|  |  |
| --- | --- |
| Project | **IEEE 802.21a** **<https://mentor.ieee.org/802.21>** |
| Title | **Suggested Remedies on Fragmentation** |
| DCN | 21-10-00xx-00-0sec |
| Date Submitted |  |
| Source(s) | Yoshihiro Ohba (Toshiba) |
| Re: |  |
| Abstract | This document contains remedies to 802.21a/D03 associated with the author’s LB5b comment #37. |
| Purpose | Proposes changes in the current draft |
| 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#_blank) <[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#_blank)> and in *Understanding Patent Issues During IEEE Standards Development* [http://standards.ieee.org/board/pat/faq.pdf](http://standards.ieee.org/board/pat/faq.pdf#_blank) |

# Proposal

[1] Revise the following text in Clause 8.4.2.1:

An MIH message is fragmented only when MIH message is sent natively over an L2 medium such as Ethernet. The message is fragmented when the message size exceeds aFragmentationThreshold. The size of each of the fragments is the same except the last one, which may be smaller. The maximum fragment size is defined as the maximum value of aFragmentationThreshold, which shall be equal to the Maximum Transmission Unit (MTU) (in octets) of the link layer that is on the path between two MIHF nodes, ~~minus 44 octets, which is the maximum data expansion for each protected MIH PDU~~minus *securityOverhead* octets. When there is no MIH SA, *securityOverhead* is zero. The calculation of *securityOverhead* is described in Annex K. When the MTU of the link layer between two MIHF nodes is known, the maximum fragment size is set to the MTU (in octets) minus ~~44~~ *securityOverhead* octets. The method of determining such an MTU is outside the scope of this standard. When the MTU of the link layer between two MIHF nodes is unknown, the maximum fragment size is set to the minimum MTU of 1500 ~~1456~~ octets minus *securityOverhead* octets. When MIH message is sent using an L3 or higher layer transport, L3 takes care of any fragmentation issue, and the MIH protocol does not handle fragmentation in such cases.

[2] Revert the content of Figure K.1 to what is described in IEEE 802.21-2008.

[3] In Annex K, Insert the following text after Figure K.1 and before Figure K.2:

**Calculation of securityOverhead when there is an MIH SA**

To calculated *securityOverhead* when there is an MIH SA, the following parameters are used:

* x is 0 when Source MIHF Identifier TLV and Destination MIHF Identifier TLV are contained in the protected MIH message, otherwise, x is 1.
* y is 1 for TLS-generated MIH SA. Otherwise, y is 0.
* LSAID denotes the octet length of the SA ID TLV carried in the protected MIH message. LSAID depends on the implementation.
* LSID denotes the octet length of the Source MIHF Identifier TLV optionally carried in the protected MIH message. LSID depends on the implementation.
* LDID denotes the octet length of the Destination MIHF Identifier TLV optionally carried in the protected MIH message. LDID depends on the implementation.
* OSECTLV denotes the overhead of the Security TLV carried in the protected MIH message.
* OTYPE(y) denotes the overhead of the MIH data type contained in the Security TLV.
* OTLS denotes the overhead of the TLS record.

OTLS = 5, i.e., 1-octet TLSCiphertext.type plus 2-octet TLSCiphertext.version plus 2-octet TLSCiphertext.length [RFC5246].

* OENC denotes the overhead of encryption. OENC depends on the ciphersuite.
* OINTG denotes the overhead of integrity protection. OINTG depends on the ciphersuite.

*securityOverhead* is calculated as follows:

*securityOverhead* = LSAID –x (LSID + LDID)+ OSECTLV + OTYPE(y) + yOTLS + OENC + OINTG

Note that *securityOverhead* can be a negative value when x=1.

Since the maximum size of Security TLV is no more than the maximum size of Variable Payload of MIH message, which is 2^16-1 octets, the maximum values of OSECTLV and OTYPE(y) are shown below.

* OSECTLV = 3 (i.e., 1-octet TLV Type plus 2-octet TLV Length).
* OTYPE(0) = 6, i.e., 1-octet CHOICE Selector in CHOICE(TLS\_RECORD, MIH\_SPS\_RECORD) plus 2-octet Length field of ENCR\_BLOCK data plus 1-octet CHOICE Selector in MIH\_SPS\_RECORD plus 2- octet Length field of INTG\_BLOCK data.
* OTYPE(1) = 3, i.e., 1-octet CHOICE Selector in CHOICE(TLS\_RECORD, MIH\_SPS\_RECORD) plus 2-octet Length field of TLS\_RECORD data.

Table N.1 shows OENC and OINTG values for the MIH ciphersuites for EAP-generated MIH SA.

Table N.1: Encryption and integrity protection overhead for EAP-generated MIH SA

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Ciphersuite Code | Encryption | Integrity Protection | OENC | OINTG |
| 00000010 | AES-CBC | HMAC-SHA1-96 | 32( IV + padding) | 12 (MIC) |
| 00000100 | Null | HMAC-SHA1-96 | 0 | 12 (MIC) |
| 00000101 | Null | CMAC-AES | 0 | 12 (MIC) |
| 00000110 | AES\_CCM | 38 (SN+padding+MIC) | 0 |

For example, consider a case where Ciphersuite Code 00000010 (AES-CBC + HMAC-SHA1-96) is used for EAP-generated MIH SA (y=0) without containing Source MIHF Identifier TLV and Destination MIHF Identifier TLV in the protected MIH message (x=0), and the length of SAID TLV, the length of Source MIHF Identifier TLV, the length of Destination MIHF Identifier TLV are 30 octets, 20 octets and 30 octets , respectively. Then *securityOverhead* is computed as:

*securityOverhead* = LSAID –(LSID + LDID) + OSECTLV + OTYPE(0) + OENC + OINTG

= 30–(20+30)+3+6+44 = 33 (octets).

[4] Change Figure K.2 and its explanation as follows:

Figure K.2 shows the protected fragments for the original message shown in Figure K.1, when operating in the same condition as described in the above example with *securityOverhead*=33 (octets). The integer number within the curved-brackets of each field in Figure K.2 indicates the length of the field in octets. In Figure K.2, the fragment size before applying MIH protection is set to 1424 (=16\*89) octets to have the fragment size of 1499 octets after applying MIH protection, which gives the largest number of 16-octet blocks (89) under the condition that the resulting protected fragment does not exeeds 1500 octets.

Figure K.2—Example of protected MIH fragment message