Dr. Roger B. Marks
NIST
325 Broadway, MC 818.00
Boulder, CO 80305 USA
Tel: +1 303 497 7837
mailto: marks at nist.gov

As requested in your liaison letter of May 5, 2005, the IETF has completed a security review of IEEE 802.16e D8. The liaison from 802.16, Jeff Mandin, was very helpful during all aspects of the process.

The reviewers included Russ Housley, IETF Security Area Director, participants from the IETF EAP Working Group, as well as a team from Stanford University lead by Prof. John C. Mitchell.

EAP WG participants included:

Jari Arkko, Co-Chair EAP WG, jari.arkko at piuha.net
Gabriel Montenegro, gmonte at microsoft.com
Yoshihiro Ohba, yohba at tari.toshiba.com

Participants from Professor Mitchell's team included:

Professor John C. Mitchell, mitchell at cs.stanford.edu
Anupam Datta, danupam at theory.stanford.edu
Changhua He, changua at stanford.edu
Arnab Roy, arnab at stanford.edu
Mukund Sundararajan, mukunds at stanford.edu

The review focused on the "EAP only" mode, since the scope of the review is 802.16e's use of EAP.

The review consisted of the following elements:

a. EAP compatibility review.
This portion of the review included a comparison of IEEE 802.16e D8 against the lower layer requirements of [RFC3748], Section 3.1.

b. AAA Key Management requirements review.
This portion of the review focussed on whether IEEE 802.16e D8 conforms to the key management requirements described in [RFC4017]. Additional information on these criteria is in [AAAKEY].

c. EAP Key Management Framework review.
This portion of the review focussed on whether IEEE 802.16e D8 conforms to the EAP Key Management framework document [KEYFRAME].

d. Security review.

This portion of the review, lead by Prof. Mitchell's team, focussed on a formal description and analysis of IEEE 802.16e D8. Their analysis of IEEE 802.16e is available here:
http://www.drizzle.com/~aboba/EAP/802.16eNotes.pdf

Overall, significant issues were found in the usage of EAP by 802.16e. Issues were found with IEEE 802.16e compatibility with RFC 3748, the EAP Key Management Framework as well as AAA Key Management Requirements. Several of the issues discovered are considered "critical" in that if they are not repaired, IEEE 802.16e will provide little in the way of guaranteed security.

Sincerely yours,

Bernard Aboba
IETF Liaison to IEEE 802

EAP Compatibility Review
---

1. Section 2.4: Does the lower layer enable peer to peer operation?
   a. Support for bi-directional session key derivation?
   b. Support for tie breaking?
   c. Support for peer and authenticator roles?

IEEE 802.16e D8 does not support peer to peer operation and therefore support for bi-directional session key derivation, tie-breaking and joint peer/authenticator roles is not required.

2. Section 3.1: Lower Layer Requirements
   a. Does the lower layer support error detection?
   b. Does the lower layer provide an EAP MTU of 1020 octets or greater?
   c. Does the lower layer support fragmentation and reassembly?
   d. Does the lower layer provide ordering guarantees?
   e. Does the lower layer provide non-duplication?

IEEE 802.16e supports error detection and provides for an EAP MTU of 1020 octets or greater. Fragmentation and reassembly is supported as are ordering guarantees and non-duplication.

It appears that protection of EAP messages is not provided in all circumstances where this is possible.

We would suggest that the use of the HMAC/CMAC TLV be required for carrying EAP re-authentication messages. Protecting EAP messages with CMAC/HMAC during re-authentication will permit a BS or MS to discard EAP messages originated by someone who doesn't have the current AK (i.e.. an attacker). Care must be taken however, to ensure that inordinate delays are not imposed in a case where a BS or MS actually loses key state for some reason.

Since there are circumstances where the BS can lose key state, EAP authentication may need to be re-initiated in situations where the BS cannot demonstrate possession of previously derived keying material. Therefore not all attacks can be prevented.

3. "Authenticated EAP" mode

[RFC3748] Section 2.1 states:

" An EAP conversation MAY utilize a sequence of methods. A common example of this is an Identity request followed by a single EAP authentication method such as an MD5-Challenge. However, the peer and authenticator MUST utilize only one authentication method (Type 4 or greater) within an EAP conversation, after which the authenticator MUST send a Success or Failure packet."

The prohibition on sequences of EAP methods was added to avoid a potential man-in-the-middle vulnerability described in [KEYFRAME] Section 6.4:

" As described in [I-D.puthenkulam-eap-binding], EAP method sequences and compound authentication mechanisms may be subject to man-in-the-middle attacks. When such attacks are successfully carried out, the attacker acts as an intermediary between a victim and a legitimate authenticator. This allows the attacker to authenticate successfully to the authenticator, as well as to obtain access to the network."

By enabling use of a sequence of EAP conversations without support for cryptographic binding, "Authenticated EAP" mode creates a vulnerability to man-in-the-middle attack.

IEEE 802.16e D8 Section 7.2.2.2.2 states:

"Note that this EAP authentication method shall not derive key material and PMK"

We assume this implies that the PMK generated by the second EAP authentication is not utilized, rather than a prohibition on EAP methods that derive keys.

However, not requiring the BS to demonstrate possession of PMKs from all EAP authentications enables the man-in-the-middle attack, described in [BINDING]. This is a critical vulnerability, and we strongly suggest that IEEE 802.16e address it prior to publication.

One potential way to achieve this is for cryptographic binding to be utilized so that the BS can demonstrate possession of all of the PMKs.

4. EAP Method Requirements

IEEE 802.16e D8, Section 7.1.3.2 states:

"The particular credentials and EAP methods that are to be used are outside the scope of this specification, but they should be selected with awareness of the security issues described in [IETF RFC 3748] Section 7."

As noted in "AAA Key Management Requirements":

" EAP selects one end-to-end authentication mechanism. The mechanisms defined in [RFC3748] only support unilateral authentication, and they do not support mutual authentication or key derivation. As a result, these mechanisms do not fulfill the security requirements for many deployment scenarios, including Wireless LAN authentication [RFC4017]. To ensure adequate security and interoperability, EAP applications need to specify mandatory-to-implement algorithms."

IEEE 802.16e D8 does not specify a mandatory-to-implement EAP method. Nor does it specify the required security properties of EAP methods to be used with it. The specification as it stands permits implementations to use EAP MD5-Challenge, which does not generate keys and is vulnerable to dictionary attacks.

This is a critical vulnerability, and we strongly recommend that at a minimum IEEE 802.16e should specify security requirements for the EAP methods to be used with it. RFC 4017 (developed as part of IEEE 802.11i) serves as an example of what this would entail.

5. Integration with the EAP State Machine

The EAP state machine is defined in [EAPSM]:
http://www.ietf.org/internet-drafts/draft-ietf-eap-statemachine-06.pdf

The EAP state machine defines interface variables that are exchanged between lower-layer and EAP-layer (peer and authenticator). The list of the interface variables is shown below. IEEE 802.16e D8 does not describe how to use the variables and how to set appropriate values to the variables within the 802.16e state machine. As we learned with IEEE 802.1X-2001, vagueness in the interaction of EAP with the lower layer statemachine can be a source of serious security vulnerabilities [MISHRA].

lower-layer to peer variables:

```
   eapReq (boolean), eapReqData (EAP packet), portEnabled (boolean),
   eapRestart (boolean), altAccept (boolean), altReject (boolean)

peer to lower-layer:

   eapResp (boolean), eapNoResp (boolean), eapSuccess (boolean),
   eapFail (boolean), eapRespData (EAP packet),
   eapKeyData (EAP key), eapKeyAvailable (boolean)

lower-layer to authenticator:

   eapResp (boolean), eapRespData (EAP packet), portEnabled (boolean),
   retransWhile (integer), eapRestart (boolean), eapSRTT (integer),
   eapRTTVAR (integer)

authenticator to lower-layer:

   eapReq (boolean), eapNoReq (boolean), eapSuccess (boolean),
   eapFail (boolean), eapTimeout (boolean), eapReqData (EAP packet),
   eapKeyData (EAP key), eapKeyAvailable (boolean)"
```

We recommend that this issue be addressed prior to publication.

EAP KEY MANAGEMENT REVIEW
---

6. Secure Ciphersuite Negotiation

[AAAKEY] states:

> " The selection of the "best" ciphersuite MUST be securely confirmed. The mechanism MUST detect attempted roll back attacks."

IEEE 802.16e securely confirms selection of the "best" ciphersuite within the 3-way handshake, but it does not securely confirm other "security-relevant" capabilities such as the MAC algorithm or replay window size.

7. Key Context

As noted in "AAA Key Management Requirements":

> " Keying material MUST be bound to the appropriate context. Any party with legitimate access to keying material can determine its context, including the scope of key usage and the key lifetime. In addition, the protocol MUST ensure that all parties legitimate access to keying material have the same context for the keying material. This requires that the parties are properly identified and authenticated, so that the key scope can be determined."

IEEE 802.16e D8 does not ensure that the PMK is bound to its context such as the key lifetime and scope. We recommend that this issue be fixed prior to publication.

The PMK Key Lifetime comes into play both before and after the 3-way handshake. Prior to the 3-way handshake the PMK Key Lifetime affects handoff performance. Since EAP authentication is a high latency operation, maintaining synchronization of the PMK Key Lifetime between the BS and MS enables the MS to efficiently use its PMK cache.

After the 3-way handshake, the PMK Lifetime needs to managed to

prevent staleness of the AKs. Since EAP methods may vary in their effective key strength (e.g. EAP-TLS may be used with <1024-bit RSA, the same SIM may be used for both GPRS authentication and 802.16e weakening the effective key strength of EAP-SIM, etc.), there may be situations where it may be desirable to set the PMK lifetime to a lower value to prevent AK compromise.

In particular:

a. IEEE 802.16e does not negotiate the PMK lifetime between the MS and BS, and as a result, these parties may be out of sync with respect to the expected lifetime. Defining a long default PMK lifetime (e.g. 8 hours) is problematic, because under load the MS may reclaim resources, and this can lead to the BS and MS getting out of sync with respect to the PMK key lifetime, reducing cache efficiency.

One way to address this issue is to define a short default PMK lifetime and to allow this lifetime to be increased by mutual agreement between the BS and MS.

b. IEEE 802.16e does not completely define the PMK scope and cache structure.

An EAP peer and authenticator may each have multiple ports, and as a result, it is possible for a single EAP conversation to result in multiple MSes and BSes sharing a PMK.

Our understanding is that the intent of IEEE 802.16e is to restrict use of a PMK to the EAP peer port over which it was derived (a single MS MAC address). This should be explicitly stated.

Since EAP authenticators may have multiple ports, the EAP peer needs to be aware of the authenticator identity; this is not defined in IEEE 802.16e D8.

The BS extends the scope of the PMK by setting the "Handover optimization flags" (section 6.3.2.3.52) to tell the MS to reuse a PMK on a target BS.

However, this does not enable the MS to know all the BSes that share a given PMK. Giving the MS complete knowledge of the authenticator key scope would enable the 3-way handshake to activate all the AKs derived from a particular PMK.

The lack of an authenticator identity also means that IEEE 802.16e provides incomplete support for Channel Bindings, described in [RFC3748] Section 7.15. Lower layer support for Channel Bindings requires that the lower layer provide the same information to the peer as the authenticator provides to the backend authentication server.

IEEE 802.16e D8 provides the peer/BS with the Called-Station-Id (BS MAC address) and Calling-Station-ID (MS MAC Address) and these same parameters can be provided to the AAA server in the Access-Request (assuming that IEEE 802.16e follows the guidelines described in [RFC3580]). The major parameter that is missing within the lower layer is the NAS-Identifier or authenticator identity.

As described in [RFC3748] Section 7.15, verifying the authenticator identity between the EAP peer, authenticator and server protects against impersonation attacks. The use of an authenticator identity also enables the MS to efficiently manage its PMK cache and to determine whether the PMK is being used outside its authorized scope.

In order to bind identities to the keying material, the lower layer authenticator and peer identities need to be explicitly stated within the 3-way handshake, and bound to PMK.

c. IEEE 802.16e D8 does not define the PMK SA in sufficient detail. In order to prevent attacks arising from PMK caching, it is necessary for the PMK SA to include all related authorizations (such as those obtained from AAA). An example of PMK SA definition is provided in IEEE 802.11i Section 8.4.1.1.1.

8. Key installation and deletion

It appears that there are circumstances where a BS could hold two PMKs for a given MS (such as during EAP re-authentication). As part of the PMK cache definition, 802.16e should explicitly describe when PMKs are installed and deleted. For example, does installation of a new PMK automatically destroy the old PMK? It appears that this is implied by IEEE 802.16e D8, but it is not explicitly stated.

Does failure of the 3-way handshake result in automatic deletion of the AK and PMK? 802.16e D8 is not explicit about this either. We would suggest that it is best not to delete the PMK in this case to prevent DoS attacks. In situations where the MS has corrupted the PMK, this should not result in a deadlock as long as the MS can choose whether to initiate EAP re-authentication after a 3-way handshake failure.

Does failure of EAP authentication result in automatic deletion of the PMK? 802.16e is not explicit about this; we would suggest that it is best not to delete the PMK in this case to prevent DoS attacks.

9. Key Selection and Naming

[AAAKEY] states:

" AAA key management proposals require a robust key naming scheme, particularly where key caching is supported. Objects that cannot be named cannot be managed. All keys MUST be uniquely named, and the key name MUST NOT be based on the keying material itself."

In Section 7.2.2.2.3 the AK is directly derived from the PMK (for pure EAP authentication). As a result, the AK and PMK lifetimes are the same. However, IEEE 802.16e D8 does not insist that discard of the AK context result in discard of the PMK context.

IEEE 802.16e D7 referenced the EAP Session-ID, which presumably is carried in the RADIUS & Diameter Key-Name attribute; this reference was removed in D8.

10. AAA Integration

IEEE 802.16e has no equivalent of RFC 3580 -- a description of AAA attributes to be used with it. If left unaddressed, this is likely to result in interoperability problems with backend authentication servers.

We recommend that the needed attributes be defined in a RADEXT WG document.

AAA-Key Management Criteria Review

Algorithm independent protocol

" The AAA key management protocol MUST be algorithm independent. The ability to negotiate the use of a particular algorithm provides resilience against compromise of a particular cryptographic algorithm. The AAA protocol MUST be algorithm independent, both in terms of its own security mechanisms as well as mechanisms supported for user authentication. Algorithm independence is also REQUIRED with a Secure Association Protocol if one is defined. This is usually accomplished by including an algorithm identifier in the protocol, and by specifying the algorithm requirements in the protocol specification. For interoperability, at least one suite of mandatory-to-implement algorithms MUST be selected. Note that without protection by IPsec as described in [RFC3579] Section 4.2, RADIUS [RFC2865] does not meet this requirement, since the integrity protection algorithm can not be negotiated."

IEEE 802.16e does securely negotiate the ciphersuite used to protect data. It also supports selection of MACs and KDFs (section 11.8.4.3 and 7.5.7.1).

Strong, fresh session keys

" While preserving algorithm independence, session keys MUST be strong and fresh. Each session deserves an independent session key.

Some EAP methods are capable of deriving keys of varying strength, and these EAP methods MUST permit the generation of keys meeting a minimum equivalent key strength as defined in [RFC3766].

A fresh cryptographic key is one that is generated specifically for the intended use. In this context, that means that the AAA key management scheme MUST generate a separate session key for each session. Further, the keys MUST NOT be dependent on one another. That is, disclosure of one session key does not aid the attacker in discovering any other session keys."

Within IEEE 802.16e D8, the TEK is transported rather than derived. Since the TEK is based on input from only one party (the BS), the strength of the TEK depends entirely on the quality of the BS random number generator. The AKs cannot be refreshed without an EAP re-authentication, so that their freshness and session independence depends on the selected EAP method. This underlines the importance of EAP method requirements such as those described in [RFC4017].

Limit key scope

" Follow the principle of least privilege. Parties MUST NOT have access to keying material that is not needed to perform their own role. A party has access to a particular key if it has access to all of the secret information needed to derive it."

In our reading it appeared that IEEE 802.16e does not permit access to keying material to parties other than the EAP peer and authenticator. However, during subsequent discussion it appears that some 802.16e participants have a different interpretation. We recommend that this issue be clarified.

Replay detection mechanism

" The AAA key management protocol exchanges MUST MUST be replay protected, including AAA, EAP and Secure Association Protocol exchanges. Replay protection allows a protocol message recipient to discard any message that was recorded during a

> previous legitimate dialogue and presented as though it belonged to the current dialogue."

The IEEE 802.16e 3-way handshake is not replay protected in one of the HMAC variants.

Authenticate all parties

> " Each party in the AAA key management protocol MUST be authenticated to the other parties with whom it communicates. Authentication mechanisms MUST maintain the confidentiality of the authenticator.
>
> Authentication mechanisms MUST NOT employ plaintext passwords."

IEEE 802.16e does not utilize plaintext passwords, and provides for mutual authentication of the BS and MS within the 3-way handshake.

Peer and authenticator authorization

> " Peer and authenticator authorization MUST be performed. Authorization is REQUIRED whenever a peer associates with a new authenticator. The authorization checking prevents an elevation of privilege attack, and it ensures that an unauthorized authenticator is detected."

Via the 802.16e 3-way handshake the BS and MS both demonstrate possession of the PMK (via the AK). However, 802.16e does not ensure synchronization of key context (see above) or authorizations (such as key usage restrictions) between the BS and MS.

Session key confidentiality

> " While preserving algorithm independence, confidentiality of session keys MUST be maintained."

As long as the PMK is not compromised, confidentiality of TEKs is maintained.

Prevent the Domino effect

> " Compromise of a single authenticator MUST NOT compromise any other part of the system, especially session keys and long-term keys. There are many implications of this requirement; however, one implication deserves highlighting. An authenticator MUST NOT share keying material with another authenticator."

In our reading, it appeared that IEEE 802.16e D8 does not share keying material between authenticators, nor does it introduce additional parties into the EAP conversation beyond those defined in RFC 3748: the EAP peer, authenticator and server. However, subsequent discussion indicates that not all 802.16e participants share this view. We recommend that this issue be clarified.

Since the 802.16e TEKs are cryptographically independent of the PMK and AK, compromise of the TEK does not compromise the PMK, AK or long-term credentials. Since the independence of TEKs from each other depends on the quality of the MS random number generator, there should probably be text in 802.16e emphasizing the importance of a high quality random number generator.

NITS

Comments on Section 7.1.3.2:

o The text talks about "vendor-selected EAP methods".

This is only partially true -- presumably the vendor has to be involved on the client side but not necessarily on the access point side.

o The text also talks about "vendor-selected standardized EAP methods". But there aren't any, since EAP MD5 et al. are not applicable for wireless, and EAP TLS/SIM/AKA that are RFCs or will be aren't standardized, they are experimental or informational RFCs.

Terminology

The term "AAA-Key" should be replaced throughout the document with the term "MSK". Since the MSK is derived no matter whether AAA is in use or not, the term "AAA-Key" is confusing and will be removed from future versions of the EAP Key Management specification. The definition of the MSK is provided in RFC 3748.

References

o In the normative references section, the mobile IPv6 reference should be updated to RFC 3775.

IPv6 Address Assignment issues

IEEE 802.16e D8 Section 6.3.9.10 has made some incorrect assumptions about how IPv6 address assignment works and this section should be revised or deprecated.

IEEE 802.16e D8 Section 6.3.9.10 states:

"For an MS, if mobile IP is being used, the MS may secure it's address on the secondary management connection using Mobile IP."

Since Mobile IP does not provide for CoA assignment, we assume that this is referring to dynamic HoA assignment. Please clarify.

"For MS using IPv6 the MS shall either invoke DHCPv6 [IETF RFC 3315] or IPv6 Stateless Address Autoconfiguration [IETF RFC 2462] based on the value of a TLV tuple in REG_RSP."

In IPv6, this determination is made based on contents of the Router Advertisement, not within the lower layer. Doing the assignment in the lower layer may result in issues with DNAv6 and SEND.

REFERENCES

[802.1X] IEEE Standards for Local and Metropolitan Area Networks: Port based Network Access Control, IEEE Std 802.1X-2004, December 2004.

[802.11i] Institute of Electrical and Electronics Engineers, "Supplement to Standard for Telecommunications and Information Exchange Between Systems -- LAN/MAN Specific Requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Specification for Enhanced Security", IEEE 802.11i, July 2004.

[AN] M. Abadi and R. Needham, "Prudent Engineering Practice for Cryptographic Protocols", Proc. IEEE Computer Society Symposium on Research in Security and Privacy, May 1994.

[BINDING] Puthenkulam, J., "The Compound Authentication Binding Problem", draft-puthenkulam-eap-binding-04 (work in progress), October 2003.

[H] Housley, R., "Key Management in AAA", Presentation to the AAA WG at IETF 56, March 2003, http://www.ietf.org/proceedings/03mar/slides/aaa-5/index.html.

[RFC3748] Aboba, B., Blunk, L., Vollbrecht, J., Carlson, J. and H. Levkowetz, "Extensible Authentication Protocol (EAP)", RFC 3748, June 2004.

[RFC3766] Orman, H. and P. Hoffman, "Determining Strength for Public Keys Used For Exchanging Symmetric Keys", RFC 3766, April 2004.

[RFC4017] Stanley, D., Walker, J. and B. Aboba, "Extensible Authentication Protocol (EAP) Method Requirements for Wireless LANs", RFC 4017, March 2005.

[EAPSM] Vollbrecht, J., Eronen, P., Petroni, N. and Y. Ohba, "State Machines for Extensible Authentication Protocol (EAP) Peer and Authenticator", draft-ietf-eap-statemachine-06.txt, Internet draft (work in progress), December 2004.

[KEYFRAME] Aboba, B., Simon, D., Arkko, J., Eronen, P. and H. Levkowetz, "Extensible Authentication Protocol Key Management Framework", draft-ietf-eap-keying-06.txt, Internet draft (work in progress), April 2005.

[AAAKEY] Housley, R. and B. Aboba, "AAA Key Management", draft-housley-aaa-key-mgmt-00.txt, Internet draft (work in progress), June 2005. Available here: http://www.drizzle.com/~aboba/EAP/housley.txt

[MISHRA] Mishra, A. and W. Arbaugh, "An Initial Security Analysis of the IEEE 802.1X Standard", CS-TR-4328, University of Maryland, February 2002. Available at: http://www.cs.umd.edu/~waa/1x.pdf