IEEE P802.11  
Wireless LANs

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| AP Protected TXOP Negotiation | | | | |
| Date: 2010-09-10 | | | | |
| Author(s): | | | | |
| Name | Affiliation | Address | Phone | email |
| Alex Ashley | NDS Ltd | One London Road, Staines, Middlesex, TW18 4EX, UK | +44 1784 848770 | aashley at nds dot com |
| Dan Harkins | Aruba Networks | 1322 Crossman ave, Sunnyvale, CA | +1 408 227 4500 | dharkins at arubanetworks dot com |

Abstract

P802.11aa draft 1.0 defines a protocol to allow access points to co-operatively schedule their HCCA TXOPs to avoid conflict. However, this protocol provides no protection from denial of service attacks due to the lack of any entity authentication. For example in its current form it would be fairly simple to fake the TA MAC address in a request to allow a non-AP STA to create a frame that appears to come from an AP.

This proposal builds upon the security features added by P802.11s to allow a peer-to-peer negotiation of a peerwise master key (PMK) over an unsecured channel between Access Points, using the Diffie–Hellman protocol. Existing RSN methods are then used to derive the keys required for protected transmission of frames between these APs, using “protected dual of public action” frames.

This submission is based upon P802.11aa draft 1.02

***Instructions to the TGaa editor are marked in purple italics.***

***Changes to the draft are marked in blue italics. underline indicates an addition and ~~strikethrough indicates deletion~~.***

7.3 Management frame body components

7.3.1 Fields that are not information elements

7.3.1.9 Status Code field

***Modify Table 7-23 as shown:***

|  |  |
| --- | --- |
| Table 7-23—Status codes | |
| Status code | Meaning |
| <ANA1> (#235) | Requested TCLAS processing has been terminated by the AP (#84) |
| <ANA2> (#235) | The TS ~~should not be created because the schedule~~ conflicts with an existing schedule; however, a suggested alternate schedule is provided ~~so that the initiating STA may attempt to create a schedule with the suggested changes~~ (#84) |
| *(<ANA>+2) ~~31~~*— *65535* | Reserved |

7.3.2.25 RSN element

**7.3.2.25.2 AKM Suites**

Insert one new row and change the existing ‘Reserved’ row in Table 7-34 (AKM suite selectors) as follows.

Table 7-34—AKM suite selectors

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **OUI** | **Suite Type** | **Authentication type** | **Key management type** | **Key derivation type** |
| 00-0F-AC | <ANA> | APPeerKey Authentication with SHA-256 or using PMKSA caching as defined in 8.4.6.2 (Cached PMKSAs and RSNA key management) with SHA-  256 Key Derivation | RSNA key management as defined in 8.5 (Keys and key distribution), PMKSA caching as defined in 8.4.6.2 (Cached PMKSAs and RSNA key management) with SHA256 Key Derivation | As defined in  8.5.1.5.2 |
| 00-0F-AC | <ANA>+1 – 255 | Reserved | Reserved | Reserved |

7.3.2.27 Extended Capabilities information element

***Modify Table 7-35a as shown:***

***Insert the rows for Bit <ANA>, and change the Reserved row in Table 7-35a as follows (note that the entire table is not shown here):***

Table 7-35a—Capabilities field

|  |  |  |
| --- | --- | --- |
| Bit | Information | Notes |
| <ANA> | Public TXOP Negotiation | The STA sets the Public TXOP Negotiation field to 1 when the MIB attribute dot11PublicTXOPNegotiationActivated is true and sets it to 0 otherwise. See 11.aa24.3. |
| <ANA> | Protected TXOP Negotiation | The STA sets the Protected TXOP Negotiation field to 1 when the MIB attribute dot11ProtectedTXOPNegotiationActivated is true and sets it to 0 otherwise. See 11.aa24.3. |
| *(<ANA>*  *+1) —*  *n\*8* | Reserved | All other bits are reserved, and are set to 0 on transmission and ignored on reception. |

*Modify Table 7-57e as shown.*

7.4.7.1 Public Action frames

***Change Table 7-57e by inserting new rows and change Reserved row Action field value as shown below***

|  |  |
| --- | --- |
| Table 7-57e—Public Action field values | |
| Action Field Value | Description |
| <ANA> | QLoad Request |
| <ANA+1> | QLoad Report |
| <ANA+2> | HCCA TXOP Advertisement |
| <ANA+3> | HCCA TXOP Response |
| <ANA+4> | Public Key |
| <ANA+5>-255 | Reserved |

*Add the following clause after 7.4.7.aa21.*

7.4.7.aa22 Public Key frame

The Public Key frame is used by an AP to provide its public key to peer APs and to request the peer’s public key. The format is defined in Figure 7-aa23.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Category | Action | Request Type | Group | Public Key |
| Octets | 1 | 1 | 1 | 2 | variable |

Figure 7-aa23— Public Key frame body format

The Category field is set to the value indicating a Public Action frame, as specified in Table 7-24.

The Action field is set to the value specified in Table 7-57e for a Public Key Public Action frame.

The Request Type field is set to a number to identify the usage mode of this frame. The Request Types are shown in Table 7-aa9.

|  |  |
| --- | --- |
| Table 7-aa9—Request Type definitions | |
| Name | Usage Mode |
| Request | 0 |
| Response | 1 |
| Reserved | 2-255 |

The Request Type field is set to Request to indicate that a Public Key is being requested from a peer AP.

The Request Type field is set to Response to indicate that this frame is in response to a Public Key Public Action frame.

The group field is used to indicate which cryptographic group was used when generating the public key, and is defined in 7.3.1.40 (Finite Cyclic Group field)

The Public Key field contains the public key of the AP that is sending this Public Key Public Action frame and is defined in 7.3.1.37 (Scalar field).

The use of the Public Key frame is described in clause .

*Add editing instructions for 7.4.9a.1 as shown.*

**7.4.9a.1 Protected Dual of Public Action details**

Insert two new rows in Table 8-57m as indicated and update the reserved range accordingly.

**Table 7-57m—Public Action field values defined for Protected Dual of Public Action frames**

|  |  |
| --- | --- |
| **Public Action**  **field value** | **Description** |
| <ANA> | Protected HCCA TXOP Advertisement frame |
| <ANA> | Protected HCCA TXOP Response frame |

***Add the following two new sub-clauses to the end of 7.4.9a.1:***

7.4.9a.9aa Protected HCCA TXOP Advertisement frame format

The Protected HCCA TXOP Advertisement frame format is the same as the HCCA TXOP Advertisement frame format (see 7.4.7.aa20 (HCCA TXOP Advertisement frame format)). It is used instead of the HCCA TXOP Advertisement frame format when protected TXOP Negotiation is enabled.

7.4.9a.10aa Protected HCCA TXOP Response frame format

The Protected HCCA TXOP Response frame format is the same as the HCCA TXOP Response frame format (see 7.4.7.aa21 (HCCA TXOP Response frame format)). It is used instead of the HCCA TXOP Response frame format when protected TXOP Negotiation is enabled.

**8. Security**

8.1 Framework

***Add the following two new sub-clauses to the end of 8.2a:***

***Insert 8.2aa after clause 8.2a***

8.2aa RSNA AP PeerKey Support

8.2aa.1 AP PeerKey overview

The AP PeerKey protocol provides session identification and data confidentiality for an AP-to-AP connection. An AP PeerKey association, composed of an SMKSA and an STKSA, shall be allowed only between APs that support:

* Robust AV Streaming (as indicated by the Robust AV Streaming bit being set in the Extended Capabilities element), and
* HCCA TXOP Negotiation (as indicated by the Protected TXOP Negotiation bit being set in the Extended Capabilities element).

AP PeerKey uses various functions and data to accomplish its task and assumes certain properties about each function. These are:

— H is an “extractor” function (see RFC 5869) that concentrates potentially dispersed entropy from an input to create an output that is a cryptographically strong, pseudo-random key. This function takes as input a non-secret “salt” and a secret input and produces a fixed-length output.

— A finite cyclic group is negotiated for which solving the discrete logarithm problem is computationally infeasible.

When used with AKM <ANA> (“AP PeerKey”) from Table 7.3.2.25.2 (AKM Suites), H shall be instantiated as H(salt, ikm) = HMAC-SHA256(salt, ikm)

Other instantiations of function H require creation of a new AKM identifier.

8.2aa.2 AP PeerKey Protocol

AP PeerKey uses the same discrete logarithm cryptography as SAE (as described in 8.2a) to achieve key agreement. Each party to the exchange has a public and private key with respect to a particular set of domain parameters that define a finite cyclic group. Groups may be based on Elliptic Curve Cryptography (ECC) or Discrete Logarithm Cryptography (FFC). Each component of a group is referred to as an “element”. Groups are negotiated using an identifying number from a repository maintained by IANA as “Group Description” attributes for RFC 2409 (IKE). The repository maps an identifying number to a complete set of domain parameters for the particular group. For the purpose of interoperability, APs that have dot11ProtectedHCCATXOPNegotiationImplemented set to true shall support group nineteen (19), an ECC group defined over a 256-bit prime order field.

AP PeerKey uses one arithmetic operator that takes one element and one scalar value to produce another element (called the “scalar operation”). The convention used here is to represent group elements in upper-case and scalar values in lower-case. The scalar operation takes an element and a scalar and is denoted *scalar-op(x,Y)*.

The private key (d) shall be chosen randomly such that 1 < d < r, where r is the order of the group. The public key (Q) shall be produced using the formula:

Q = scalar-op(d,G)

where G is the generator (aka. base point) of the group

An AP for which dot11ProtectedTXOPNegotiationActivated is true shall support at least one public key from cyclic group nineteen. An AP for which dot11ProtectedTXOPNegotiationActivated is true may support multiple public keys from multiple cyclic groups. An AP for which dot11ProtectedTXOPNegotiationActivated is true and that supports the Multiple BSSID capability may use one public key across multiple BSSIDs or it may choose to generate a public key for each supported BSSID.

An AP requests the public key of a peer AP by sending a Public Key Public Action frame with the Request Type field set to “Request”. This frame contains the public key of the initiating AP.

An AP for which dot11ProtectedTXOPNegotiationActivated is true shall reply to a Public Key Public Action frame for which the Request Type is “Request”, with a Public Key Public Action frame with the Request type set to “Response”. If the group field in the public key request is a group that is supported by the responding AP, the AP replies with a public key of the same group as the request, generating such a key pair if required. If the group field in the public key request specifies a group that is not supported by the responding AP, or it is not able to generate a key in this group, the response will be of a different group to the request.

Regardless of the value of the Request Type field, the Public Key field shall contain the public key and the Group field the group identifier of the domain parameters used to construct the public-private key pair of the AP transmitting the Public Key Public Action frame.

Once an AP has the public key of a peer AP, it can compute the Diffie-Hellman shared secret for an AP-to-AP peer link using scalar-op() and function F from section 8.2a.4:

*k* = F(scalor-op(d,Qp))

Where

*d* is the private key of the AP that is calculating *k*

*Qp* is the public key of the peer AP

An entropy of the shared secret shall then be extracted using function H to produce *keyseed*:

*keyseed* = H(<0>32, *k*)

and the PMK shall be derived using the KDF from section 8.5.1.5.2:

*PMK* = KDF-256(*keyseed*, “AP Peerkey Protocol”,

Max(LOCAL-MAC, PEER-MAC) || Min(LOCAL-MAC, PEER-MAC) )

Where LOCAL-MAC is the AP’s BSSID and PEER-MAC is the peer AP’s BSSID. *Keyseed* shall be irretrievably destroyed after the PMK is generated.

To enable the use of Protected HCCA TXOP Advertisement frames and Protected HCCA TXOP Response frames, the Authenticated Mesh Peering Exchange (as defined in 11C.5) is used to enable Security Capability Selection, Key Management and prove possession of the PMK (and implicitly the private key that corresponds to the peer’s public key). If the Authenticated Mesh Peering Exchange procedure completes successfully, Protected HCCA TXOP Advertisement frames and Protected HCCA TXOP Response frames may be used in the HCCA TXOP Negotiation procedures, as defined in 11.aa24.3.

NOTE—The PMK, and any key derived from it, is not authenticated in any way nor is it bound to any identity. This protocol protects an AP that cooperates in scheduling its HCCA TXOPs using Protected HCCA TXOP Advertisements because no other entity knows its private key and cannot forge Protected HCCA TXOP Advertisement frames. An AP who receives a Protected HCCA TXOP frame knows it was sent by the holder of a particular private key, and no one else, and can therefore establish which APs are acting in good faith in their HCCA TXOP scheduling.

10. Layer management

10.3 MLME SAP Interface

***Add the following two new sub-clause to the end of 10.3:***

10.3.aa74 AP PeerKey management

These set of primitives support the AP PeerKey protocol to provide session identification and data confidentiality for an AP-to-AP connection, as described in .

10.3.aa74.1 MLME-APPEERKEY.request

10.3.aa74.1.1 Function

This primitive is used by an AP to transmit a public key to a specified AP and to request the peer’s public key.

10.3.aa74.1.2 Semantics of the service primitive

The primitive parameters are as follows:

MLME-APPEERKEY.request (

Peer MAC Address,

Request Type,

Group,

Public Key

)

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Valid range | Description |
| Peer MAC Address | MACAddress | Any valid individual MACAddress | The address of the peer MAC entity to which the Public Key Public Action frame shall be sent. |
| Request Type | Integer | As defined in | Specifies the type of request |
| Group | Finite Cyclic Group field | As defined in 7.3.1.40 | Specifies cyclic group from which the public key was generated |
| Public Key | Scalar field | As defined 7.3.1.37 | The Public Key of the AP sending this Public Key Public Action frame |

10.3.aa74.1.3 When Generated

The primitive is generated by the SME at an AP to request the sending of a Public Key Public Action frame to another AP indicated by Peer MAC Address.

10.3.aa74.1.4 Effect of Receipt

On receipt of this primitive, the MLME constructs a Public Key Public Action frame. The AP then attempts to transmit this frame to the AP indicated by Peer MAC Address.

10.3.aa74.2 MLME-APPEERKEY.confirm

10.3.aa74.2.1 Function

This primitive reports the result of a request to send a Public Key Public Action frame.

10.3.aa74.2.2 Semantics of the service primitive

The primitive parameters are as follows:

MLME-APPEERKEY.confirm (

Result Code

)

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Valid range | Description |
| Result Code | Enumeration | SUCCESS, INVALID PARAMETERS or UNSPECIFIED FAILURE | Reports the outcome of a  request to send a Public Key |

10.3.aa74.2.3 When Generated

This primitive is generated by the MLME when the request to transmit a Public Key Public Action frame completes.

10.3.aa74.2.4 Effect of Receipt

On receipt of this primitive, the SME evaluates the result code.

10.3.aa74.3 MLME-APPEERKEY.indication

10.3.aa74.3.1 Function

This primitive indicates that a Public Key Public Action frame has been received from a peer entity.

10.3.aa74.3.2 Semantics of the service primitive

The primitive parameters are as follows:

MLME-APPEERKEY.indication (

Peer MAC Address,

Request Type,

Group,

Public Key

)

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Type | Valid range | Description |
| Peer MAC Address | MACAddress | Any valid individual MACAddress | The address of the peer MAC entity from which the Public Key Public Action frame has been received. |
| Request Type | Integer | As defined in | Specifies the type of request |
| Group | Finite Cyclic Group field | As defined in 7.3.1.40 | Specifies cyclic group from which the public key was generated |
| Public Key | Scalar field | As defined 7.3.1.37 | The Public Key of the AP that sent this Public Key Public Action frame |

10.3.aa74.3.3 When Generated

This primitive is generated by the MLME when a valid Public Key Public Action frame is received.

10.3.aa74.3.4 Effect of Receipt

On receipt of this primitive, the SME performs the behavior defined in

11. MLME

11.aa24 Procedures to Manage Overlapping BSS (OBSS)

11.aa24.2 HCCA TXOP Advertisement

***Modify the 4th paragraph of 11.aa24.2 as shown.***

If an AP for which dot11PublicTXOPNegotiationActivated is true receives an HCCA TXOP Response Public Action frame with the status field set to <ANA> (“The TS ~~should not be created because the~~ schedule conflicts with an existing schedule…”) the AP should create a new schedule for the TSPEC request using the suggestion provided in the HCCA TXOP Response frame. If an AP for which dot11ProtectedTXOPNegotiationActivated is true receives a Protected HCCA TXOP Response Protected Dual of Public Action frame with the status field set to <ANA> (“The TS schedule conflicts with an existing schedule…”) the AP should create a new schedule for the TSPEC request using the suggestion provided in the HCCA TXOP Response frame. This allows HCCA APs to cooperatively create new HCCA schedules ~~within a beacon period~~ that do not collide. Failure of an AP to use the information in a HCCA TXOP Response frame when scheduling a HCCA TXOP may lead to contention free period collisions with an overlapping HCCA AP.

***Modify 11.aa24.3 as shown.***

11.aa24.3 HCCA TXOP Negotiation

An AP for which ~~dot11RobustAVStreamingImplemented~~ dot11PublicTXOPNegotiationActivated is true or dot11ProtectedTXOPNegotiationActivated is true shall be able to maintain an avoidance TXOP Reservation for each overlapping HCCA AP. This avoidance TXOP is a period of time that the AP should try to avoid using when creating schedules for new TS requests.

An AP for which dot11PublicTXOPNegotiationActivated is false shall discard any received HCCA TXOP Advertisement Public Action frames.

Upon reception of an HCCA TXOP Advertisement Public Action frame, an AP for which ~~dot11RobustAVStreamingImplemented~~ dot11PublicTXOPNegotiationActivated is true shall discard any records for the AP that sent the HCCA TXOP Advertisement frame and shall prepare a response using the procedures below.

An AP for which dot11ProtectedTXOPNegotiationActivated is false shall discard any received Protected HCCA TXOP Advertisement Protected Dual of Public Action frames.

An AP for which dot11ProtectedTXOPNegotiationActivated is true that does not have an active security association with the peer AP shall use the AP PeerKey Protocol (as defined in 8.2aa.2) and the Authenticated Mesh Peering Exchange (as defined in 11C.5) to negotiate security parameters and create a new SMKSA and STKSA to secure the Protected HCCA TXOP Advertisement Protected Dual of Public Action frames. The use of the Authenticated Mesh Peering Exchange proves possession of the PMK (generated using the procedures described in ) and implicitly the private key that corresponds to the peer’s public key.

Upon reception of a valid Protected HCCA TXOP Advertisement Protected Dual of Public Action frame, an AP for which dot11ProtectedTXOPNegotiationActivated is true shall discard any records for the AP that sent the Protected HCCA TXOP Advertisement frame and shall prepare a response using the procedures below.

If the HCCA TXOP Advertisement frame (either protected or public) has not been discarded due to the procedures above, the ~~The~~ AP shall inspect its HCCA schedule to check if the TXOP given in the HCCA TXOP Advertisement frame is in conflict with an existing accepted HCCA TXOP. If there is no conflict, the AP shall send an HCCA TXOP Response frame with the status field set to 0 (“Successful”) and add the schedule given in the HCCA TXOP Advertisement frame to the list of time periods to avoid when scheduling new HCCA TXOPs.

If the HCCA Advertisement was sent using a Protected Dual of Public Action frame, the HCCA TXOP Response shall be sent using a Protected HCCA TXOP Response Protected Dual of Public Action frame.

If the AP detects that the TXOP given in the HCCA TXOP Advertisement frame is in conflict with an existing accepted HCCA TXOP and this AP is not itself in the process of processing an ADDTS request, it shall send a (Protected) HCCA TXOP Response frame with the status field set to <ANA> (“The TS ~~should not be created because the~~ schedule conflicts with an existing schedule…”) and the Alternate Schedule field set to a period of time that does not conflict with any currently accepted HCCA TXOPs and the Avoidance Request field absent. The duration sub-field of the Alternate Schedule field should be greater than or equal to the duration sub-field of the schedule field in the (Protected) HCCA TXOP Advertisement frame.

If the AP detects that the TXOP given in the (Protected) HCCA TXOP Advertisement frame is in conflict with an in-progress ADDTS request for a HCCA TXOP for which TXOP Response frames have not been received, it shall send a (Protected) HCCA TXOP Response frame with the status field set to <ANA> (“The TS ~~should not be created because the~~ schedule conflicts with an existing schedule…”) with the Alternate Schedule and Avoidance Request fields set according to the following rules:

* If *MIX*(MACr) < *MIX*(MACi) ~~the MAC address of the AP that received the TXOP Advertisement frame is less than the MAC address of the AP that sent the TXOP Advertisement frame,~~ the Alternate Schedule field is set to a value that does not conflict with any accepted HCCA TXOPs and also does not conflict with the TXOP of the in-progress ADDTS request. The Avoidance Request field is set to the TXOP of the in-progress ADDTS request.
* If *MIX*(MACr) > *MIX*(MACi) ~~the MAC address of the AP that received the TXOP Advertisement frame is greater than the MAC address of the AP that sent the TXOP Advertisement frame,~~ the Alternate Schedule field is set to the value from the TXOP Reservation from the TXOP Advertisement frame. The Avoidance Request field is set to a time period that does not conflict with any accepted HCCA TXOPs nor the TXOP in the Alternate Schedule field and has sufficient duration and service interval to meet the requirements of the in-progress ADDTS request.

Where:

MACr is the MAC address of the AP that received the TXOP Advertisement frame

MACi is the MAC address of the AP that sent the TXOP Advertisement frame

The *MIX* function takes the 6 octets of a MAC address and computes a new 6 octet value:

*MIX*(MAC) = MAC[4] || MAC[5] || MAC[0] || MAC[1] || MAC[2] || MAC[3]

Table 11-2aa—Contents of TXOP Response Frame

|  |  |  |  |
| --- | --- | --- | --- |
| **Case** | **Status Code** | **Alternate Schedule Field** | **Avoidance Request Field** |
| No conflict with existing or in-progress schedules | “OK” | Not present | Not Present |
| Conflicts with existing schedule, no ADDTS request in progress | “The TS ~~should not be created because the~~ schedule conflicts with an existing or in-progress schedule…” | Period of time that does not conflict with any currently accepted HCCA TXOPs | Not Present |
| Conflict in-progress schedules, ~~RA[[1]](#footnote-2) < TA[[2]](#footnote-3)~~ *MIX*(MACr)<MIX(MACi) | “The TS ~~should not be created because the~~ schedule conflicts with an existing or in-progress schedule…” | Period of time that does not conflict with any currently accepted HCCA TXOPs nor the in-progress ADDTS request | Schedule of in-progress ADDTS request |
| Conflict in-progress schedules, ~~RA > TA~~ *MIX*(MACr)>MIX(MACi) | “The TS ~~should not be created because the~~ schedule conflicts with an existing or in-progress schedule…” | Same schedule that was in the TXOP Advertisement | Period of time that does not conflict with any currently accepted HCCA TXOPs nor the period given in the Alternate Schedule field |

The AP shall keep a record of the TXOP proposed in the alternate schedule field in a TXOP avoidance record and should avoid scheduling any new HCCA TXOPs in this proposed period until any of the following conditions occurs:

* A period of ~~for~~ dot11HCCATXOPBeaconTimeout multiplied by dot11BeaconPeriod TUs has elapsed
* ~~or until it~~ The AP has dot11PublicTXOPNegotiationActivated set to true and receives a~~nother~~ HCCA TXOP Advertisement Public Action frame from the AP to which the HCCA TXOP Response frame was sent
* The AP has dot11ProtectedTXOPNegotiationActivated set to true and receives a Protected HCCA TXOP Advertisement Protected Dual of Public Action frame from the AP to which the Protected HCCA TXOP Response frame was sent

Annex D

**ASN.1 encoding of the MAC and PHY MIB**

***Add dot11ProtectedHCCATXOPNegotiationImplemented object to annex D just before dot11HCCATXOPNegotiationActivated as shown and assign it a dot11StationConfigEntry number.***

dot11PublicHCCATXOPNegotiationImplemented OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

“This is a capability variable.

Its value is determined by device capabilities.

This attribute, when TRUE, indicates that the station implementation

supports the negotiation of HCCA TXOPs using public action frames.”  
 DEFVAL { true }

::= { dot11StationConfigEntry aa?? }

***Modify dot11HCCATXOPNegotiationActivated object in annex D as shown.***

dot11PublicHCCATXOPNegotiationActivated OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-~~only~~write

STATUS current

DESCRIPTION

“This is a control variable.

It is written by the SME or external management entity.

Changes take effect for the next MLME-START.request primitive

This attribute, when TRUE, indicates that the AP

can negotiate HCCA TXOPs using public action frames.”  
 DEFVAL { true }

::= { dot11StationConfigEntry aa17 }

***Add dot11ProtectedHCCATXOPNegotiationImplemented object to annex D as shown and assign it a dot11StationConfigEntry number.***

dot11ProtectedHCCATXOPNegotiationImplemented OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-only

STATUS current

DESCRIPTION

“This is a capability variable.

Its value is determined by device capabilities.

This attribute, when TRUE, indicates that the station implementation

supports the negotiation of HCCA TXOPs using protected dual of public

action frames.”  
 DEFVAL { true }

::= { dot11StationConfigEntry aa?? }

***Add dot11ProtectedHCCATXOPNegotiationActivated object to annex D as shown and assign it a dot11StationConfigEntry number.***

dot11ProtectedHCCATXOPNegotiationActivated OBJECT-TYPE

SYNTAX TruthValue

MAX-ACCESS read-write

STATUS current

DESCRIPTION

“This is a control variable.

It is written by the SME or external management entity.

Changes take effect for the next MLME-START.request primitive

This attribute, when TRUE, indicates that the AP

can negotiate HCCA TXOPs using protected dual of public action frames.”  
 DEFVAL { true }

::= { dot11StationConfigEntry aa?? }

**References:**

1. ~~RA is the MAC address of the STA that received the TXOP Advertisement frame~~ [↑](#footnote-ref-2)
2. ~~TA is the MAC address of the STA that sent the TXOP Advertisement frame~~ [↑](#footnote-ref-3)