IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Draft Text for 9.6.7 eBCS Info Frame | | | | |
| Date: 2020-01-13 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Hitoshi Morioka | SRC Software | Fukuoka, JAPAN |  | [hmorioka@src-soft.com](mailto:hmorioka@src-soft.com) |
| Antonio de la Oliva | InterDigital | Madrid, Spain |  | aoliva@it.uc3m.es |

Abstract

This document describes a draft text proposal for eBCS Info frame.

*Yellow marked numbers are temporal and to be assigned by ANA.*

9. Frame formats

9.6 Action frame format details

9.6.7 Public Action details

9.6.7.1 Public Action frames

*Add new entry (and adjust the reserved value) to Table 9-362 as shown below.*

|  |  |
| --- | --- |
| **Public Action field value** | **Description** |
| <ANA> | eBCS Info |
| <ANA+1> - 255 | Reserved |

*Add the following new subclause under 9.6.7*

9.6.7.52 eBCS Info frame format

The format of the Action field of the eBCS Info frame is shown in Figure 9-bc1 (eBCS Info frame Action field format).

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Category | Public Action | Sequence Number | Timestamp | Flags | eBCS Info Interval | Authentication Algorithm | Allowable Time Difference | Certificate Length |
| Octets: | 0 | 1 | 8 | 8 | 1 | 1 | 1 | 1 | 2 |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
|  | Certificate | Contents Information Number | Contents Information 1 | Contents Information 2 | … | Contents Information N | Singnature |
| Octets: | variable | 1 | variable | variable |  | variable | variable |

**Figure 9-bc1 eBCS Info frame Action field format**

The Category field is defined in 9.4.1.11 (Action field).

The Public Action field is defined in 9.6.7.1 (Public Action frames).

The Sequence Number field is a 64bit unsigned integer that is dot11EBCSInfoSequence.

The Timestamp field is a 64bit unsigned integer that is the ellapsed time from 1970/1/1 0:00 UTC in units of millisecond.

The Flags field is shown in Figure 9-bc2 (eBCS Info frame Flags field format)

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | B0 | B1 | B2 | B3 | B4 | B5 | B6 | B7 |
|  | Fragmentation Number | | | Fragmentation Index | | | Reserved | |
| Bits: | 3 | | | 3 | | | 2 | |

**Figure 9-bc2 eBCS Info frame Flags field format**

The Fragmentation Number subfield in the Flags field is a 3bit unsigned integer that is the total number of the fragmentation.

The Fragmentation Index subfield is the Flags field is a 3bit unsigned integer that is the fragmentation index of the eBCS Info frame.

Fragmentation procedure is described in 11.55.2 (eBCS Info frame generation).

The eBCS Info Interval field is an 8bit undsigned integer that is the interval of the eBCS Info frame transmission interval configured as dot11EBCSInfoInterval in units of 100millisecond. In case of using PKFA, the eBCS Info Interval field is set to 0.

The Authentication Algorithm field is an 8bit unsigned integer that is show in Table 9-bc1 (eBCS Info frame Authentication Algorithm field)

**Table 9-bc1 eBCS Info frame Authentication Algorithm field**

|  |  |
| --- | --- |
| **Value** | **Authentication Algorithm** |
| 0 | Public key frame authentication (PKFA) with RSA-2048 (12.15.2 eBCS public key frame authentication) |
| 1 | Public key frame authentication (PKFA) with DSA-L2048N256 (12.15.2 eBCS public key frame authentication) |
| 2 | Public key frame authentication (PKFA) with ECDSA-P256 (12.15.2 eBCS public key frame authentication) |
| 3-15 | reserved |
| 16 | Hash chain frame authentication (HCFA) (12.15.3 eBCS Hash chain frame authentication) with RSA-2048 and SHA3-256 |
| 17 | Hash chain frame authentication (HCFA) (12.15.3 eBCS Hash chain frame authentication) with DSA-L2048N256 and SHA3-256 |
| 18 | Hash chain frame authentication (HCFA) (12.15.3 eBCS Hash chain frame authentication) with ECDSA-P256 and SHA3-256 |
| 19-31 | reserved |
| 32 | Hash chain frame authentication (HCFA) (12.15.3 eBCS Hash chain frame authentication) with RSA-2048 and SHA3-384 |
| 33 | Hash chain frame authentication (HCFA) (12.15.3 eBCS Hash chain frame authentication) with DSA-L2048N256 and SHA3-384 |
| 34 | Hash chain frame authentication (HCFA) (12.15.3 eBCS Hash chain frame authentication) with ECDSA-P256 and SHA3-384 |
| 35-255 | reserved |

The Allowable Time Difference field is 8bit unsigned integer that is the allowable time difference between the eBCS transmitter and the eBCS receivers in case of the PKFA, or the TESLA key change interval in case of the HCFA. The unit of the value is millisecond.

The Certificate Length field is a 16bit unsigned integer that is the length of the certificate in units of octet.

The Certificate field is the X.509 certificate of the eBCS transmitter in the DER format (Distinguished Encoding Rules, ITU-T Recommendation X.680 (2002)).

The Contents Information Number field is an 8bit unsigned integer that indicates the number of the Contents Information fields.

The Contents Information field is shown in Figure 9-bc3 (eBCS Info frame Contents Information field format).

|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | Flags | Higher Layer Destination Address | TESLA Base Key  (optional) | Previous Period TESLA Base Key Number (optional) | Previous Period TESLA Base Key Sequence 0 (optional) | Previous Period TESLA Base Key 0 (optional) | … | Previous Period TESLA Base Key Sequence N (optional) | Previous TESLA Base Key N (optional) |
| Octets: | 1 | variable | variable | 1 | 1 | variable |  | 1 | variable |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | Instant Authenticator 0 Number (optional) | Instant Authenticator 1 Number (optional) | Instant Authenticators 0 (optional) | Instant Authenticator 1 (optional) | Title Length (optional) | Title (optional) |
| Octets: | 2 | 2 | variable | variable | 1 | variable |

|  |  |  |
| --- | --- | --- |
|  | Data Length (optional) | Data (optional) |
| Octets: | 2 | variable |

**Figure 9-bc3 eBCS Info frame Contents Information field format**

The Flags subfield in the Contents Information field is shown in Figure 9-bc4 (Flags subfield in the Contents Informatino field format)

|  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | B0 | | B1 | B2 | | B3 | B4 | B5 | B6 | B7 |
|  | Title Included | | Data Included | Higher Layer Protocol | | | | Reserved | | |
| Bits: | 1 | 1 | | | 3 | | | 3 | | |

**Figure 9-bc4 Flags subfield in the Contents Information field format**











The Title Included subfield in the Flags subfield is set to 1 if the Title Length subfield and the Title subfield are present, otherwise set to 0.

The Data Included subfield in the Flags subfield is set to 1 if the Data Length subfield and the Data subfield are present, otherwise set to 0.

The Higher Layer Protocol subfield in the Flags subfield is a 3bit unsigned integer that shown in Table 9-bc2 (Higher Layer Protocol subfield). UDP/hostname may be used for uplink eBCS only. Others are ud

**Table 9-bc2 Higher Layer Protocol subfield**

|  |  |
| --- | --- |
| **Value** | **Higher Layer Protocol** |
| 0 | UDP/IPv4 |
| 1 | UDP/IPv6 |
| 2 | UDP/hostname (Uplink only) |
| 3 | MPEG Transport stream identifier |
| 4 | MAC Address |
| 5-7 | Reserved |

The Higher Layer Destination Address subfield in the Contents Information field is the higher layer destination address and port encoded as following.

If the higher layer protocol is UDP/IPv4, the format of the Higher Layer Destination Address subfield format is shown in the Figure 9-bc5 (Higher Layer Destination Address subfield format for UDP/IPv4). The Destinatin UDP Port subfield is encoded in little endian.

|  |  |  |
| --- | --- | --- |
|  | Destination IPv4 Address | Destination UDP Port |
| Octets: | 4 | 2 |

**Figure 9-bc5 Higher Layer Destination Address subfield format for UDP/IPv4**

If the higher layer protocol is UDP/IPv6, the format of the Higher Layer Destination Address subfield format is shown in the Figure 9-bc6 (Higher Layer Destination Address subfield format for UDP/IPv6). The Destinatin UDP Port subfield is encoded in little endian.

|  |  |  |
| --- | --- | --- |
|  | Destination IPv6 Address | Destination UDP Port |
| Octets: | 16 | 2 |

**Figure 9-bc6 Higher Layer Destination Address subfield format for UDP/IPv6**

If the higher layer protocol is UDP/hostname, the format of the Higher Layer Destination Address subfield format is shown in the Figure 9-bc7 (Higher Layer Destination Address subfield format for UDP/hostname). The Hostname Length subfield indicates the length of the Hostname subfield. The Hostname subfield is the hostname in UTF-8. The Destinatin UDP Port subfield is encoded in little endian.

|  |  |  |  |
| --- | --- | --- | --- |
|  | Hostname Length | Hostname | Destination UDP Port |
| Octets: | 1 | variable | 2 |

**Figure 9-bc7 Higher Layer Destination Address subfield format for UDP/hostname**

If the higher layer protocol is MPEG Transport stream, the format of the Higher Layer Destination Address subfield format is shown in the Figure 9-bc8 (Higher Layer Destination Address subfield format for MPEG Transport stream). The MPEG Transport stream Length subfield indicates the length of the MPEG Transport stream subfield. The MPEG Transport stream subfield is the MPEG Transport stream identifier in UTF-8.

|  |  |  |
| --- | --- | --- |
|  | MPEG Transport stream length | MPEG Transport stream |
| Octets | 1 | Variable |

**Figure 9-bc8 Higher Layer Destination Address subfield format for MPEG Transport stream**

If the higher layer protocol is MAC Address, the format of the Higher Layer Destination Address subfield format is shown in the Figure 9-bc9 (Higher Layer Destination Address subfield format for MAC Address).

|  |  |
| --- | --- |
|  | MAC Address |
| Octets | 6 |

**Figure 9-bc9 Higher Layer Destination Address subfield format for MAC Address**

The TESLA Base Key subfield is present if the authentication algorithm is HCFA.

The TESLA Base Key subfield contains the first TESLA base key of the HCFA period that starts from this eBCS Info frame. The length of the TESLA Base Key frame is determined by the authenticatino algorithm.

The Previous Period TESLA Base Key Number subfield, the Previous Period TESLA Base Key Sequence *n* subfield(s) and the Previous Period TESLA Base Key *n* subfield(s) are present if the authentication algorithm is HCFA and the previous HCFA period exists.

The Previous Period TESLA Base Key Number subfield contains the number of the TESLA Base Key(s) to be disclosed for the previous HCFA period. The Previous Period TESLA Base Key Sequence *n* subfield contains the TESLA sequence number of the Previous TESLA Base Key *n* subfield. The Previous Period TESLA Base Key *n* subfield contains the TESLA base key to be disclosed for the previous HCFA period. The length of the Previous Period TESLA Base Key subfiled is determined by the authentication algorithm.

The Instant Authenticator 0 Number, the Instant Authenticator 1 Number, the Instant Authenticators 0 and the Instant Authenticators 1 subfield is present if the authentication algorithm is HCFA.

The Instant Authenticator 0 Number subfield and the Instant Authenticator 1 Number subfield contain the number of the instant authendicators in the TESLA period 0 and 1 respectively. The Instant Authenticators 0 subfield and the Instant Authenticators 1 subfield contain the Instant Authenticators for the TESLA period 0 and 1 respectively. The order of the instant authenticators in the Instant Authenticators 0 and 1 subfield is ascending order of the data sequence number. The length of the Instant Authenticators 0 and 1 is determined by the authentication algorithm and the number of the instant authenticators.

The Title Length subfield and the Title subfield are present if the Title Included flag in the Flags subfield is set to 1.

The Title Length subfield in the Contents Information field is 8bit unsigned integer that is the length of the following Title field in units of octes.

The Title subfield in the Contents Information field is the human readable title of the contents coded in UTF-8.

The Data Length subfield is the length of the Data subfield.

The Data subfield is shown in the Figure 9-bc10 (Data subfield format).

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | Data Flags | Service URL Length  (Optional) | Service URL  (Optional) | Vendor specific  (Optional) |
| Octets | 1 | 1 | Variable | variable |

**Figure 9-bc10 Data subfield format**

The Data Flags subfield is shown in Figure 9-bc11 (Data Flags subfield format).

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | B0 | | B1 | B2 | | B3 | | | B4 | B5 | B6 | B7 |
|  | Content with Restriction | | Service URL Present | Vendor Specific Present | | | Reserved | | | | | |
| Bits: | 1 | 1 | | | 1 | | | 5 | | | | |

**Figure 9-bc11 Data Falgs subfield format**

The Content with restriction bit indicates if the content requires of some kind of offline registration to be accessed. The registration process is out of the scope of this document.

The Service URL Present bit indicates that the Service URL Length subfield and Service URL subfield are present in the Data subfield.

The Voendor Specific Present bit indicates that the Vendor Specific subfield is present in the Data subfield.

In the case the Data Flags indicate the Service URL subfield is present, the Service URL Length subfield is a 1-octet field whose value is set to 1 plus the number of octets in the Service URL field.

The Service URL field is a variable length field that indicates the URL at which information relevant to the corresponding eBCS service might be retrieved. The Service URL field is formatted in accordance with IETF RFC 3986.

The Vendor specific subfield is to be defined by application specific requirements.

The Signature field is the digital signature of the eBCS Info frame that is generated by the certificate of the eBCS transmitter. The length of the Signature field is determined from the public key algorithm of the authentication algorithm.