IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Authentication and Deauthentication in an MBSS | | | | |
| Date: 2011-05-18 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Dan Harkins | Aruba Networks | 1322 Crossman ave, Sunnyvale, CA | +1 408 227 4500 | dharkins at arubanetworks dot com |
|  |  |  |  |  |

Abstract

This submission describes the specific behavior a STA performs in an MBSS or when using SAE.

***Modify section 11.3.1.1 as indicated:***

* + - 1. Authentication—originating STA

If the requested authentication mechanism is other than FT authentication, the SME shall delete any PTKSA and temporal keys held for communication with the indicated STA by using MLME-DELETEKEYS.request primitive (see 8.4.10 (RSNA security association termination)), and the STA shall be set to state 1 before invoking MLME-AUTHENTICAT.request primitive.

Upon receipt of an MLME-AUTHENTICATE.request primitive, the originating STA shall authenticate with the indicated STA using the following procedure:

1. The MLME shall execute one of the following:
   1. For the Open System or Shared Key authentication algorithm, the authentication mechanism described in 8.2.2.2 (Open System authentication) or 8.2.2.3 (Shared Key authentication), respectively.
   2. For the FT authentication algorithm in an ESS, the authentication mechanism described in 11A.5 (FT Protocol).
   3. For SAE authentication in an ESS, IBSS, or MBSS, the authentication mechanism described in 8.2a (Authentication using a password).
2. If the authentication was successful within the AuthenticateFailureTimeout, the state for the indicated STA shall be set to state 2.
3. The MLME shall issue an MLME-AUTHENTICATE.confirm primitive to inform the SME of the result of the authentication.

***Modify section 11.3.1.2 as indicated:***

* + - 1. Authentication—Destination STA

Upon receipt of an Authentication frame with authentication transaction sequence number equal to 1, the destination STA shall authenticate with the originating STA using the following procedure:

1. The MLME shall issue an MLME-AUTHENTICATE.indication primitive to inform the SME of the authentication request.
2. Upon receiving a MLME-AUTHENTICATE.indication primitive, if the requested authentication mechanism is other than FT authentication and if Management Frame Protection was not negotiated when the PTKSA(s) were created,
   1. If the destination STA is not in an MBSS, the SME shall delete any PTKSA and temporal keys held for communication with the originating STA by using the MLME-DELETEKEYS.request primitive (see 8.4.10 RSNA security association termination)), and
   2. The STA shall be set to state 1.
3. Upon receipt of an MLME\_AUTHENTICATE.response primitive, if the ResultCode is not SUCCESS, the MLME shall transmit an Authentication frame with authentication transaction sequence number set to to with a status code, as defined in 7.3.1.9 (Status Code field), other than Successful, and the state for the originating STA shall be left unchanged.
4. If the ResultCode in the MLME-AUTHENTICATE.response primitive was SUCCESS, the STA shall execute one of the following:
   1. For the Open System or Shared Key authentication algorithm, the authentication mechanism described in 8.2.2.2 (Open System authentication) or 8.2.2.3 (Shared Key authentication), respectively.
   2. For the FT authentication algorithm in an ESS, the authentication mechanism described in 11A.5 (FT Protocol).
   3. For SAE authentication in an ESS, IBSS, or MBSS, the authentication mechanism described in 8.2a (Authentication using a password).
5. If the authentication was successful, the state for the originating STA shall be set to state 2.
6. If the destination STA is an AP, its SME shall inform the DS of the disassociation, if the state for the originating STA was State 3 or 4.

If the STA is in an IBSS, if the SME decides to initiate an RSNA, and if the SME does not know the security policy of the peer, it may issue an individually addressed Probe Request frame to the peer by invoking an MLME-SCAN.request primitive to discover the peer’s security policy.

***Modify section 11.3.1.3 as indicated:***

* + - 1. Deauthentication—originating STA

The SME shall delete any PTKSA and temporal keys held for communication with the indicated STA by using the MLME-DELETEKEYS.request primitive (see 8.4.10 (RSNA security association termination)) and by invoking MLME-SETPROTECTION.request(None), and the STA shall be set to state 1 before invoking the MLME-DEAUTHENTICAT.request primitive.

Upon receipt of an MLME-DEAUTHENTICATE.request primitive, the originating STA shall deauthenticate with the indicated STA using the following procedure:

1. If the State for the indicated STA is State 2, State 3, or State 4, the MLME shall transmit a Deauthentication frame to the indicate STA.
2. The MLME shall include in theDeauthentication frame the appropriate reason code for the STA deauthentication as defined in Table 7-22 (Reason codes) of 7.3.1.7 (Reason Code field).
3. The state for the indicated STA shall be set to State 1.
4. The MLME shall issue an MLME-DEAUTHENTICATE.confirm primitive to inform the SME of the completion of the deauthentication.
5. If the STA is an AP, its SME shall inform the DS of the disassociation, if the state for the indicated STA was State 3 or State 4.
6. If the STA is a mesh STA, its SME shall inform the Mesh peering instance controller (see 11C.3.4 (Mesh peering instance controller)) of the deauthentication.

***Modify section 11.3.1.4 as indicated:***

* + - 1. Deauthentication—destination STA

Upon receipt of a deauthentication frame from a STA for which the state is State 2, State 3, or State 4, the destination STA shall deauthenticate with the originating STA using the following procedure:

1. The state for the originating STA shall be set to State 1.
2. The MLME shall issue an MLME-DEAUTHENTICATE.indication primitive to inform the SME of the deauthentication.
3. Upon receiving an MLME-DEAUTHENTICATE.indication primitive, if Management Frame Protection was not negotiated when the PTKSA(s) were created,
   1. The SME shall delete any PTKSA and temporal keys held for communication with the originating STA by using the MLME-DELETEKEYS.request primitive (see 8.4.10 (RSNA security association termination)) and by invoking MLME-SETPROTECTION.request(None), and
   2. The STA shall be set to state 1.
4. If the STA is an AP, its SME shall inform the DS of the disassociation, if the state for the originating STA was State 3 or State 4.
5. If the STA is a mesh STA, its SME shall inform the Mesh peering instance controller (see 11C.3.4 (Mesh peering instance controller)) of the deauthentication.

* Deleting mesh peering instances

The mesh peering instance controller deletes a mesh peering instance after either:

* Expiry of a holding timer (see 11C.4.4 (Timers))
* The acceptance of a peer’s response to an existing request to close the peering (see 11C.4.3 (Events and actions)).
* Indication from the SME that the peer mesh STA, or candidate peer mesh STA, has deauthenticated.

When the deletion occurs, the mesh TKSA that is bound to the mesh peering shall be deleted.

**References:**