|  |  |
| --- | --- |
| Project | **IEEE 802.21 Media Independent Handover Services**  **IEEE 802.21d: Multicast Group Management**  **<**[**http://www.ieee802.org/21/**](http://www.ieee802.org/21/)**>** |
| Title | **Proposal for IEEE 802.21d solution** |
| Date Submitted | January, 2013 |
| Source(s) | **21-13-0004-00-MuGM** |
| Re: | IEEE 802.21d TG |
| Authors: | Antonio de la Oliva (UC3M), Daniel Corujo (ITAv), Carlos Guimarães (ITAv) |
| Abstract | This contribution provides a solution for the IEEE 802.21d |
| Purpose | Task Group Discussion and Acceptance |
| 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. |

# Introduction

This document presents a solution for:

* Modification to the Link\_ID data type so it can be used in group addressed primitives
* Description on how to use the digital signature to authenticate multicast messages.

## Modification of the Link\_ID data type

There are multiple primitives defined in the IEEE 802.21 base specification that can be useful in a multiple receiver scenario. Some example of these primitives are:

* MIH\_Event\_Subscribe
* MIH\_Link\_Get\_Parameters
* MIH\_Configure\_Thresholds
* MIH\_Link\_Actions

A common problem to all these primitives when sending them to multiple receivers is that all of them refer to a specific Link through a LINK\_ID parameter (non-optional). In case the primitive is addressed to multiple receivers, the LINK\_ID cannot be specified as it is defined now, since it will not make sense to all the nodes. Note that almost all primitives use the LINK\_TUPLE\_ID to refer to the link over which the primitive takes effect. This data type is defined as a LINK\_ID plus an optional LINK\_ADDR which refers to the PoA address.

For this reason we propose to extend the LINK\_ID definition in the following way:

|  |  |  |
| --- | --- | --- |
| Data type name | Derived from | Definition |
| LINK\_TUPLE\_ID | SEQUENCE( LINK\_ID, CHOICE(NULL, LINK\_ADDR)) | The identifier of a link that is associated with a PoA. The LINK\_ID contains the MN LINK\_ADDR. The optional LINK\_ADDR contains a link address of PoA. |
| LINK\_ID | SEQUENCE( LINK\_TYPE  LINK\_ADDR  ) | The identifier of a link that is not associated with the peer node. The LINK\_ADDR contains the address of this link. |
| LINK\_TYPE | UNSIGNED\_INT(1) | Represents the link type.a  Number assignments:  0: Reserved  1: Wireless – GSM  2: Wireless - GPRS  3: Wireless – EDGE  **5: Multicast (Virtual)**  15: Ethernet  18: Wireless - Other  19: Wireless – IEEE 802.11  22: Wireless – CDMA2000  23: Wireless – UMTS  24: Wireless – cdma2000-HRPD  27: Wireless – IEEE 802.16  28: Wireless – IEEE 802.20  29: Wireless – IEEE 802.22 |
| LINK\_ADDR | CHOICE( MAC\_ADDR, 3GPP\_3G\_CELL\_ID, 3GPP\_2G\_CELL\_ID, 3GPP\_ADDR, 3GPP2\_ADDR, OTHER\_L2\_ADDR) | A data type to represent an address of any link layer. **In case the LINK\_TYPE is 5 (Multicast/Virtual) or the LINK\_ADDR corresponds to a groupcast message, then the LINK\_ADDR must contain MAC\_ADDR (FF:FF:FF:FF:FF:FF)** |

Hence through the use of the Multicast Data Type we can provide 3 different behaviours:

* Target: All links 🡪 [[5,FF:FF:FF:FF:FF:FF],NULL]
* Target: Links of a certain technology 🡪 [[19,FF:FF:FF:FF:FF:FF],NULL]
* Target: Nodes of a certain technology connected to a specific PoA🡪[[19,FF:FF:FF:FF:FF:FF],PoA]

## Proposal for Digital signature authentication

This proposal focuses on the transport of the authentication information in the MIH protocol, not considering the algorithm used for performing the authentication. The definition of the key distribution mechanism is out of the scope of this document.

In order to carry the authentication payload, we propose the extension of current IEEE 802.21a mechanisms in the following way:

We define a new type of MIH frame

Security TLV

(Authentication payload)

MIH service specific TLVs

SAID

TLV

Destination MIHF

Identifier TLV

Source MIHF

Identifier TLV

MIH header

(S=1)

The SAID TLV and Security TLV are also modified according to the following table:

|  |  |  |
| --- | --- | --- |
| Name | Data Type | Description |
| SAID | SEQUENCE(ID\_TYPE, ID\_VALUE) |  |
| ID\_TYPE | ENUMERATED | The type of security association.  0: TLS-generated;  1: EAP-generated  **2: Authenticated (Need to be expanded with mechanisms of authentication)** |
| ID\_VALUE | OCTET\_STRING | Represents a security association identifier |
| SECURITY | CHOICE(TLS\_RECORD, MIH\_SPS\_RECORD,  **INTG\_BLOCK**) | Represents information which is carried in the security TLV. |
| TLS\_RECORD | OCTET\_STRING | Represents a TLS record. |
| MIH\_SPS\_RECORD | SE- QUENCE(ENCR\_BLOCK , CHOICE(INTG\_BLOCK, NULL)) | Represents data protected by an MIH security association. |
| ENCR\_BLOCK | OCTET\_STRING | Represents encrypted data. |
| INTG\_BLOCK | OCTET\_STRING | Represents integrity protected data. |