IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Unambiguous Parsing of an SAE Commit message | | | | |
| Date: 2018-09-03 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Dan Harkins | HPE | 3333 Scott boulevard,  Santa Clara, California,  United States of America |  |  |
|  |  |  |  |  |

Abstract

The introduction of the password identifier to SAE introduces some parsing challenges when multiple, optional components may be part of the frame. This submission suggests a procedure to ensure that the frame can be properly and unambiguously parsed.

Four possibilities of a received Commit message. Group, Token, Scalar and Element are fields; the Password Identifier is an Element. All but the Group are variable length and the token and Password Idetifier are optional.

Group Scalar Element

Group Token Scalar Element

Group Scalar Element Identifier

Group Token Scalar Element Identifier

***Instruct the editor to add the following text to section 12.4.7.4:***

**12.4.7.4 Encoding and decoding of SAE Commit messages**

When transmitting an SAE Commit message, the scalar and element shall be converted to octet strings and

placed in the Scalar field and FFE field, respectively. The scalar shall be treated as an integer and converted

into an octet string of length m such that 28m > r , where r  is the order of the group, according to 12.4.7.2.2

(Integer to octet string conversion), and the element shall be converted into (an) octet string(s) according to

12.4.7.2.4 (Element to octet string conversion). When receiving an SAE Commit message the component

octet strings in the Scalar field and Element field shall be converted into a scalar and element, respectively,

according to 12.4.7.2.3 (Octet string to integer conversion) and 12.4.7.2.5 (Octet string to element

conversion), respectively.

NOTE—Anti-clogging tokens, password identifiers, and vendor specific additions may be optionally present in a received Commit message. Since the size of the Scalar field and Element field are determined by the Group field, any anti-clogging token present will be of a size determined by the recipient, and the Password Identifier is an element with a well-defined prefix, the Commit message can be unambiguously parsed using the following technique:

1. Compute the following values:
   1. Base length is the sum of the length of the Group field, the length of the Scalar field, and the length of the Element field
   2. Token length is the size of a requested anti-clogging token
2. If the length of the Commit message equals the base length then there is no token, no password identifier, and no vendor specific additions;
3. If the length of the Commit message is greater than the base length but less than the sum of the base length and token length and a Password Identifier element follows the Element field, then there is a password identifier and no token. If a Password Identifier element does not follow the Element field or the length of the Commit message indicates there are additional octets following the Password Identifier element, then there are vendor specific additions.
4. If the length of the Commit message is greater than the sum of the base length and the token length and a Password Identifier element follows the Element field, then there is a password identifier and a token. If a Password Identifier element does not follow the Element field or the length of the Commit message indicates there are additional octets following the Password Identifier element, then there are vendor specific additions.

**References:**