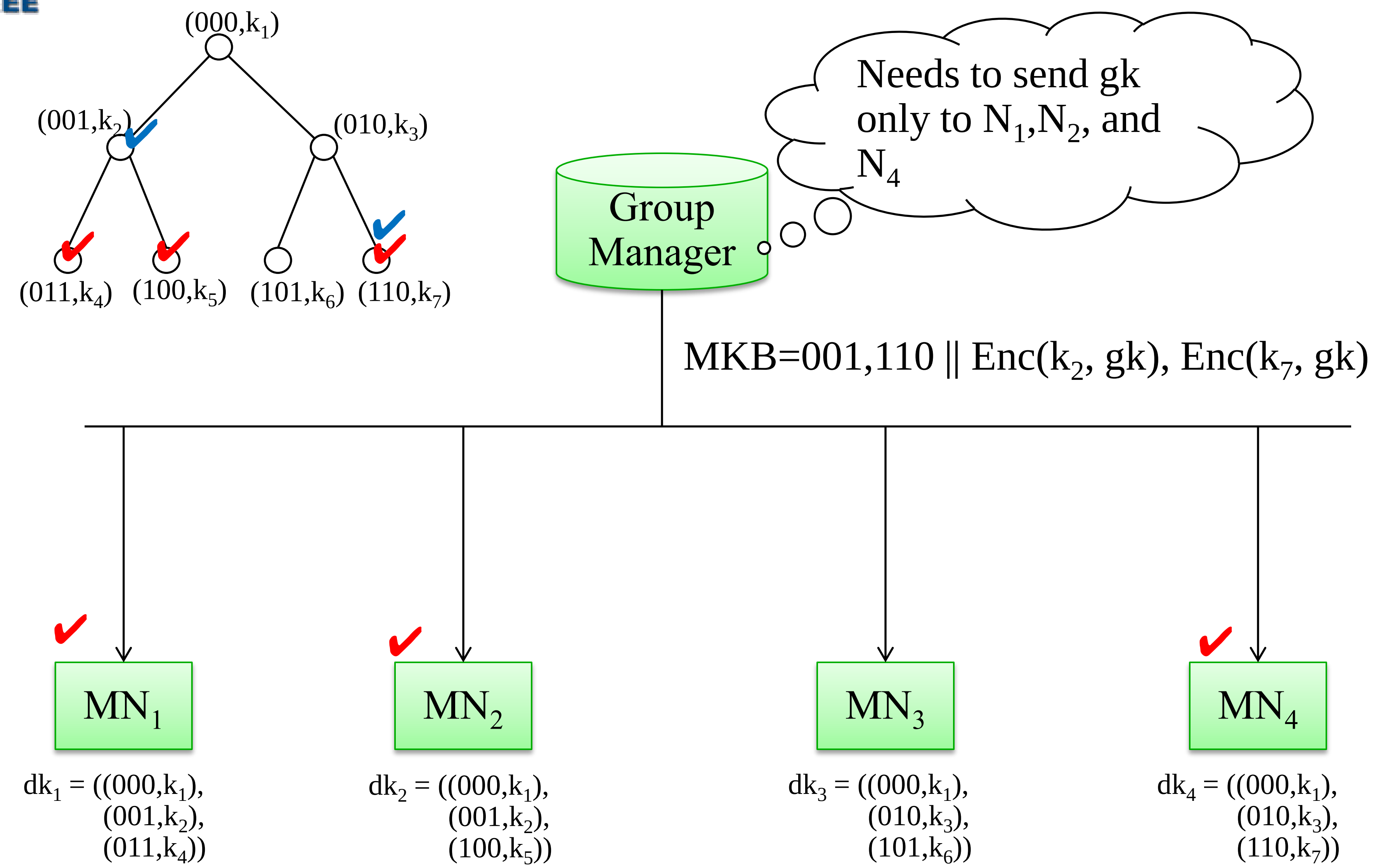
$dk_1$ = ((000,$k_1$), (001,$k_2$), (011,$k_4$))

$dk_2$ = ((000,$k_1$), (001,$k_2$), (100,$k_5$))

$dk_3$ = ((000,$k_1$), (010,$k_3$), (101,$k_6$))

$dk_4$ = ((000,$k_1$), (010,$k_3$), (110,$k_7$))

# Toy example of MKB (3/3)



(000,$k_1$)

(001,$k_2$) (010,$k_3$)

(011,$k_4$) (100,$k_5$) (101,$k_6$) (110,$k_7$)

Group Manager

Needs to send gk only to $N_1$,$N_2$, and $N_4$

MKB=001,110 || Enc($k_2$, gk), Enc($k_7$, gk)

$MN_1$ $MN_2$ $MN_3$ $MN_4$

$dk_1$ = ((000,$k_1$),<br>(001,$k_2$),<br>(011,$k_4$))

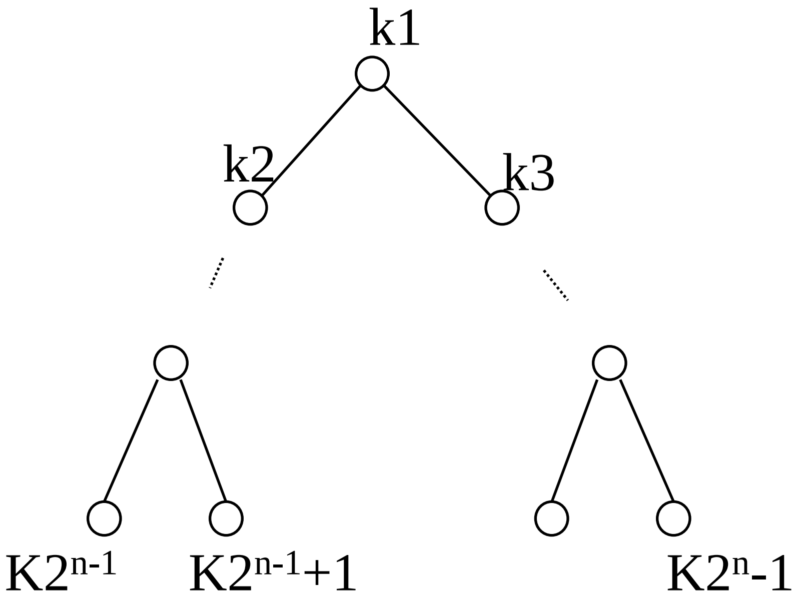
$dk_2$ = ((000,$k_1$),<br>(001,$k_2$),<br>(100,$k_5$))

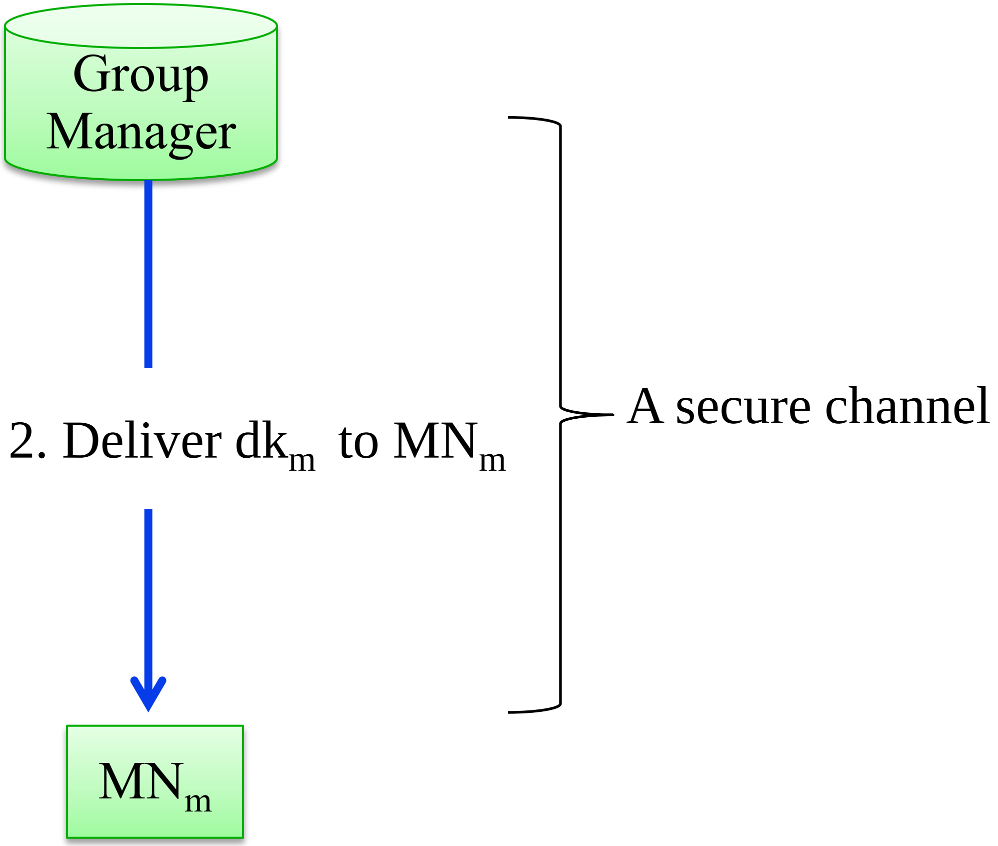
$dk_3$ = ((000,$k_1$),<br>(010,$k_3$),<br>(101,$k_6$))

$dk_4$ = ((000,$k_1$),<br>(010,$k_3$),<br>(110,$k_7$))

# Setup of Device keys

1. Generate a device key ($dk_m$) for $MN_m$.



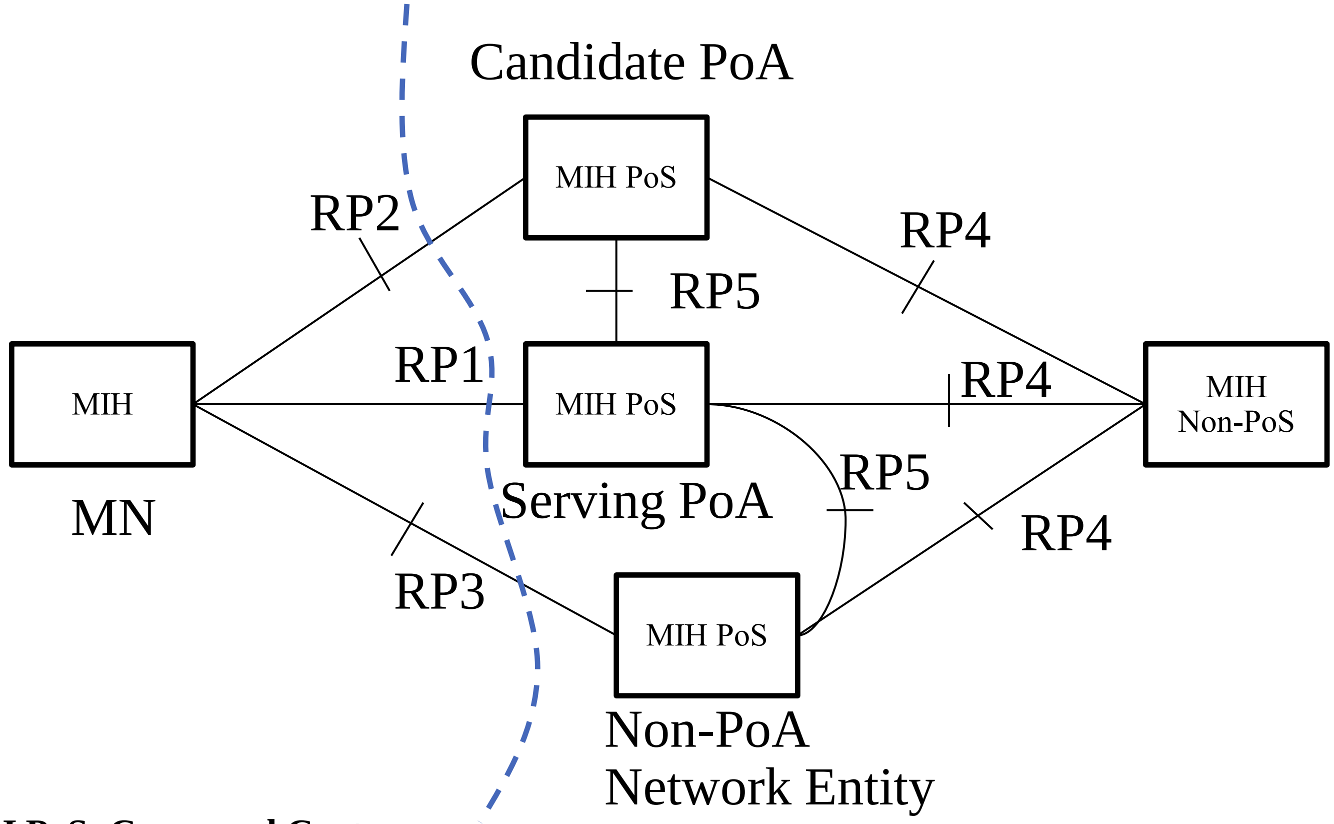


2. Deliver $dk_m$ to $MN_m$

3. $MN_m$ installs $dk_m$ as its device key.

# Mapping to MIH framework

MIHF Communication Model
(same as Figure 2 of IEEE 802.21-2008)



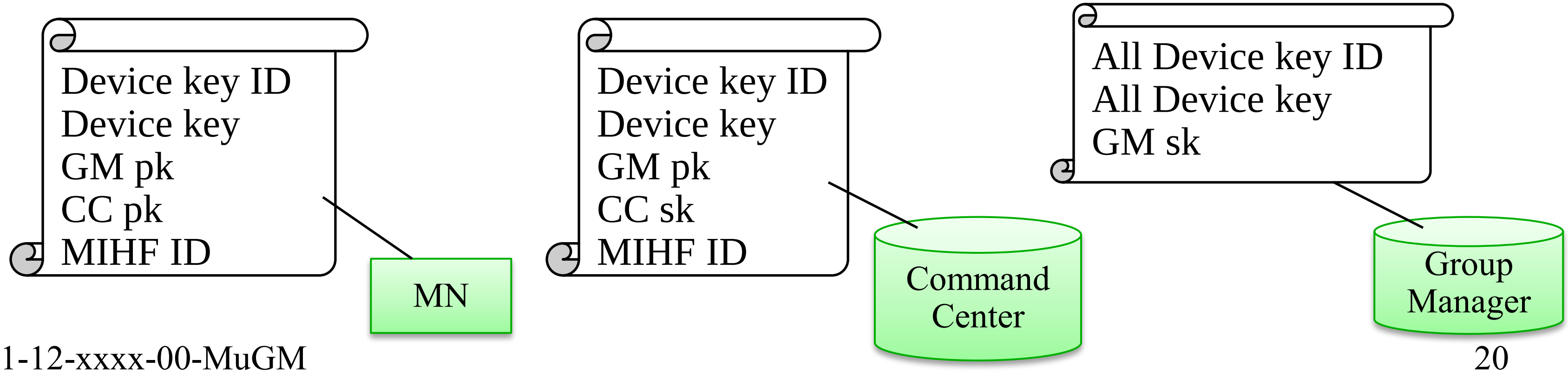
- **MIH PoS: Command Center**
- **Key Server (not shown above) is connected to each PoS using an interface (e.g., AAA) defined outside the scope of 802.21**
- **Multicast sender: MIH PoS, Multicast receiver: MN, MIH PoS**

# Outline

- Introduction
- Definitions
- Architecture and Concept
  - System model
  - Concept of our solution
  - MKB Basics
  - Mapping to MIH framework
- Prerequisites for System
- Proposal
  - Group ID
  - Group manipulation
  - Group command
  - MIH Primitives
  - MIH Messages
  - Group key hierarchy
- Conformance to the requirements
- Summary

# Prerequisites for System

- Device key IDs and Device keys are pre-installed.
- A Command Center (CC)'s secret key to generate a digital signature is pre-installed in the PoS.
- A Group Manager (GM)'s secret key to generate a digital signature is pre-installed in the GM.
- The GM's public key and the CC's public key to verify digital signatures are pre-installed in all the PoSs and all the MNs.
- The Individual MIHF IDs are pre-installed in all the PoSs and all the MNs.
- The unicast security convention follows IEEE 802.21a.
- Mapping between a Group ID and a multicast transport address is managed and maintain within MIH_NET_SA.
- Underlying multicast transport is used to carry MIH messages related to the group manipulation commands and the group commands.

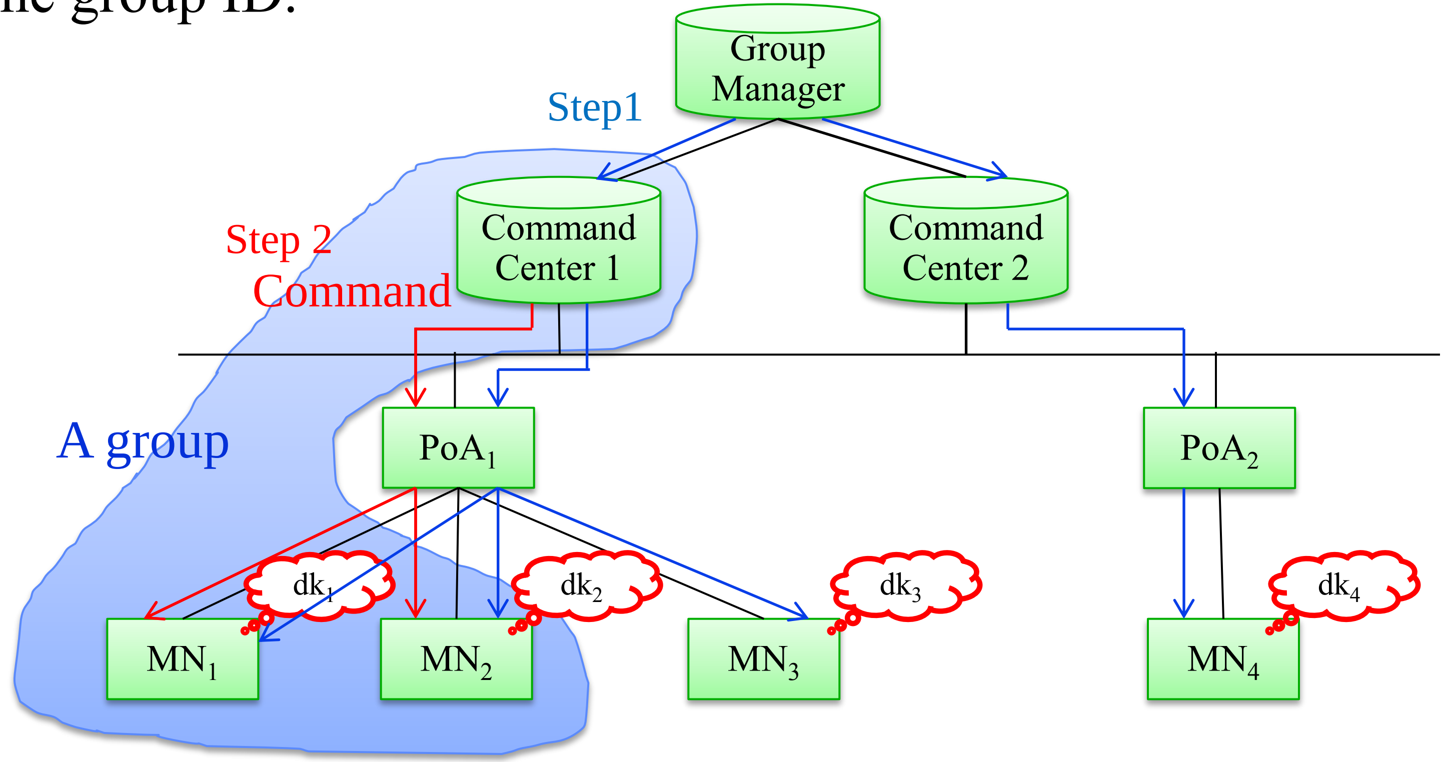


# Outline

- Introduction
- Definitions
- Architecture and Concept
  - System model
  - Concept of our solution
  - MKB Basics
  - Mapping to MIH framework
- Prerequisites for System
- Proposal
  - Group ID
  - Group manipulation
  - Group command
  - MIH Primitives
  - MIH Messages
  - Group key hierarchy
- Conformance to the requirements
- Summary

# Two step approach

- Step 1: Group Manager manipulates a group by distributing a group ID and  a group key using an MKB via multicast channels.
  - The group is a multicast group, and the group ID represents the multicast group.
- Step 2: Command Center issues group commands to a group designated by the group ID.

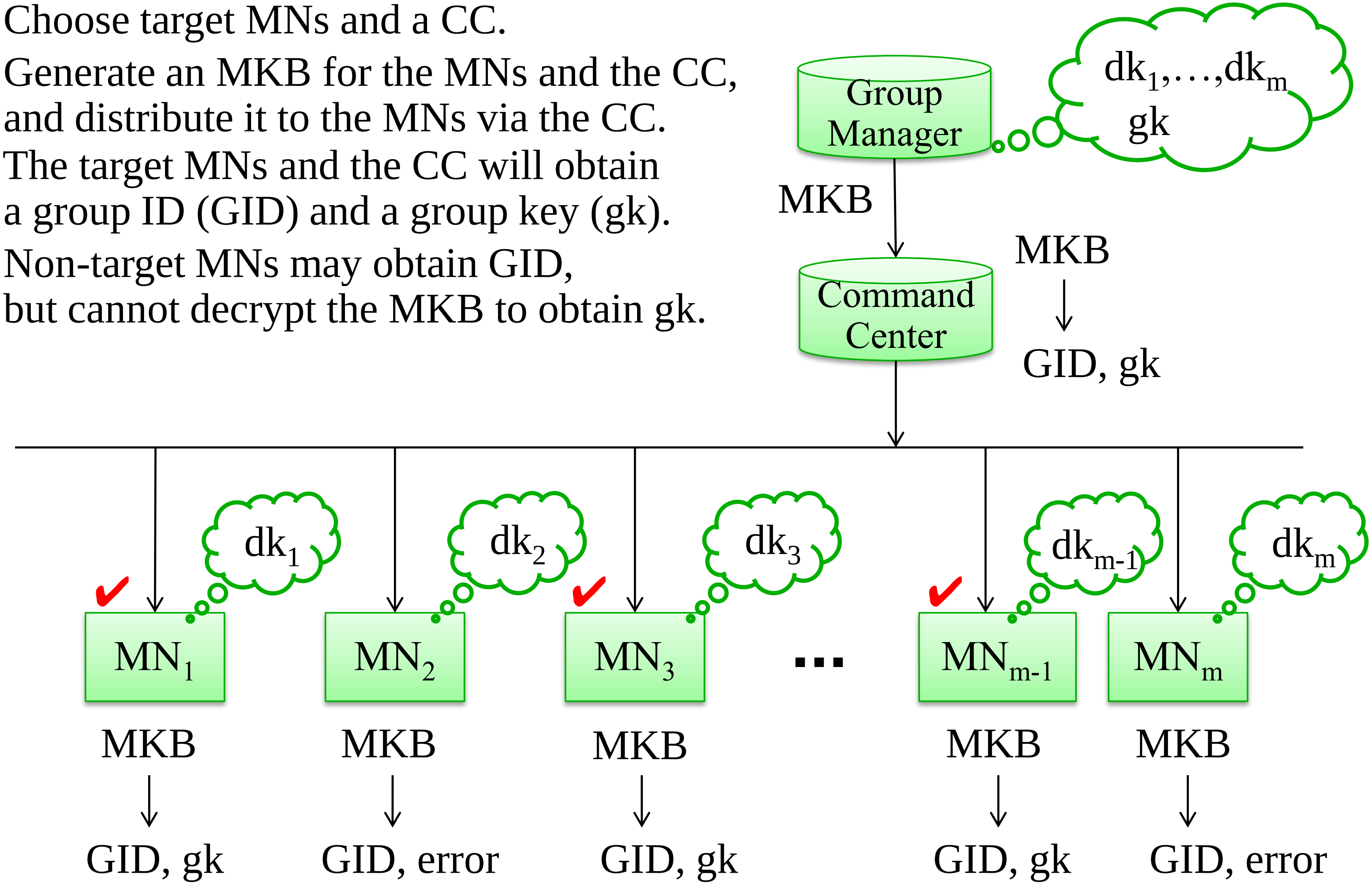


# Group ID

- Use MIHF ID as Group ID
  - Alternative 1: We do not modify the current MIHF ID rule, but newly define an out-of-band mechanism to distinguish a Group MIHF ID from Individual MIHF IDs.
    - Ex. XXXX@YYYY, and  an attribute which says "this is a Group MIHF ID".
  - Alternative 2: We restrict the current MIHF ID rule to distinguish a Group MIHF ID from Individual MIHF IDs.
    - Ex. Group-XXXX@YYYY i.e., "Group-" is reserved word.
  - Which alternative is suitable?

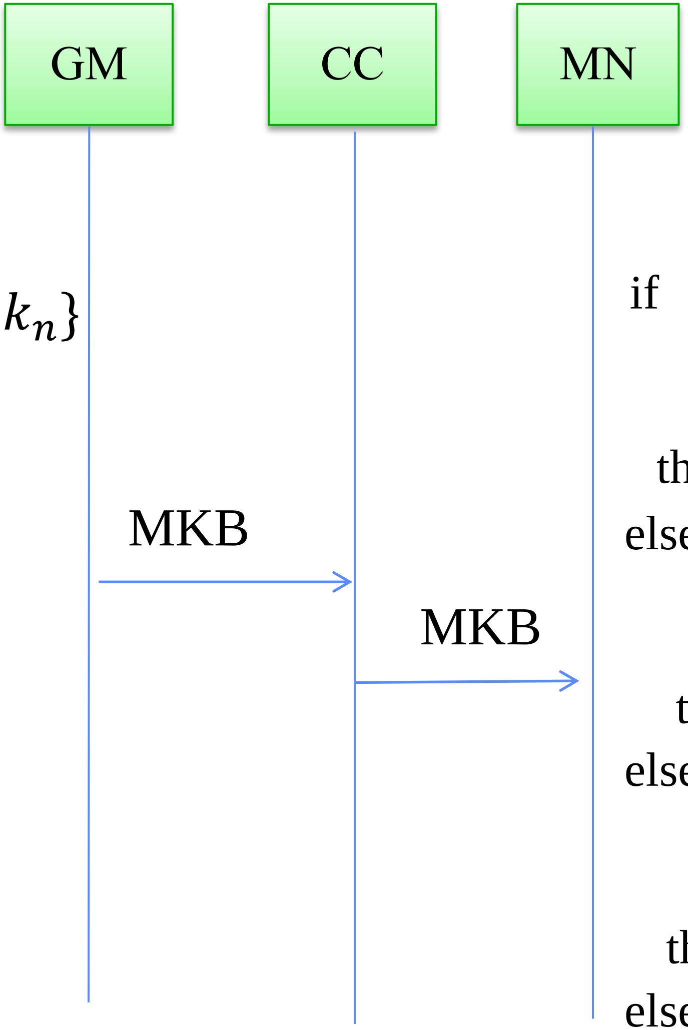
# Step1 Group manipulation : Overview

1. Choose target MNs and a CC.
2. Generate an MKB for the MNs and the CC, and distribute it to the MNs via the CC.
3. The target MNs and the CC will obtain a group ID (GID) and a group key (gk).
4. Non-target MNs may obtain GID, but cannot decrypt the MKB to obtain gk.



# Group manipulation procedure

$(pk, sk)$
G-List
$(dk_1, \dots, dk_n)$



GM

CC

MN

MKB

MKB

$pk$
G’-List
$dk_i$

Choose $DK \subseteq \{dk_1, \dots, dk_n\}$

$G_j$: Group ID

$gk$: Group Key

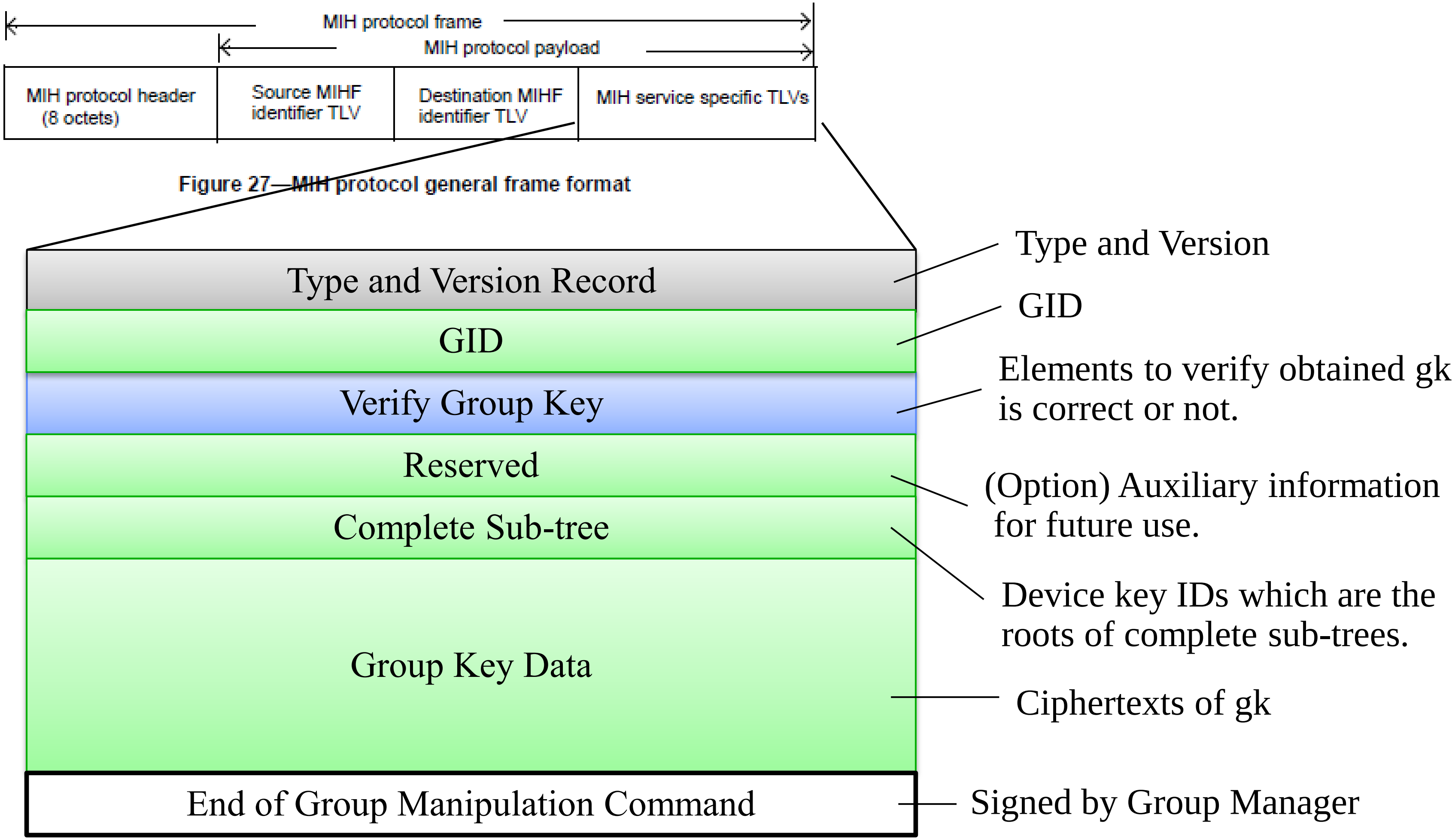
Generate an MKB using $DK, G_j, gk$ and $sk$.

Update G-List

if $(G_j,*) \in G'$-List∧
$Verify(pk, MKB) = 1 \wedge$
$Dec_{MKB}(dk_i, MKB) = gk$
then $Update\ groupky\ of\ G_j$
else if $(G_j,*) \notin G'$-List∧
$Verify(pk, MKB) = 1 \wedge$
$Dec_{MKB}(dk_i, MKB) = gk$
then $Join\ group\ G_j$
else if $(G_j,*) \in G'$-List∧
$Verify(pk, MKB) = 1 \wedge$
$Dec_{MKB}(dk_i, MKB) = \perp$
then $Leave\ group\ G_j$
else Exit.

# MIH service specific TLVs for Group Manipulation

Group manipulation belongs to the service management service of MIH.



MIH protocol frame

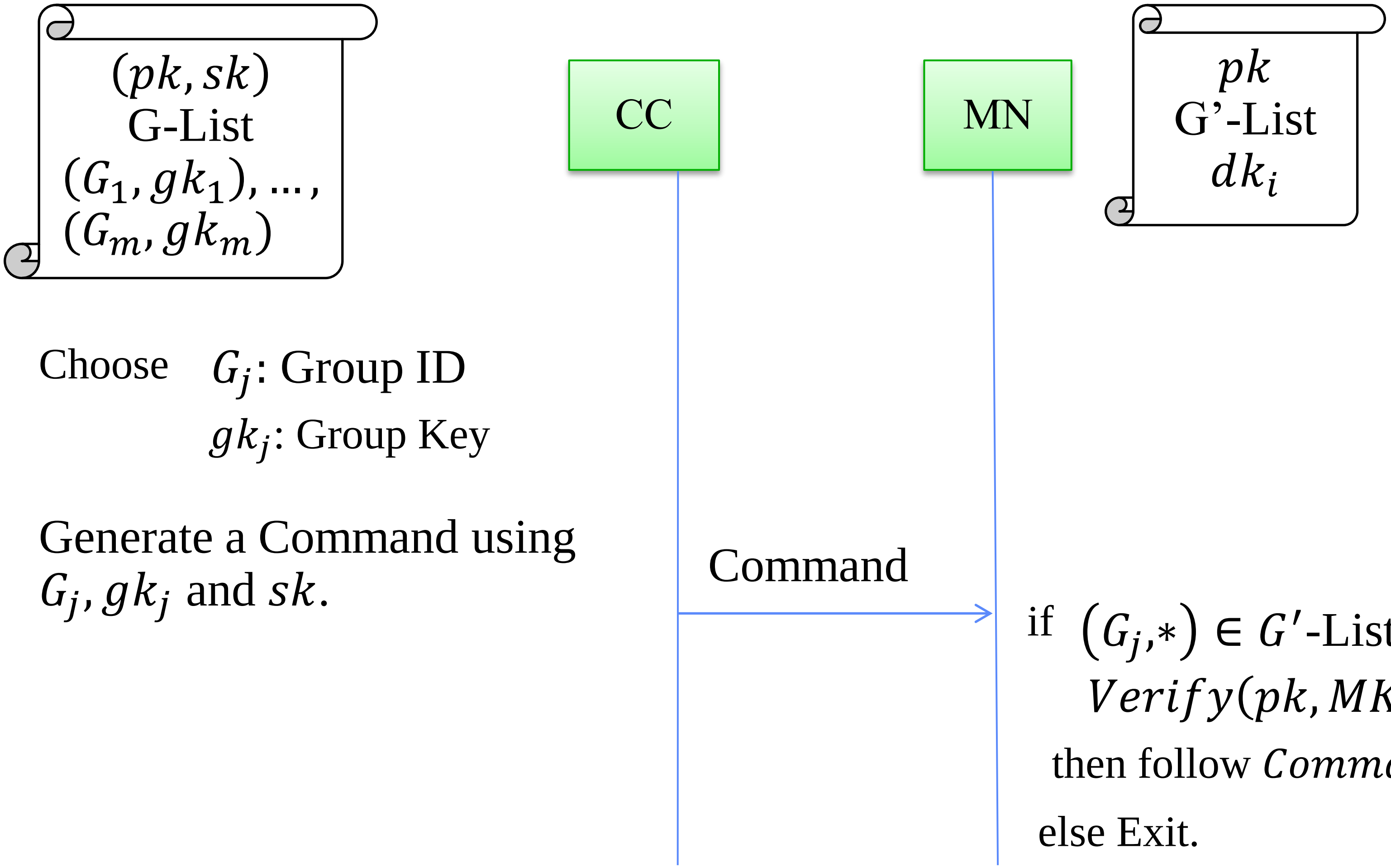
MIH protocol payload

| | | | |
|---|---|---|---|
| MIH protocol header (8 octets) | Source MIHF identifier TLV | Destination MIHF identifier TLV | MIH service specific TLVs |

Figure 27—MIH protocol general frame format

| | |
|---|---|
| Type and Version Record | Type and Version |
| GID | GID |
| Verify Group Key | Elements to verify obtained gk is correct or not. |
| Reserved | (Option) Auxiliary information for future use. |
| Complete Sub-tree | Device key IDs which are the roots of complete sub-trees. |
| Group Key Data | Ciphertexts of gk |
| End of Group Manipulation Command | Signed by Group Manager |

# Step2 Group command procedure



$(pk, sk)$
G-List
$(G_1, gk_1), \dots,$
$(G_m, gk_m)$

CC

MN

$pk$
G’-List
$dk_i$

Choose $G_j$: Group ID
$gk_j$: Group Key

Generate a Command using $G_j, gk_j$ and $sk$.

Command

if $(G_j,*) \in G'$-List$\wedge$
$Verify(pk, MKB) = 1 \wedge$
then follow $Command$
else Exit.

Group command procedure belongs to the command service of MIH.

Failover/Restoration

Load Balancing

Configuration Parameters Update

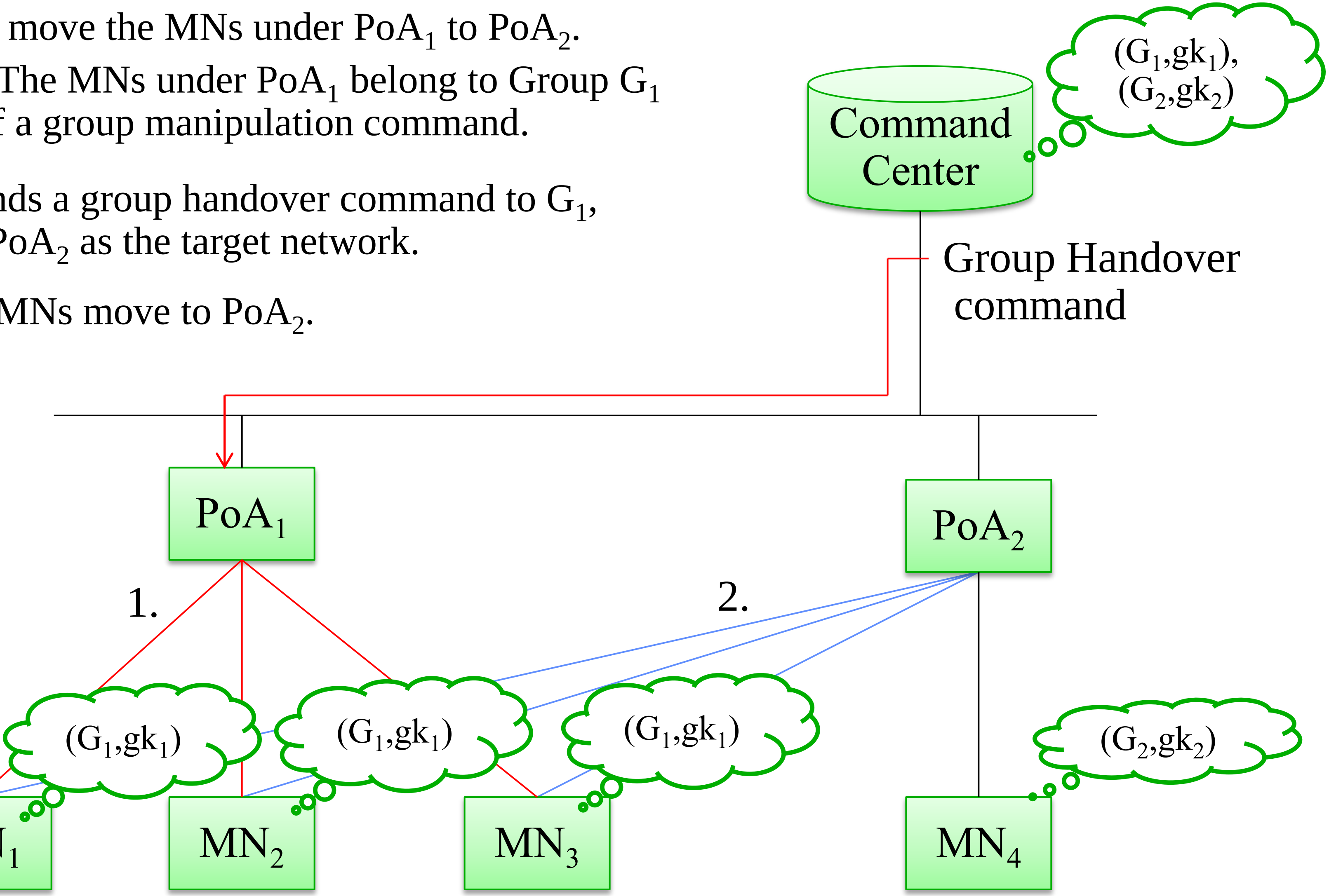
F/W Update

# Failover/Restoration (1/2)

The CC need move the MNs under $PoA_1$ to $PoA_2$.

Assumption: The MNs under $PoA_1$ belong to Group $G_1$ as a result of a group manipulation command.

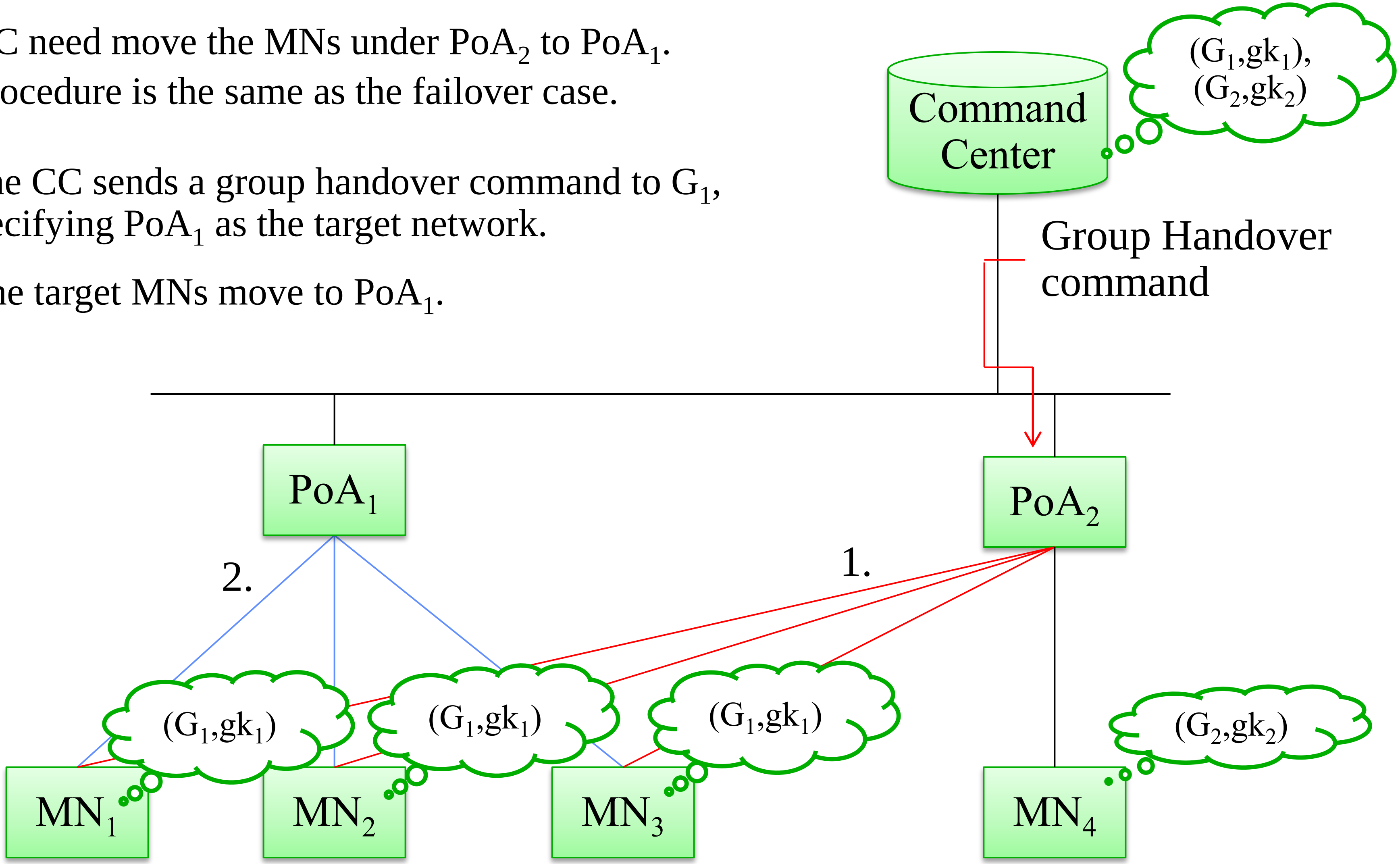
1. The CC sends a group handover command to $G_1$, specifying $PoA_2$ as the target network.
2. The target MNs move to $PoA_2$.



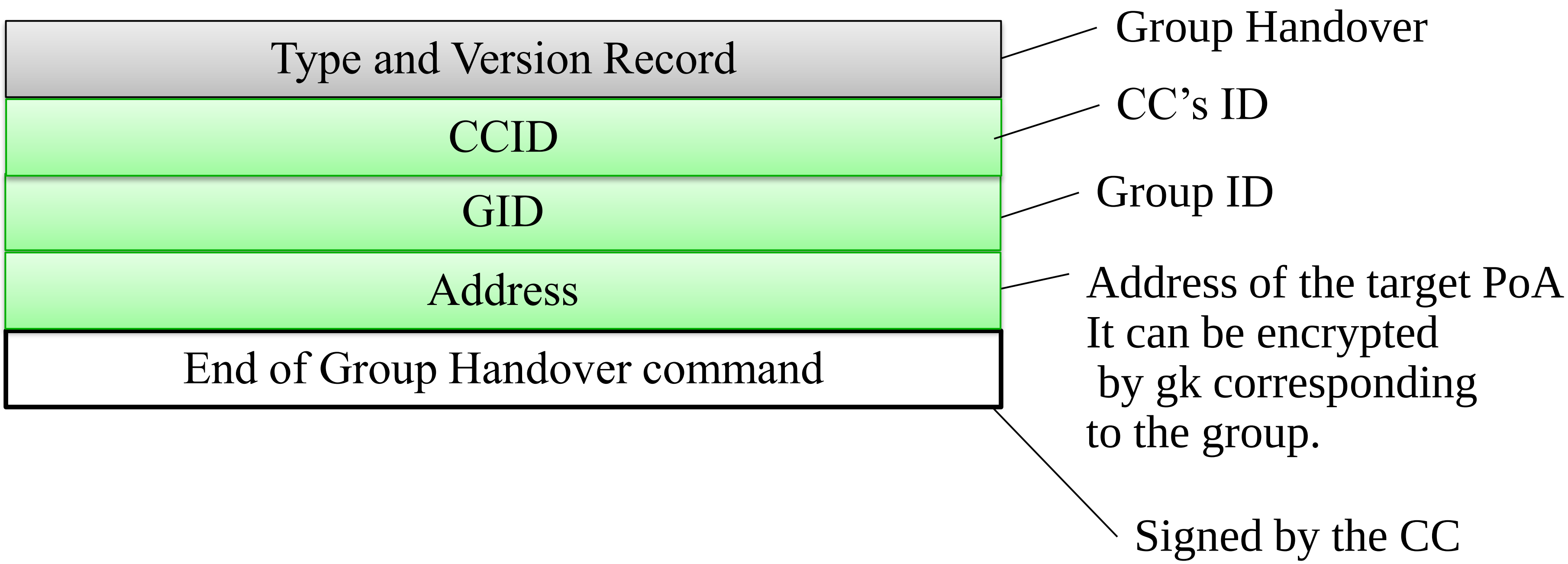
# Failover/Restoration (2/2)

The CC need move the MNs under $PoA_2$ to $PoA_1$.
The procedure is the same as the failover case.

1. The CC sends a group handover command to $G_1$, specifying $PoA_1$ as the target network.
2. The target MNs move to $PoA_1$.



# Group Handover Command Format



Type and Version Record

CCID

GID

Address

End of Group Handover command

Group Handover

CC’s ID

Group ID

Address of the target PoA
It can be encrypted
by gk corresponding
to the group.

Signed by the CC

Load Balancing is also covered by commands of this format.

Failover/Restoration

Load Balancing

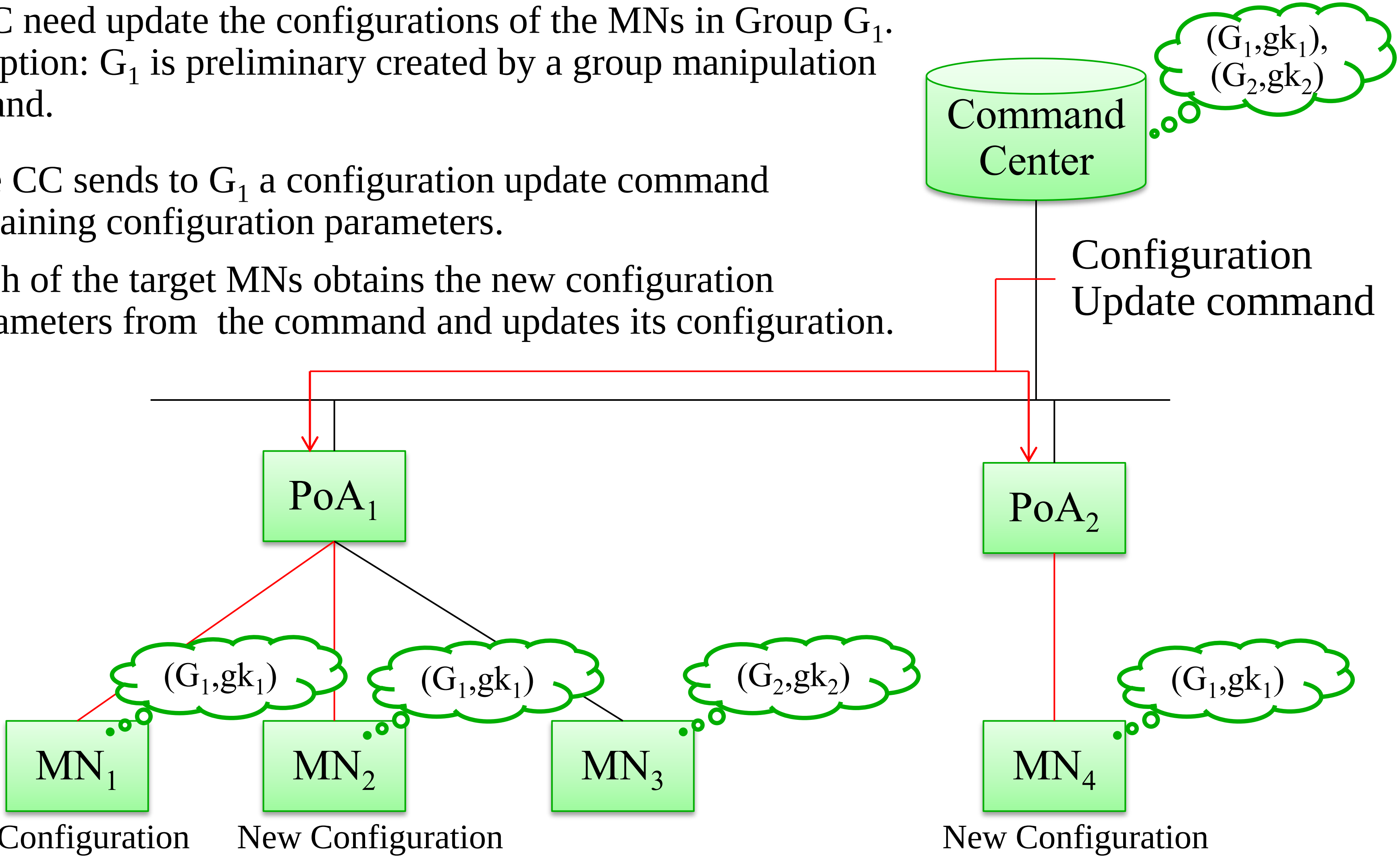
Configuration Parameters Update

F/W Update

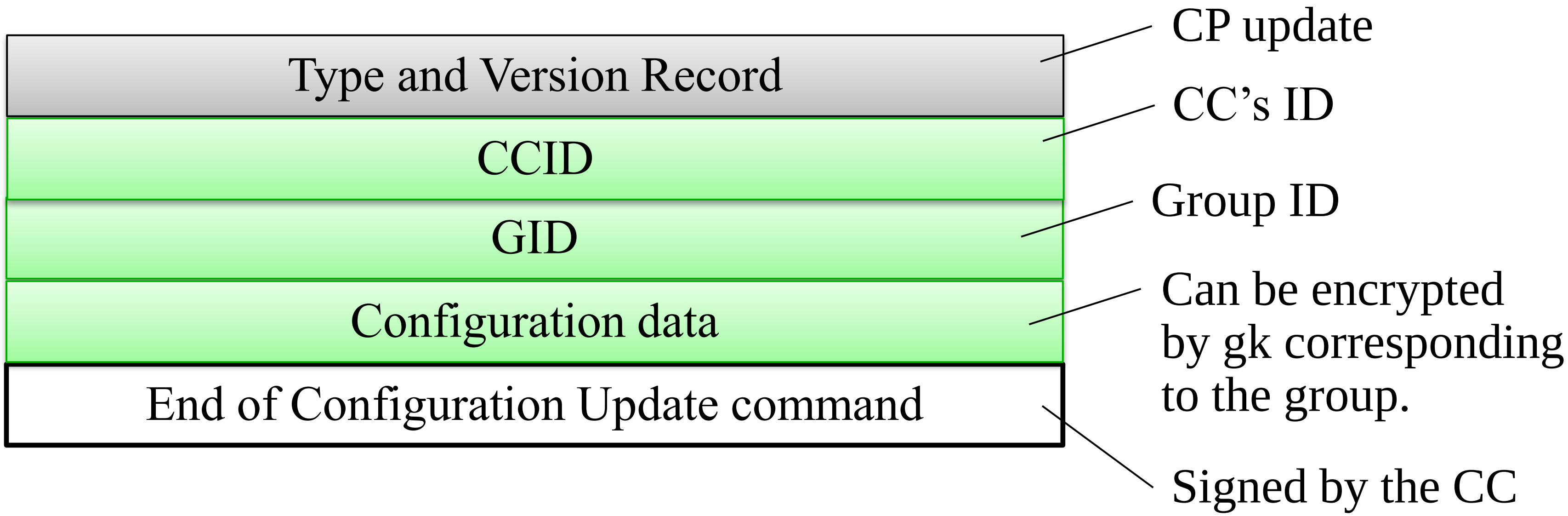
# Configuration Parameters Update

The CC need update the configurations of the MNs in Group $G_1$. Assumption: $G_1$ is preliminary created by a group manipulation command.

1. The CC sends to $G_1$ a configuration update command containing configuration parameters.
2. Each of the target MNs obtains the new configuration parameters from the command and updates its configuration.



# Configuration Update Command Format



Type and Version Record

CCID

GID

Configuration data

End of Configuration Update command

CP update

CC’s ID

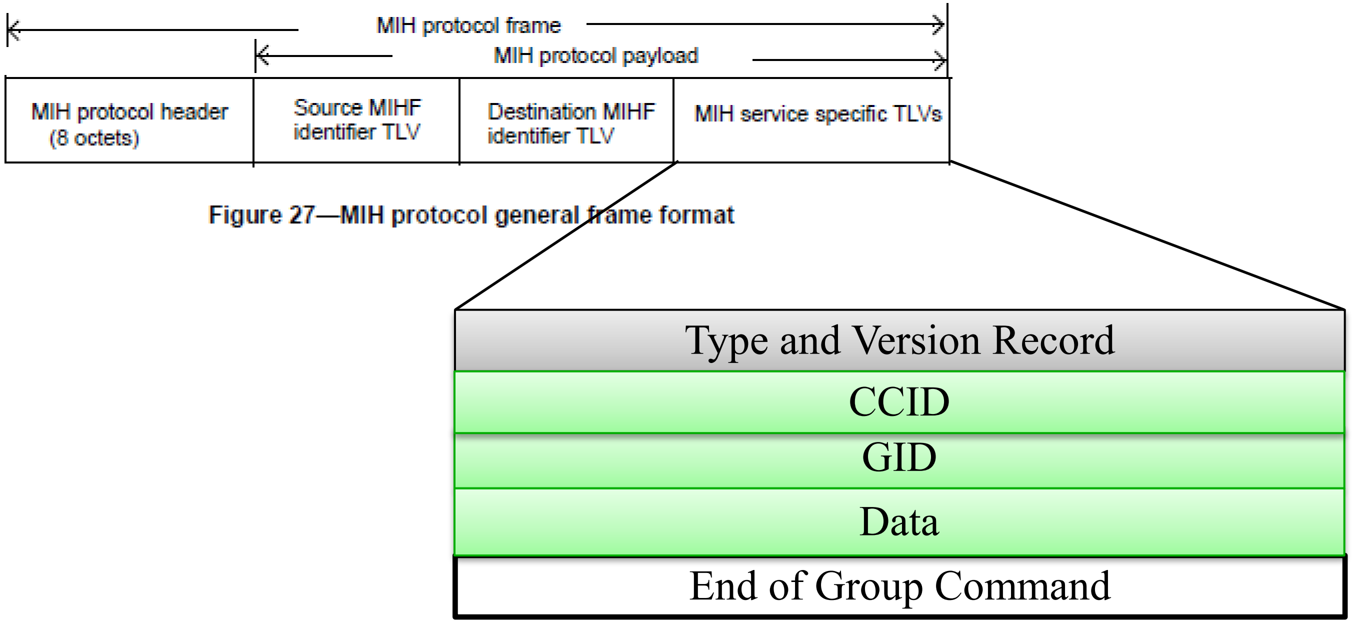
Group ID

Can be encrypted by gk corresponding to the group.

Signed by the CC

Firmware update is also covered by a command of this format.

# MIH service specific TLVs for Group Commands



MIH protocol frame

MIH protocol payload

MIH protocol header (8 octets)

Source MIHF identifier TLV

Destination MIHF identifier TLV

MIH service specific TLVs

Figure 27—MIH protocol general frame format

Type and Version Record

CCID

GID

Data

End of Group Command

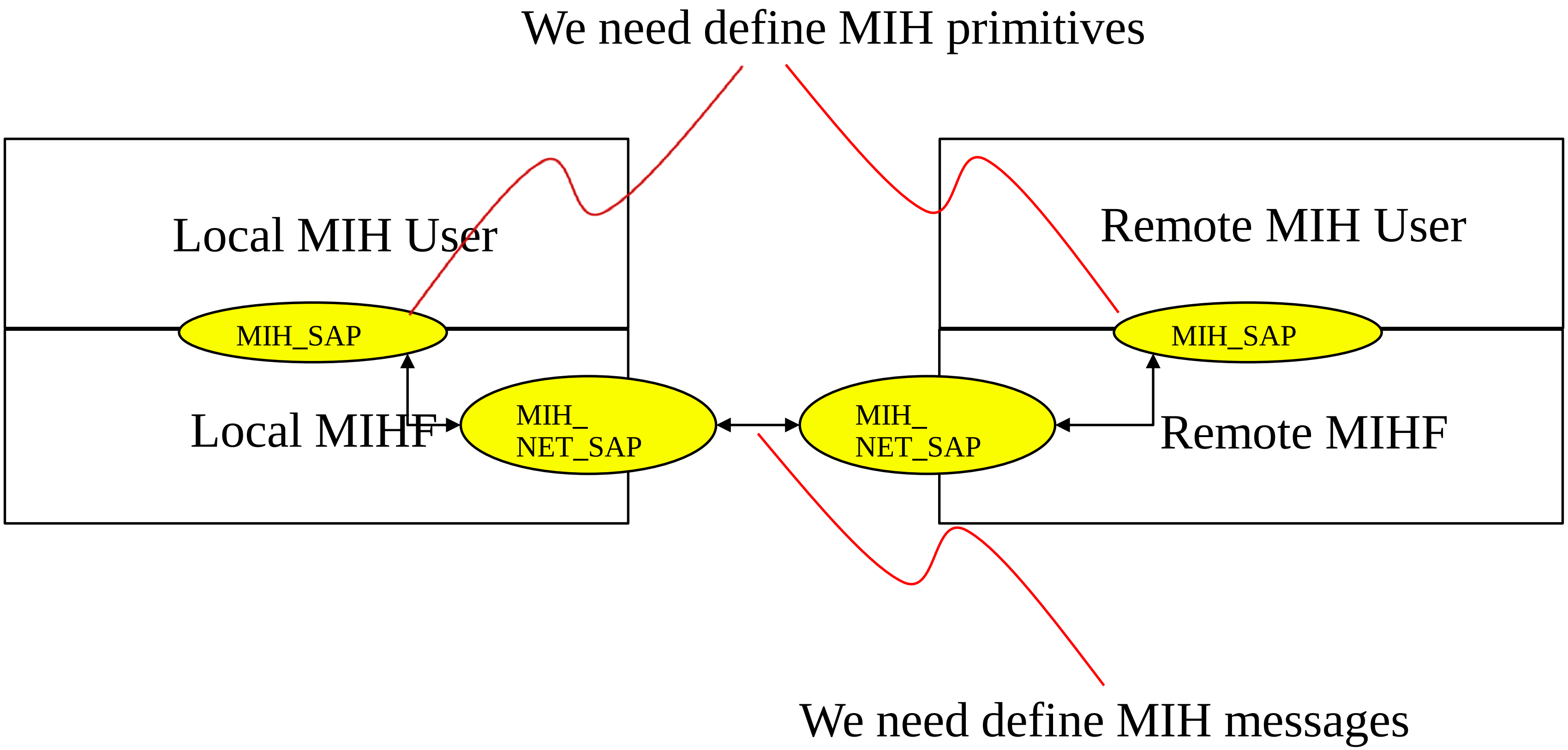
| | Type and Version | CCID | GID | Data | End of command |
|---|---|---|---|---|---|
| Group Handover | Group Handover | CCID | GID | Target PoA | Sig by CC |
| Configuration Update | Configuration Update | CCID | GID | (Encrypted) Configuration | Sig by CC |
| F/W Update | F/W Update | CCID | GID | (Encrypted) F/W | Sig by CC |

# Outline

- Introduction
- Definitions
- Architecture and Concept
    - System model
    - Concept of our solution
    - MKB Basics
    - Mapping to MIH framework
- Prerequisites for System
- Proposal
    - Group ID
    - Group manipulation
    - Group command
    - MIH Primitives and MIH Messages
    - Group key hierarchy
- Conformance to the requirements
- Summary

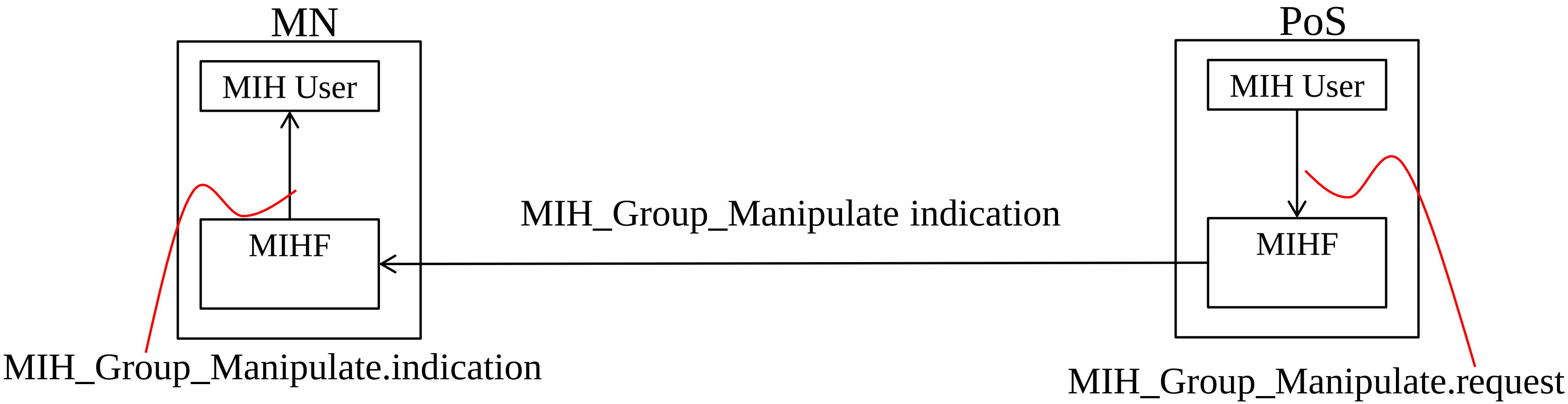
# MIH Primitives and Messages

- Local and Remote MIH Users exchange Group Manipulation Commands or Group Commands together with their Command attributes via MIH_SAP.
- Local and Remote MIHFs exchange the  attributes via MIH_NET_SAP using MIH protocol.



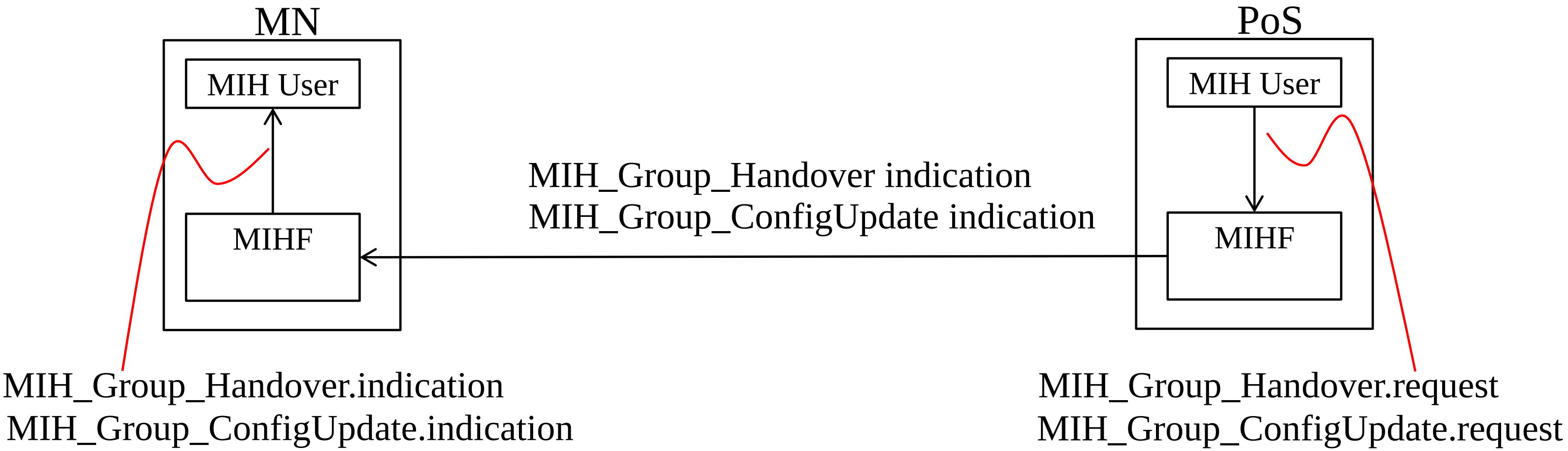
# MIH Primitives and Message for Group Manipulation

- Group manipulation based on unicast communication
  - We use MIH registration for sending a group ID and a group key so that an MN can join or leave a group.
- Group manipulation based on multicast communication
  - We define new primitives in the service management service.
    - MIH_Group_Manipulate.request,
    - MIH_Group_Manipulate.indication.
  - We define a new message.
    - MIH_Group_Manipulate indication


MN
MIH User
MIHF
PoS
MIH User
MIHF
MIH_Group_Manipulate indication
MIH_Group_Manipulate.indication
MIH_Group_Manipulate.request

# MIH Primitives and Message for Group Command

- Group command based on multicast communication
  - We define new primitives in the command service.
    - MIH_Group_Handover.request, MIH_Group_Handover.indication.
    - MIH_Group_ConfigUpdate.request, MIH_Group_ConfigUpdate.indication.
  - We define a new message.
    - MIH_Group_Handover indication
    - MIH_Group_ConfigUpdate indication

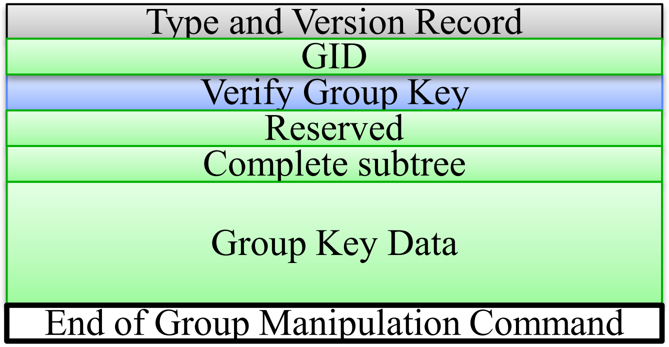

MN
MIH User
MIHF
PoS
MIH User
MIHF
MIH_Group_Handover indication
MIH_Group_ConfigUpdate indication
MIH_Group_Handover.indication
MIH_Group_ConfigUpdate.indication
MIH_Group_Handover.request
MIH_Group_ConfigUpdate.request

# Summary of MIH Primitives

- Group manipulation based on unicast communication:
  - We use MIH registration to send a group ID and a group key so that an MN can join or leave a group.
  - Parameters corresponding to MIH service specific TLVs for Group Manipulation should be added to MIH_Register.request and MIH_Register.indication primitives.
- Group manipulation based on multicast communication:
  - We should define new primitives in the service management service.
    - MIH_Group_Manipulate.request, MIH_Group_Manipulate.indication.
- Group command based on multicast communication:
  - We should define new primitives in the command service.
    - MIH_Group_Handover.request, MIH_Group_Handover.indication.
    - MIH_Group_ConfigUpdate.request, MIH_Group_ConfigUpdate.indication

# Summary of MIH Messages

- We modify MIH registration.
- We define MIH_Group_Manipulate indication message containing following TLV format.



Type and Version Record

GID

Verify Group Key

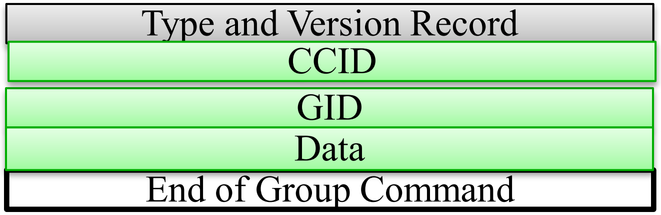
Reserved

Complete subtree

Group Key Data

End of Group Manipulation Command

- We define “MIH_Group_Handover indication” and “MIH_Group_Update indication” containing following TLV format.



Type and Version Record

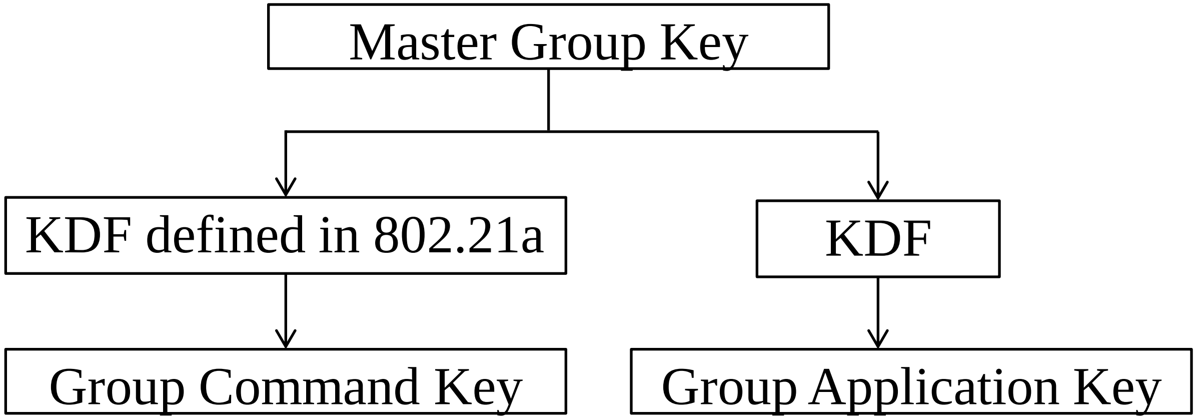
CCID

GID

Data

End of Group Command

# Group key hierarchy



A key to encrypt command arguments (Group Command Key) is not in fact a group key. The key is derived from the group key (Master Group Key).

- Master group key is distributed by Group Manager using MKB.
- A Group Command Key is derived from the Master Group Key using the KDF defined in 802.21a.
- A Group Application Key, which is an application-specific key, is derived from the Master Group Key using a KDF which is application-specific.

# Outline

- Introduction
- Definitions
- Architecture and Concept
    - System model
    - Concept of our solution
    - MKB Basics
    - Mapping to MIH framework
- Prerequisites for System
- Proposal
    - Group ID
    - Group manipulation
    - Group command
    - MIH Primitives
    - MIH Messages
    - Group key hierarchy
- Conformance to the requirements
- Summary

# Conformance of the requirements

| Supported Functionality | Requirement # in TR document | Note |
|---|---|---|
| Multicast Communication | Req2.1.1.1 | Supported in the assumption |
| Addressing | Req2.1.2.1 | Supported by Group MIHF_ID |
| Multicast Transport | Req2.1.3.{1,2} | Supported in the assumption |
| Group Management | Req2.1.4.1 | Supported by the group ID and the group key |
| Security Requirements | Req2.1.5.{1,2} | See the next slide |
| Transparency to MIH Users | Req2.2.1.1 | Supported by newly defined primitives |
| Reduced signaling | Req2.2.2.1 | Supported by the MKB technology |
| Scalability | Req2.2.3.{1,2} | Supported by the MKB technology |
| Backward compatibility | Req2.3.1.{1,2,3} | The state machine is not modified, and Broadcast MIHF ID is used as it was. |

# Conformance of the requirements

| | Authentication | Data Integrity | Confidentiality | Availability | Key management |
|---|---|---|---|---|---|
| Group manipulation | Supported by the signature of GM | Supported by the signature of GM | Supported by encryption by device keys | ? | Supported |
| Group command | Supported by the signature of CC | Supported by the signature of CC | Supported by encryption by a group key | ? | Supported |

# Summary

- The authors proposed a mechanism of group manipulation and a mechanism to issue commands to the group.
- The mechanism of group manipulation is based on an MKB technology.
- The authors provided the schemes to realize the use cases.
- The authors showed the advantages of the schemes, especially in reduction of signals and in scalability. The scheme in parallel satisfies the confidentiality requirement.
- The schemes satisfy the requirements listed in the CfP.