The following text describes changes that need to be made to Annex D of 802.1X-2001 in order to align it with the latest IETF draft on 802.1X and RADIUS guidelines (draft-congdon-radius-8021x-20.txt). This document should be published as an informational RFC soon.

In the following sections, a description of the change is provided as well as instructions on how to update Annex D. New text extracted from the IETF draft is included in a different font (Courier New). This text may need further updates to align it with IEEE editor's conventions and references.

D.1 Introduction

There has been a minor expansion of the opening paragraph to include further discussion on stand-alone mode and generalization about AAA clients. Consider replacing the 1st paragraph with the following three paragraphs.

IEEE 802.1X provides "network port authentication" for IEEE 802 media, including Ethernet, Token Ring and 802.11 wireless LANS.

IEEE 802.1X does not require use of a backend authentication server, and thus can be deployed with stand-alone switches or access points, as well as in centrally managed scenarios.

In situations where it is desirable to centrally manage authentication, authorization and accounting (AAA) for IEEE 802 networks, deployment of a backend authentication and accounting server is desirable. In such situations, it is expected that IEEE 802.1X Authenticators will function as AAA clients.

D.2.1 Acct-Terminate-Cause

A discussion periodic re-authentication was added just after the paragraph discussing the re-authentication failure (20) termination cause. Insert the following paragraphs before the paragraph discussing Admin Reset (6).

Within 802.11 [22], periodic re-authentication may be useful in preventing reuse of an initialization vector with a given key. Since successful re-authentication does not result in termination of the session, accounting packets are not sent as a result of re-authentication unless the status of the session changes. For example:

a. The session is terminated due to re-authentication failure. In this case the Reauthentication Failure (20) termination cause is used.
b. The authorizations are changed as a result of a successful re-authentication. In this case, the Service Unavailable (15) termination cause is used. For accounting purposes, the portion of the session after the authorization change is treated as a separate session.

Where IEEE 802.1X authentication occurs prior to 802.11 association, accounting packets are not sent until an association occurs.

D.2.2 Acct-Multi-Session-Id

Clarification updates in the second paragraph. Replace the second paragraph with the following paragraph.

Where supported by the Access Points, the Acct-Multi-Session-Id attribute is used to link together the multiple related sessions of a roaming Supplicant. In such a situation, if the session context is transferred between access points, accounting packets may be sent without a corresponding authentication and authorization exchange. However, in such a situation it is assumed that the Acct-Multi-Session-Id is transferred between the Access Points as part of the Inter-Access Point Protocol.

Replace the 4th paragraph with the following text.

As a result, the Acct-Multi-Session-Id attribute is unique among all the Access Points, Supplicants and sessions. In order to provide this uniqueness, it is suggested that the Acct-Multi-Session-Id be of the form:

An additional paragraph on encoding the Acct-Muilt-Session-Id was added at the end. Add the following paragraph to the end of section D.2.2

Since the Acct-Multi-Session-Id is of type String as defined in [RFC2866], for use with IEEE 802.1X, it is encoded as an ASCII string of Hex digits. Example: "00-10-A4-23-19-C0-00-12-B2-14-23-DE-AF-23-83-C0-76-B8-44-E8"

D.3 RADIUS authentication

In the opening paragraph, additional references are made to new documents. These will need to be augmented in the reference section of 802.1X as well. Replace the first paragraph with the following paragraph:

This section describes how attributes defined in [RFC2865], [RFC2867], [RFC2868], [RFC2869] and [RFC3162] are used in IEEE 802.1X authentication.

The list of items following the first paragraph has been removed in the IETF draft in favor of the summary section at the end. This summary section is not included in IEEE 802.1X, but should now be. Delete the list of items after the first paragraph.

D.3.1 User-Name

An additional reference is added to the end of the first paragraph. This reference needs to be added to the end of 802.1X as well. The new reference is RFC 2869. Thus replace the last sentence of the first paragraph with:

Where available, the Supplicant identity is included in the User-Name attribute, and included in the RADIUS Access-Request and Access-Reply messages as specified in [RFC2865] and [RFC2869].

D.3.3 NAS-IP-Address

Add NAS-IPv6-Address to the clause heading, and replace the paragraph with the following text:

For use with IEEE 802.1X, the NAS-IP-Address contains the IPv4 address of the bridge or Access Point acting as an Authenticator, and the NAS-IPv6-Address contains the IPv6 address. If the IEEE 802.1X authenticator has more than one interface, it may be desirable to use a loopback address for this purpose so that the Authenticator will still be reachable even if one of the interfaces were to fail.

D.3.4 NAS-Port

Additional description of how to handle the case where authentication occurs prior to association. Replace the paragraph with the following text:

For use with IEEE 802.1X the NAS-Port will contain the port number of the bridge, if this is available. While an 802.11 Access Point does not have physical ports, it does assign a unique "association ID" to every mobile station upon a successful association exchange. As a result, for an 802.11 Access Point, if the association exchange has been completed prior to authentication, the NAS-Port attribute will contain the association ID, which is a 16-bit unsigned integer.

D.3.7 Framed-IP-Address, Framed-IP-Netmask

Clarification about IP address assignment and what types of devices might support this attribute has been added. Replace the paragraph with the following text:

IEEE 802.1X does not provide a mechanism for IP address assignment. Therefore the Framed-IP-Address and Framed-IP-Netmask attributes can only be used by IEEE 802.1X Authenticators that support IP address assignment mechanisms. Typically this capability is supported by layer 3 devices.

D.3.9 Filter-ID

Clarification about what type of devices might support this attribute has been added. Replace the paragraph with the following text:

This attribute indicates the name of the filter list to be applied to the Supplicant's session. For use with an IEEE 802.1X Authenticator, it may be used to indicate either layer 2 or layer 3 filters. Layer 3 filters are typically only supported on IEEE 802.1X Authenticators that act as layer 3 devices.

D.3.12 Reply-Message

This section has been completely re-written and re-named to deal with a number of deployment issues. Rename this section “Displayable Messages” and replace the text with the following paragraphs:

The Reply-Message attribute, defined in section 5.18 of [RFC2865], indicates text which MAY be displayed to the user. This is similar in concept to the EAP Notification Type, defined in [RFC2284]. When sending a displayable message to an IEEE 802.1X Authenticator, the RADIUS server SHOULD encapsulate displayable messages within EAP-Message/EAP-Request/Notification attribute(s), and SHOULD NOT use Reply-Message attribute(s) for this purpose.

An IEEE 802.1X Authenticator receiving Reply-Message attribute(s) MAY copy the Text field(s) into the Type-Data field of an EAP-Request/Notification packet, fill in the Identifier field, and send this to the Peer. However, several issues may arise from this:

a) Unexpected Responses. On receiving an EAP-Request/Notification, the Supplicant will send an EAP-Response/Notification, and the Authenticator will pass this on to the RADIUS server, encapsulated within EAP-Message attribute(s). However, the RADIUS server may not be expecting an Access-Request containing an EAP-Message/EAP-Response/Notification attribute.

   For example, consider what happens when a Reply-Message is included within an Access-Accept or Access-Reject packet with no EAP-Message attribute present. If the value of the Reply-Message attribute is copied into the Type-Data of an EAP-Request/Notification and sent to the peer, this will result in an Access-Request containing an EAP-Message/EAP-Response/Notification attribute being sent by the Authenticator to the RADIUS server. Since an Access-Accept or Access-Reject packet terminates the RADIUS conversation, such an Access-Request would not be expected.

b) Identifier conflicts. While the EAP-Request/Notification contains an an Identifier, a Reply-Message attribute does not. As a result, an Authenticator receiving a Reply-Message attribute and wishing to translate this to an EAP-Request/Notification will need to choose an Identifier. It is possible that the chosen Identifier will conflict with a value chosen by the RADIUS server for another packet within the EAP conversation. This would violate the requirement that Identifier values be unique within an EAP conversation.

D.3.14 Framed-Route

Add NAS-IPv6-Route to the clause heading, and replace the paragraph with the following text:

The Framed-Route and Framed-IPv6-Route attributes provide routes that are to be configured for the supplicant. These attributes are therefore only relevant for IEEE 802.1X Authenticators that act as layer 3 devices, and cannot be understood by a bridge or Access Point.

New Section D.3.16 Vendor Specific

A new section has been inserted, so the section numbering after this point will need to be adjusted. I'll keep using the original numbering below so changes can be found by referencing the original 802.1X-2001 document. This new section describes some Vendor Specific attributes that are critical to making WEP keys work. Insert this new section with the following text:

Vendor-specific attributes are used for the same purposes as described in [RFC2865]. The MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes, described in section 2.4 of [RFC2548], MAY be used to encrypt and authenticate the RC4 EAPOL-Key descriptor [IEEE8021X, Section 7.6]. Examples of the derivation of the MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes from the master secret negotiated by an EAP method are given in [RFC2716]. Details of the EAPOL-Key descriptor are provided in Section D.4.

D.3.16 Session-Timeout

This section appears to have undergone a considerable re-write. The first sentence also appears to be one that was added by 802.1X and may remain if desired. The next two paragraphs should be replaced with the following text:

When sent along in an Access-Accept without a Termination-Action attribute or with a Termination-Action attribute set to Default, the Session-Timeout attribute specifies the maximum number of seconds of service provided prior to session termination.

When sent in an Access-Accept along with a Termination-Action value of RADIUS-Request, the Session-Timeout attribute specifies the maximum number of seconds of service provided prior to re-authentication. In this case, the Session-Timeout attribute is used to load the reAuthPeriod constant within the Reauthentication Timer state machine of 802.1X. When sent with a Termination-Action value of RADIUS-Request, a Session-Timeout value of zero indicates the desire to perform another authentication (possibly of a different type) immediately after the first authentication has successfully completed.

As described in [6], when sent in an Access-Challenge, this attribute represents the maximum number of seconds that an IEEE 802.1X authenticator should wait for an EAP-Response before retransmitting. In this case, the Session-Timeout attribute is used to load the suppTimeout constant within the Backend state machine of IEEE 802.1X.

D.3.17 Idle-Timeout

A very minor change in the first sentence to reference the RADIUS RFC has been added. Insert the following sentence at the beginning of the first paragraph and fix the reference appropriately:

The Idle-Timeout attribute is described in [4].

D.3.18 Terminate-Action

The second sentence has been re-written and additional information about how to terminate the session when a default value is specified has been added. Replace all of the text in this clause with the following paragraph:

This attribute indicates what action should be taken when the service is completed. The value RADIUS-Request(1) indicates that re-authentication should occur on expiration of the Session-Time. The value Default (0) indicates that the session should terminate.

D.3.19 Called-Station-Id

A description of a special case for 802.11 access points has been added. The text in 802.1X differs from the original IETF draft because 802.1X references IEEE Std 802 for a description of Canonical format. This part of the text should remain, and if possible the IETF draft should be updated before going to RFC. Replace the text in this clause with the following paragraph

For IEEE 802.1X authenticators, this attribute is used to store the bridge or Access Point MAC address, represented as an ASCII character string in Canonical format (see IEEE Std 802). For Example: "00-10-A4-23-19-C0". For 802.11 Access Points, the 802.11 SSID SHOULD be appended to the Access Point MAC address, separated from the MAC address with a ":". Example "00-10-A4-23-19-C0:AP1".

D.3.29 Framed-Pool

Add Framed-IPv6-Pool to the clause heading and replace the paragraph with the following text:

IEEE 802.1X does not provide a mechanism for IP address assignment. Therefore the Framed-Pool and Framed-IPv6-Pool attributes can only be used by IEEE 802.1X Authenticators that support IP address assignment mechanisms. Typically this capability is supported by layer 3 devices

D.3.30 Tunnel Attributes

A fair amount of re-write has taken place starting at the second paragraph. Replace all the text after the first paragraph with the following:

In particular, it may be desirable to allow a port to be placed into a particular Virtual LAN (VLAN), defined in [IEEE8021Q], based on the result of the authentication. This can be used, for example, to allow a wireless host to remain on the same VLAN as it moves within a campus network.

The RADIUS server typically indicates the desired VLAN by including tunnel attributes within the Access-Accept. However, the IEEE 802.1X Authenticator may also provide a hint as to the VLAN to be assigned to the Supplicant by including Tunnel attributes within the Access-Request.

For use in VLAN assignment, the following tunnel attributes are used:

Tunnel-Type=VLAN (13)
Tunnel-Medium-Type=802
Tunnel-Private-Group-ID=VLANID

Note that the VLANID is 12-bits, taking a value between 1 and 4094, inclusive. Since the Tunnel-Private-Group-ID is of type String as defined in [RFC2868], for use with IEEE 802.1X, the VLANID is encoded as a string, rather than an integer.

When Tunnel attributes are sent, it is necessary to fill in the Tag field. As noted in [RFC2868], section 3.1:

> means of grouping attributes in the same packet which refer to the same tunnel. Valid values for this field are 0x01 through 0x1F, inclusive. If the Tag field is unused, it MUST be zero (0x00).

For use with Tunnel-Client-Endpoint, Tunnel-Server-Endpoint, Tunnel-Private-Group-ID, Tunnel-Assignment-ID, Tunnel-Client-Auth-ID or Tunnel-Server-Auth-ID attributes (but not Tunnel-Type, Tunnel-Medium-Type, Tunnel-Password, or Tunnel-Preference), a tag field of greater than 0x1F is interpreted as the first octet of the following field.

Unless alternative tunnel types are provided, (e.g. for IEEE 802.1X Authenticators that may support tunneling but not VLANs), it is only necessary for tunnel attributes to specify a single tunnel. As a result where it is only desired to specify the VLANID, the tag field SHOULD be set to zero (0x00) in all Tunnel attributes. Where alternative tunnel types are to be provided, tag values between 0x01 and 0x1F SHOULD be chosen.

D.4 RC4 EAPOL-Key Frame

An entire new section has been added to the I-D to better describe the RC4 EAPOL-Key frame and how it is used. Insert this new section into the document and renumber the subsequent sections as necessary:

The RC4 EAPOL-Key frame is created and transmitted by the Authenticator in order to provide media specific key information. For example, within 802.11 the RC4 EAPOL-Key frame can be used to distribute multicast/broadcast ("default") keys, or unicast ("key mapping") keys. The "default" key is the same for all stations within a broadcast domain.

The RC4 EAPOL-Key frame is not acknowledged and therefore the Authenticator does not know whether the Supplicant has received it. If it is lost, then the Supplicant and Authenticator will not have the same keying material, and communication will fail. If this occurs, the problem is typically addressed by re-running the authentication.

The RC4 EAPOL-Key frame is sent from the Authenticator to the Supplicant in order to provision the "default" key, and subsequently in order to refresh the "default" key. It may also be used to refresh the key-mapping key. Note that rekey is typically only required with weak ciphersuites such as WEP, defined in [IEEE80211].

Where keys are required, an EAP method that derives keys is typically selected. Therefore the initial "key mapping" keys can be derived from EAP keying material, without requiring the Authenticator to send an RC4 EAPOL-Key frame to the Supplicant. An example of how EAP keying material can be derived and used is presented in [RFC2716].

As described in the paragraphs that follow, the MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes are defined from the point of view of the Authenticator. From the Supplicant point of reference, the terms are reversed. Thus the MS-MPPE-Recv-Key on the Supplicant corresponds to the MS-MPPE-Send-Key on the Authenticator, and the MS-MPPE-Send-Key on the Supplicant corresponds to the MS-MPPE-Recv-Key on the Authenticator.

While the RC4 EAPOL-Key frame is defined in IEEE 802.1X-2001, a more complete description is provided here as follows:

```
 0                   1                   2                   3
 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1 2 3 4 5 6 7 8 9 0 1
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|    Version    |  Packet Type  |  Packet Body Length           |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|     Type      |           Key  Length         |Replay Counter...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Replay Counter...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Replay Counter    |   Key IV...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Key IV...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Key IV...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Key IV...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Key IV...         |F| Key Index   |
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Key Signature...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Key Signature...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Key Signature...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Key Signature...
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
|                             Key...
```

```
+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+-+
```

Version

The Version field is one octet. For [IEEE8021X] it contains the value 0x01.

Packet Type

The Packet Type field is one octet, and determines the type of packet being transmitted. For an EAPOL-Key Descriptor, the Packet Type field contains 0x03.

Packet Body Length

The Packet Body Length is two octets, and contains the length of the EAPOL-Key descriptor in octets, not including the Version, Packet Type and Packet Body Length fields.

Type

The Type field is a single octet. The Key descriptor is defined differently for each Type; this specification documents only the RC4 Key Descriptor (Type = 0x01).

Key Length

The Key Length field is two octets. If Packet Body Length = 44 + Key Length, then the Key Field contains the key in encrypted form, of length Key Length. This is 5 octets (40 bits) for WEP, and 13 octets (104 bits) for WEP-128. If Packet Body Length = 44, then the Key field is absent, and Key Length represents the number of least significant octets from the MS-MPPE-Send-Key attribute to be used as the keying material.

Replay Counter

The Replay Counter field is 8 octets. It does not repeat within the life of the keying material used to encrypt the Key field and compute the Key Signature field. A 64-bit NTP timestamp MAY be used as the Replay Counter.

Key IV

The Key IV field is 16 octets and includes a 128-bit cryptographically random number.

F

The Key flag (F) is a single bit, describing the type of key that is included in the Key field. Values are:

0 = for broadcast (default key)
1 = for unicast (key mapping key)

Key Index

The Key Index is 7 bits.

Key Signature

The Key Signature field is 16 octets. It contains an HMAC-MD5 message integrity check computed over the EAPOL-Key descriptor, starting from the Version field, with the Key field filled in if present, but with the Key Signature field set to zero. For the computation, the 32 octet (256 bit) MS-MPPE-Send-Key is used as the HMAC-MD5 key.

Key

If Packet Body Length = 44 + Key Length, then the Key Field contains the key in encrypted form, of length Key Length. If Packet Body Length = 44, then the Key field is absent, and the least significant Key Length octets from the MS-MPPE-Send-Key attribute is used as the keying material. Where the Key field is encrypted using RC4, the RC4 encryption key used to encrypt this field is formed by concatenating the 16 octet (128 bit) Key-IV field with the 32 octet MS-MPPE-Recv-Key attribute. This yields a 48 octet RC4 key (384 bits).

D.4 Security considerations

This section has been almost completely re-written. Save the first paragraph, but everything after that replace with the following text and new subsections:

Vulnerabilities include:

a) Packet modification or forgery
b) Dictionary attacks
c) Known plaintext attacks
d) Replay
e) Outcome mismatches
f) 802.11 integration
g) Key management issues

D.5.1. Packet modification or forgery

RADIUS, defined in [RFC2865], does not require all Access-Requests to be authenticated or integrity protected. However, IEEE 802.1X is based on EAP, and as described in [RFC2869]:

> The Message-Authenticator attribute MUST be used to protect all Access-Request, Access-Challenge, Access-Accept, and Access-Reject packets containing an EAP-Message attribute.

As a result, when used with IEEE 802.1X, all RADIUS packets are authenticated and integrity protected.

D.5.2. Dictionary attacks

The RADIUS shared secret is vulnerable to offline dictionary attack, based on capture of the Response Authenticator or Message-Authenticator attribute. In order to decrease the level of vulnerability, [RFC2865] recommends:

> The secret (password shared between the client and the RADIUS server) SHOULD be at least as large and unguessable as a well-chosen password. It is preferred that the secret be at least 16 octets.

In addition, the risk of an offline dictionary attack can be further mitigated by employing IPsec ESP with non-null transform in order to encrypt the RADIUS conversation, as described in [RFC3162].

D.5.3. Known plaintext attacks

Since IEEE 802.1X is based on EAP, which does not support PAP, the RADIUS User-Password attribute is not used to carry hidden user passwords. The hiding mechanism utilizes MD5, defined in [RFC1321], in order to generate a key stream based on the RADIUS shared secret and the Request Authenticator. Where PAP is in use, it is possible to collect key streams corresponding to a given Request Authenticator value, by capturing RADIUS conversations corresponding to a PAP authentication attempt using a known password. Since the User-Password is known, the key stream corresponding to a given Request Authenticator can be determined and stored.

Since the key stream may have been determined previously from a known plaintext attack, if the Request Authenticator repeats, attributes encrypted using the RADIUS attribute hiding mechanism should be considered compromised. In addition to the User-Password attribute, this includes attributes such as Tunnel-Password [RFC2868, section 3.5] and MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes [RFC2548, section 2.4], which include a Salt field as part of the hiding algorithm. To avoid this, [RFC2865] advises:

> Since it is expected that the same secret MAY be used to authenticate with servers in disparate geographic regions, the Request Authenticator field SHOULD exhibit global and temporal uniqueness.

Where the Request Authenticator repeats, the Salt field defined in [RFC2548] does not provide protection against compromise. This is because MD5 [RFC1321], rather than HMAC-MD5 [RFC2104], is used to generate the key stream, which is calculated from the 128-bit RADIUS shared secret (S), the 128-bit Request Authenticator (R), and the Salt field (A), using the formula b(1) = MD5(S + R + A). Since the Salt field is placed at the end, if the Request Authenticator were to repeat on a network where PAP is in use, then the salted keystream could be calculated from the User-Password keystream by continuing the MD5 calculation based on the Salt field (A), which is sent in the clear.

Even though IEEE 802.1X Authenticators do not support PAP authentication, a security vulnerability can still exist where the same RADIUS shared secret is used for hiding User-Password as well as other attributes. This can occur, for example, if the same RADIUS proxy handles authentication requests for both IEEE 802.1X (which may hide the Tunnel-Password, MS-MPPE-Send-Key and MS-MPPE-Recv-Key attributes) and GPRS (which may hide the User-Password attribute).

The threat can be mitigated by protecting RADIUS with IPsec ESP with non-null transform, as described in [RFC3162]. In addition, the same RADIUS shared secret MUST NOT used for both IEEE 802.1X authentication and PAP authentication.

D.5.4. Replay

The RADIUS protocol provides only limited support for replay protection. Replay protection for RADIUS authentication and accounting can be provided by enabling IPsec replay protection with RADIUS, as described in [RFC3162].

RADIUS Access-Requests include liveness via the 128-bit Request Authenticator. However, the Request Authenticator is not a replay counter. Since RADIUS servers may not maintain a cache of previous Request Authenticators, the Request Authenticator does not provide replay protection.

RADIUS accounting [RFC2866] does not support replay protection at the protocol level. Due to the need to support failover between RADIUS accounting servers, protocol-based replay protection is not sufficient to prevent duplicate accounting records. However, once accepted by the accounting server, duplicate accounting records can be detected by use of the the Acct-Session-Id [RFC2866, section 5.5] and Event-Timestamp [RFC2869, section 5.3] attributes.

Unlike RADIUS authentication, RADIUS accounting does not use the Request Authenticator as a nonce. Instead, the Request Authenticator contains an MD5 hash calculated over the Code, Identifier, Length, and request attributes of the Accounting Request packet, plus the shared secret. The Response Authenticator also contains an MD5 hash calculated over the Code, Identifier and Length, the Request Authenticator field from the Accounting-Request packet being replied to, the response attributes and the shared secret.

Since the Accounting Response Authenticator depends in part on the Accounting Request Authenticator, it is not possible to replay an Accounting-Response unless the Request Authenticator repeats. While it is possible to utilize EAP methods such as EAP TLS [RFC2716] which include liveness checks on both sides, not all EAP messages will include liveness so that this provides incomplete protection.

As with the Request Authenticator, for use with IEEE 802.1X Authenticators, the Acct-Session-Id SHOULD be globally and temporally unique.

D.5.5. Outcome mismatches

[RFC2869] does not require that the EAP packet encapsulated in an EAP-Message attribute agree with the outcome of the authentication, or even that an EAP-Message attribute be included in an Access-Accept or Access-Reject. For example, an EAP-Success can be encapsulated in an Access-Reject, or an EAP-Failure can be encapsulated within an Access-Accept. Neither message should be encapsulated in an Access-Challenge because as described in [RFC2284], EAP-Success and EAP-Failure messages are not ACK'd. Since an Access-Challenge indicates a continuing EAP conversation and no client response is expected to these messages, encapsulating these messages within an Access-Challenge would constitute a contradiction.

To address the possible corner conditions and ensure that access decisions made by IEEE 802.1X Authenticators conform to the wishes of the RADIUS server, it is necessary for the Authenticator to make the decision solely based on the authentication result (Accept/Reject) and NOT based on the contents of the EAP packet encapsulated in one or more EAP-Message attributes, if one is present at all.

D.5.6. 802.11 integration

[IEEE8021X] was developed for use on wired IEEE 802 networks such as Ethernet, and therefore does not describe how to securely adapt IEEE 802.1X for use with 802.11. This is left to the enhanced security specification under development within IEEE 802.11.

For example, [IEEE8021X] does not specify whether authentication occurs prior to, or after, association, nor how the derived keys can be used to integrity protect management frames. It also does not specify ciphersuites addressing the vulnerabilities discovered in WEP, described in [Berkeley], [Arbaugh], [Fluhrer], and [Stubbl]. [IEEE8021X] only defines an authentication framework, leaving the definition of the authentication methods to other documents, such as [RFC2716].

Since [IEEE8021X] does not address 802.11 integration issues, implementors are strongly advised to consult the IEEE 802.11 enhanced security specification for guidance on how to adapt IEEE 802.1X for use with 802.11. For example, it is likely that the IEEE 802.11 enhanced security specification will define its own IEEE 802.11 key hierarchy as well as new EAPOL-Key descriptors.

D.5.7. Key management issues

The EAPOL-Key descriptor described in Section 4 is likely to be deprecated in the future, when the 802.11 enhanced security group completes its work. Known security issues include:

1. Default key-only support. IEEE 802.1X enables the derivation of per-station unicast keys, known in [IEEE80211] as "key mapping keys." Keys used to encrypt multicast/broadcast traffic are known as "default keys". However, in some 802.11 implementations, the unicast keys derived as part of the EAP authentication process are used solely in order to encrypt, authenticate and integrity protect the EAPOL-Key descriptor, as described in Section 4. These implementations only support use of default keys (ordinarily only used with multicast/broadcast traffic) to secure all traffic, unicast or multicast/broadcast, resulting in inherent security weaknesses.

   Where per-station key-mapping keys (e.g. unicast keys) are unsupported, any station possessing the default key can decrypt traffic from other stations or impersonate them. When used along with a weak cipher (e.g. WEP), implementations supporting only default keys provide more material for attacks such as those described in [Fluhrer] and [Stubbl]. If in addition the default key is not refreshed periodically, IEEE 802.1X dynamic key derivation provides little or no security benefit. For an understanding of the issues with WEP, see [Berkeley], [Arbaugh], [Fluhrer], and [Stubbl].

2. Reuse of keying material. The EAPOL-Key descriptor specified in section 4 uses the same keying material (MS-MPPE-Recv-Key) both to encrypt the Key field within the EAPOL-Key descriptor, as well as to encrypt data passed between the station and access point. Multi-purpose keying material is frowned upon, since multiple uses can leak information helpful to an attacker.

3. Weak algorithms. The algorithm used to encrypt the Key field within the EAPOL-Key descriptor is similar to the algorithm used in WEP, and as a result, shares some of the same weaknesses. As with WEP, the RC4 stream cipher is used to encrypt the key. As input to the RC4 engine, the IV and key are concatenated rather than being combined within a mixing function. As with WEP, the IV is not a counter, and therefore there is little protection against reuse.

As a result of these vulnerabilities, implementers intending to use the EAPOL-Key descriptor described in this document are urged to consult the 802.11 enhanced security specification for a more secure alternative. It is also advisable to consult the evolving literature on

WEP vulnerabilities, in order to better understand the risks, as well

obtain guidance on setting an appropriate re-keying interval.

D.6 IANA Considerations

This section was never included in the original text, but for completeness has been added here. Include a new section with the following text:

6. IANA Considerations

This specification does not create any RADIUS attributes nor any new number spaces for IANA administration. However, it does require assignment of new values to existing RADIUS attributes. These include:

Attribute	Values Required
NAS-Port-Type	Token-Ring (20), FDDI (21)
Tunnel-Type	VLAN (13)
Acct-Terminate-Cause	Supplicant Restart (19)
	Reauthentication Failure (20)
	Port Reinitialized (21)
	Port Administratively Disabled (22)

D.5 Table of attributes

A whole new section has been added which includes a table of all the RADIUS attributes may be seen in 802.1X exchanges. Include the following text and table as a new clause D.5

The following table provides a guide to which attributes may be sent and received as part of IEEE 802.1X authentication. L3 denotes attributes that will be understood only by Authenticators implementing Layer 3 capabilities. For each attribute, the reference provides the definitive information on usage.

802.1X	**#**	**Attribute**
X	1	User-Name [4]
	2	User-Password [4]
	3	CHAP-Password [4]
X	4	NAS-IP-Address [4]
X	5	NAS-Port [4]
X	6	Service-Type [4]
	7	Framed-Protocol [4]
	8	Framed-IP-Address [4]
	9	Framed-IP-Netmask [4]
L3	10	Framed-Routing [4]
X	11	Filter-Id [4]
X	12	Framed-MTU [4]
	13	Framed-Compression [4]
L3	14	Login-IP-Host [4]
L3	15	Login-Service [4]
L3	16	Login-TCP-Port [4]
X	18	Reply-Message [4]

	19	Callback-Number [4]
	20	Callback-Id [4]
L3	22	Framed-Route [4]
L3	23	Framed-IPX-Network [4]
X	24	State [4]
X	25	Class [4]
X	26	Vendor-Specific [4]
X	27	Session-Timeout [4]
X	28	Idle-Timeout [4]
X	29	Termination-Action [4]
X	30	Called-Station-Id [4]
X	31	Calling-Station-Id [4]
X	32	NAS-Identifier [4]
X	33	Proxy-State [4]
	34	Login-LAT-Service [4]
	35	Login-LAT-Node [4]
	36	Login-LAT-Group [4]
L3	37	Framed-AppleTalk-Link [4]
L3	38	Framed-AppleTalk-Network [4]
L3	39	Framed-AppleTalk-Zone [4]
X	40	Acct-Status-Type [5]
X	41	Acct-Delay-Time [5]
X	42	Acct-Input-Octets [5]
X	43	Acct-Output-Octets [5]
X	44	Acct-Session-Id [5]
X	45	Acct-Authentic [5]
X	46	Acct-Session-Time [5]
X	47	Acct-Input-Packets [5]
X	48	Acct-Output-Packets [5]
X	49	Acct-Terminate-Cause [5]
X	50	Acct-Multi-Session-Id [5]
	51	Acct-Link-Count [5]
X	52	Acct-Input-Gigawords [6]
X	53	Acct-Output-Gigawords [6]
X	55	Event-Timestamp [6]
	60	CHAP-Challenge [4]
X	61	NAS-Port-Type [4]
	62	Port-Limit [4]
	63	Login-LAT-Port [4]
X	64	Tunnel-Type [20]
X	65	Tunnel-Medium-Type [20]
L3	66	Tunnel-Client-Endpoint [20]
L3	67	Tunnel-Server-Endpoint [20]
L3	68	Acct-Tunnel-Connection [21]
L3	69	Tunnel-Password [20]
	70	ARAP-Password [6]
	71	ARAP-Features [6]
	72	ARAP-Zone-Access [6]
	73	ARAP-Security [6]
	74	ARAP-Security-Data [6]
	75	Password-Retry [6]
	76	Prompt [6]
X	77	Connect-Info [6]
X	78	Configuration-Token [6]

X	79	EAP-Message [6]
X	80	Message-Authenticator [6]
X	81	Tunnel-Private-Group-ID [20]
L3	82	Tunnel-Assignment-ID [20]
X	83	Tunnel-Preference [20]
	84	ARAP-Challenge-Response [6]
X	85	Acct-Interim-Interval [6]
X	86	Acct-Tunnel-Packets-Lost [21]
X	87	NAS-Port-Id [6]
	88	Framed-Pool [6]
L3	90	Tunnel-Client-Auth-ID [20]
L3	91	Tunnel-Server-Auth-ID [20]
X	95	NAS-IPv6-Address [23]
	96	Framed-Interface-Id [23]
L3	97	Framed-IPv6-Prefix [23]
L3	98	Login-IPv6-Host [23]
L3	99	Framed-IPv6-Route [23]
L3	100	Framed-IPv6-Pool [23]

```
Key
===
802.1X    = May be used with IEEE 802.1X authentication
L3        = Typically implemented only by Authenticators with
            Layer 3 capabilities
```

Other Stuff

The entire list of references from draft-20 has been included here to cross reference with what is in 802.1X. Unfortunately, the references haven’t been tracked above to indicate which ones are new. It isn’t clear which if any of these need to be incorporated into 802.1aa.

[RFC1321] Rivest, R., Dusse, S., "The MD5 Message-Digest Algorithm", RFC 1321, April 1992.

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", RFC 2119, March, 1997.

[RFC2284] Blunk, L., Vollbrecht, J., "PPP Extensible Authentication Protocol (EAP)", RFC 2284, March 1998.

[RFC2865] Rigney, C., Rubens, A., Simpson, W., Willens, S., "Remote Authentication Dial In User Service (RADIUS)", RFC 2865, June 2000.

[RFC2866] Rigney, C., "RADIUS Accounting", RFC 2866, June 2000.

[RFC2867] Zorn, G., Mitton, D., Aboba, B., "RADIUS Accounting Modifications for Tunnel Protocol Support", RFC 2867, June 2000.

[RFC2868] Zorn, G., Leifer, D., Rubens, A., Shriver, J., Holdrege, M., Goyret, I., "RADIUS Attributes for Tunnel Protocol Support", RFC 2868, June 2000.

[RFC2869] Rigney, C., Willats, W., Calhoun, P., "RADIUS Extensions", RFC 2869, June 2000.

[RFC3162] Aboba, B., Zorn, G., Mitton, D.,"RADIUS and IPv6", RFC 3162, August 2001.

[IEEE8021X] IEEE Standards for Local and Metropolitan Area Networks: Port based Network Access Control, IEEE Std 802.1X-2001, June 2001.

Informative references

[RFC2104] Krawczyk, H., Bellare, M., Canetti, R.,"HMAC: Keyed-Hashing for Message Authentication", RFC 2104, February 1997

[RFC2434] Alvestrand, H. and T. Narten, "Guidelines for Writing an IANA Considerations Section in RFCs", BCP 26, RFC 2434, October 1998.

[RFC2548] Zorn, G., "Microsoft Vendor-specific RADIUS Attributes", RFC 2548, March 1999.

[RFC2607] Aboba, B., Vollbrecht, J., "Proxy Chaining and Policy Implementation in Roaming", RFC 2607, June 1999.

[RFC2716] Aboba, B., Simon, D., "PPP EAP TLS Authentication Protocol", RFC 2716, October 1999.

[MD5Attack] Dobbertin, H., "The Status of MD5 After a Recent Attack." CryptoBytes Vol.2 No.2, Summer 1996.

[IEEE802] IEEE Standards for Local and Metropolitan Area Networks: Overview and Architecture, ANSI/IEEE Std 802, 1990.

[IEEE8021Q] IEEE Standards for Local and Metropolitan Area Networks: Draft Standard for Virtual Bridged Local Area Networks, P802.1Q, January 1998.

[IEEE8023] ISO/IEC 8802-3 Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Common specifications - Part 3: Carrier Sense Multiple Access with Collision Detection (CSMA/CD) Access Method and Physical Layer Specifications, (also ANSI/IEEE Std 802.3-1996), 1996.

[IEEE80211] Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific Requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, IEEE Std. 802.11-1999, 1999.

[Berkeley] Borisov, N., Goldberg, I., Wagner, D., "Intercepting Mobile Communications: The Insecurity of 802.11", ACM

SIGMOBILE, Seventh Annual International Conference on Mobile Computing and Networking, July 2001, Rome, Italy.

[Arbaugh] Arbaugh, W., Shankar, N., Wan, J.Y.C., "Your 802.11 Wireless Network has No Clothes", Department of Computer Science, University of Maryland, College Park, March 2001.

[Fluhrer] Fluhrer, S., Mantin, I., Shamir, A., "Weaknesses in the Key Scheduling Algorithm of RC4", Eighth Annual Workshop on Selected Areas in Cryptography, Toronto, Canada, August 2001.

[Stubbl] Stubblefield, A., Ioannidis, J., Rubin, A., "Using the Fluhrer, Mantin and Shamir Attack to Break WEP", 2002 NDSS Conference.