INTERNATIONAL TELECOMMUNICATION UNION

ITU-T X.1034

(04/2008)

TELECOMMUNICATION
STANDARDIZATION SECTOR
OF ITU

SERIES X: DATA NETWORKS, OPEN SYSTEM COMMUNICATIONS AND SECURITY

Telecommunication security

Guideline on extensible authentication protocol based authentication and key management in a data communication network

CAUTION !

PREPUBLISHED RECOMMENDATION

This prepublication is an unedited version of a recently approved Recommendation. It will be replaced by the published version after editing. Therefore, there will be differences between this prepublication and the published version.

FOREWORD

The International Telecommunication Union (ITU) is the United Nations specialized agency in the field of telecommunications. The ITU Telecommunication Standardization Sector (ITU-T) is a permanent organ of ITU. ITU-T is responsible for studying technical, operating and tariff questions and issuing Recommendations on them with a view to standardizing telecommunications on a worldwide basis.

The World Telecommunication Standardization Assembly (WTSA), which meets every four years, establishes the topics for study by the ITU-T study groups which, in turn, produce Recommendations on these topics.

The approval of ITU-T Recommendations is covered by the procedure laid down in WTSA Resolution 1.

In some areas of information technology which fall within ITU-T's purview, the necessary standards are prepared on a collaborative basis with ISO and IEC.

NOTE

In this Recommendation, the expression "Administration" is used for conciseness to indicate both a telecommunication administration and a recognized operating agency.

Compliance with this Recommendation is voluntary. However, the Recommendation may contain certain mandatory provisions (to ensure e.g. interoperability or applicability) and compliance with the Recommendation is achieved when all of these mandatory provisions are met. The words "shall" or some other obligatory language such as "must" and the negative equivalents are used to express requirements. The use of such words does not suggest that compliance with the Recommendation is required of any party.

INTELLECTUAL PROPERTY RIGHTS

ITU draws attention to the possibility that the practice or implementation of this Recommendation may involve the use of a claimed Intellectual Property Right. ITU takes no position concerning the evidence, validity or applicability of claimed Intellectual Property Rights, whether asserted by ITU members or others outside of the Recommendation development process.

As of the date of approval of this Recommendation, ITU [had/had not] received notice of intellectual property, protected by patents, which may be required to implement this Recommendation. However, implementers are cautioned that this may not represent the latest information and are therefore strongly urged to consult the TSB patent database at http://www.itu.int/ITU-T/ipr/.

ITU-T Recommendation X.1034

Guideline on extensible authentication protocol based authentication and key management in a data communication network

Summary

The extensible authentication protocol (EAP) is an authentication framework that supports multiple authentication mechanisms between a supplicant and an authentication server in a data communication network. EAP can be used as a basic tool for enabling user authentication and distribution of session keys in a data communication network. Since there are several EAP methods, the application designer should select the optimal EAP method among them.

This Recommendation describes a framework for EAP-based authentication and key management for securing the lower layer in a communication network. It provides guidance on the selection of EAP methods and describes the mechanism for key management for the lower layer of a data communication network. The framework described in this recommendation can be applied to protect data communication networks with either wireless access network or wired access network with a shared medium.

Contents

1 Scope

The extensible authentication protocol (EAP) is an authentication framework that supports multiple authentication mechanisms between a supplicant and an authentication server. EAP can work directly over lower layers, e.g., data link layer such as point-to-point protocol (PPP), IEEE 802, CDMA2000, UMTS, or VDSL/ADSL. For example, IEEE 802.1X is a typical transport mechanism for EAP over 802 LANs. The EAP basically performs authentication for a device attached to a LAN, establishing secure point-to-point connection or preventing access by an unauthorized device. In other words, EAP can be used to authenticate the supplicant wishing to access the network. The AAA function may be used as one of the key functions for lower-layer security of a data communication network. AAA enables transporting the secret key from the authentication server to the authenticator. Thus, defining the requirements of the EAP method and key management protocol, establishing criteria for selecting an optimal EAP method among several existing EAP methods, and defining a suitable framework for EAP and an optimal key management protocol including key derivation methods for lower-layer security in end-to-end data communication are essential. This Recommendation applies mainly to EAP-based authentication and key management protocol for data communication with a wireless access network where communication through the wireless access network should be protected by the key material derived from the key management protocol.

This Recommendation describes a framework for authentication and key management to secure the lower layer in data communication. It also provides guidance on the selection of EAP methods for a data communication network and describes the mechanism for key management and possible key hierarchy for lower-layer security in a data communication network. This Recommendation is to provide complete sets for EAP-based authentication itself but also the key management from the threat analysis to requirements, allowing the network operator to choose an adequate EAP method by using some criteria described for a specific network environment.

2 References

The following ITU-T Recommendations and other references contain provisions that constitute the provisions of this Recommendation through referencing. At the time of publication, the editions indicated were valid. All Recommendations and other references are subject to revision; thus, users of this Recommendation are encouraged to explore the possibility of applying the most recent editions of the Recommendations and other references listed below. A list of the currently valid ITU-T Recommendations is published regularly.

Any Recommendation referred to within this Recommendation as a standalone document does not give it the status of a Recommendation.

[ITU-T X.805] ITU-T Recommendation X.805 (2003), *Security Architecture for Systems Providing End-to-End Communications.*

[ITU-T X.1121] ITU-T Recommendation X.1121 (2004*), Framework of Security Technologies for Mobile End-to-end Data Communications.*

[ITU-T X.1151] ITU-T Recommendation X.1151 (2007*), Guidelines for Securing Password-Based Authentication Protocol with Key Exchange.*

[IETF RFC 4017] IETF RFC 4017 (2005), *Requirements of the Extensible Authentication Protocol (EAP) Method for Wireless LANs.*

[ISO/IEC 8802-11] ISO/IEC 8802-11:2005/Amd.6:2006, *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 - AMENDMENT 6: Medium Access Control (MAC) Security Enhancements).*

3 Terms and definitions

3.1 Terms from IETF RFC 4017 and their definitions

(a) **4-Way Handshake**: A 4-way handshake is a process consisting of 4 messages exchanged by two parties, where a pair-wise master key is involved. As a Pair-wise Authentication and Key Management Protocol (AKMP) defined in [ISO/IEC 8802-11], it confirms the mutual possession of a Pair-wise Master Key by two parties and distributes a Group Key.

(b) **AAA (authentication, authorization, accounting)**: The AAA protocol can be used as transport mechanism for the EAP message; it consists of RADIUS and Diameter. In general, the terms "AAA server" and "backend authentication server" are used interchangeably.

(c) **Authenticator**: The authenticator refers to the endpoint of the link initiating EAP authentication when a supplicant wants to access the network.

(d) **Backend authentication server**: A backend authentication server, i.e., authentication server, pertains to an entity providing authentication service to an authenticator. A typical backend authentication server is the AAA server.

(e) **EAP server**: This entity executes the EAP authentication method with the supplicant. In case no backend authentication server is used, the EAP server plays the role of the authenticator. In case a backend authentication server is used, that is, if the authenticator operates in pass-through mode, i.e., the authenticator forwards the EAP message without any modification to the supplicant or vice versa, the EAP server is placed on the backend authentication server.

(f) **Master Session Key (MSK)**: This refers to the keying material derived between the EAP peer and server and exported to the authenticator using the EAP method. MSK is at least 64 octets long. In existing implementations, an AAA server acting as an EAP server transports MSK to the authenticator. It refers to the privilege given to a supplicant by an authenticator to access the lower layer of a data communication network. In this Recommendation, MSK is used interchangeably with the Pair-wise Master Key (PMK).

(g) **Successful authentication**: This is referred to as a successful exchange of EAP messages wherein the authentication server decides to allow the supplicant access and the supplicant decides to use such access.

(h) **Supplicant**: This pertains to the endpoint responding to the authenticator. In this Recommendation, the supplicant is used interchangeably with the peer. The peer pertains to the end of the link responding to the authenticator. In [ISO/IEC 8802-11], this end is also known as the supplicant.

3.2 Terms from ITU-T Rec. X.1151 and their definitions

(a) **Man-in-the-middle attack**: This refers to an attack wherein an attacker intercepts the public key being exchanged by two entities and substitutes his/her own public key to impersonate the recipient, where the attacker can own the public key or take a copy of it while being exchanged. This attack compromises the security of the cryptosystem.

(b) **Mutual authentication**: This pertains to a type of authentication wherein the supplicant authenticates the server and the server authenticates the supplicant. Mutual authentication can prevent Phishing and Pharming attacks.

(c) **Passive attack**: This refers to an attack that involves listening, i.e., eavesdropping, without modifying or supplementing information.

(d) **PFS (Perfect Forward Secrecy):** In the cryptography of a key establishment protocol, this pertains to the condition wherein a compromised session key or long-term private key after a given session does not compromise any earlier session.

(e) **Server-compromised attack**: This refers to an attack wherein an attacker obtains verifier information from the server and launches a dictionary attack on the password file.

(f) **Server compromise-based dictionary attack:** For the password-based EAP method, the attacker is unable to impersonate the supplicant by obtaining a user password even after obtaining the hidden password file. Once the attacker compromises the server, he/she can obtain the hidden password file, i.e. hashed password file, and perform the offline dictionary attack against the hidden password file to obtain the password which can be used to impersonate the supplicant. However, this kind of attack can be prevented by encrypting the hidden password file by the secret key which is stored in the external hardware token or using some sophisticated cryptographic schemes, secret sharing schemes between the server and the hardware token. As a conclusion, this capability can be obtained by using a hardware token to store the server's secret materials.

3.3 Terms from ISO/IEC 8802-11 and their definitions

(a) **Pair-wise Master Key (PMK):** This pertains to the keying material derived between the EAP peer and server and exported to the authenticator using the EAP method. In this Recommendation, the PMK is used interchangeably with the master session key (MSK).

(b) **Pair-wise Transient Key (PTK):** This refers to the keying material derived between the EAP peer and authenticator based on the Pair-wise master key. This keying material is shared by both the peer and the authenticator.

(c) **Temporal Key (TK):** This pertains to the keying materials for the encryption and integrity of messages during later data sessions. TK generally resides in the part of PTK.

3.4 Terms defined in this Recommendation

(a) **Master Key (MK):** Top-level keying material is shared between the supplicant and the authentication server to derive the Master Session Key. In general, a Master Key is different from the Master Session Key. This is because MK represents a positive access decision for a supplicant by the authentication server.

4 Abbreviations and acronyms

This Recommendation uses the following abbreviations:

AAA Authentication, Authorization, and Accounting

ADSL Asymmetric Digital Subscriber Line

CDMA Code Division Multiple Access

DoS Denial of Service

EAP Extensible Authentication Protocol

EMSK Extended Master Session Key

IEEE Institute of Electrical & Electronics Engineers

LAN Local Area Network

MIC Message Integrity Code

MK Master Key

MSK Master Session Key

MTU Maximum Transmission Unit

NAS Network Access Server

PMK Pair-wise Master Key

PPP Point-to-Point Protocol

PTK Pair-wise Transient Key

PFS Perfect Forward Secrecy

UMTS Universal Mobile Telecommunications System

VDSL Very High Speed Digital Subscriber Line

3GPP 3rd-Generation Partnership Project

3GPP-2 3rd-Generation Partnership Project 2

5 Conventions

None

6 EAP-based authentication and key management framework

6.1 Introduction

A supplicant wishing to access the network should be authenticated by the network operator to use the network services or resources of the network operator. Moreover, when the network being accessed uses wireless transmission technology or wired access network with a shared medium, the supplicant should share the common secret with the network to protect the exchanged message in later sessions against eavesdropping, modifying, or listening. The authentication and key management framework can be used to perform mutual authentication between the supplicant and the authentication server and share the common secret between the supplicant and network access server (NAS) acting as a gateway in the access network as well. This refers to the gateway node enabling the peer to gain access to the network. The function of the authenticator generally resides in the network access server.

There are three entities required for authentication and key management: a supplicant (or peer), an authenticator, and an authentication server. The supplicant functions as an end-user or a supplicant wishing to access the network in the end-user station. The authenticator acts as a policy enforcement point mediating EAP messages between the supplicant and the authentication server. The authentication server acts as a sub-function of the AAA server, authenticating the supplicant, optionally sharing a secret that can be used to derive cryptographic keys, posting the result of authentication of an end-user to the authenticator, and forwarding the shared secret to an authenticator that can be used to derive cryptographic keys between the authenticator and the supplicant to ensure confidentiality and integrity and enable message authentication. The detailed description on policy model for the key management and key derivation is given in clause 9.1.

The path between the supplicant and the authenticator may be the wireless or wired medium used by more than one peer to exchange the message; hence the need for this path to be protected with adequate protection methods. Authentication messages for mutual authentication should be exchanged between the supplicant and authentication server using the EAP transport mechanism via the authenticator. When operating in pass-through mode, the authenticator only relays EAP messages from the supplicant to the authentication server or vice versa. There are many EAP methods that are being used in a variety of applications. Therefore, the network designer should select an adequate EAP method using some criteria for evaluating the existing EAP methods. The type and syntax of an EAP message should also be defined for authentication.

The backend protocol that transfers authentication messages from the authenticator to the authentication server should use the existing AAA protocol. There are two well-known AAA protocols: RADIUS and DIAMETER. A specific AAA protocol should be selected by defining the criteria for evaluating AAA protocols for authentication.

Authentication and key management generally consists of four operational phases: security capability discovery, EAP authentication, AAA-based key distribution , and the key management(see Figure 1). In the security capability phase, a supplicant negotiates on the security capabilities and the various parameters of the protocol to be used with the authenticator. On the other hand, in an EAP phase, the authentication server authenticates a supplicant and derives a master secret shared with the supplicant as a result of the EAP protocol. In an AAA-based key distribution phase, the authentication server transports the master secret to an authenticator to allow authentication to derive various cryptographic keys for a subsequent session between a supplicant and an authenticator. To prevent the use of the same secret key over and over and a security hole as a result of such, fresh cryptographic keys should be used in every session. Finally, in the key management phase, the authenticator exchanges random numbers with the supplicant to obtain a fresh cryptographic key; thus resulting in perfect forward secrecy.

In case the authenticator keeps the authentication-related information of a user, the authentication server is not required, i.e., the authentication server can act as part of the authenticator.

The clause is to describe an overview of the framework of the authentication and key management. The detailed operation for the key management protocol is described in clause 8.2. Since the key management function can be performed based on the policy model in clause 9.1 and the key hierarchy is constructed based on the policy model, the example of key hierarchy is described in detail in clause 9.2.

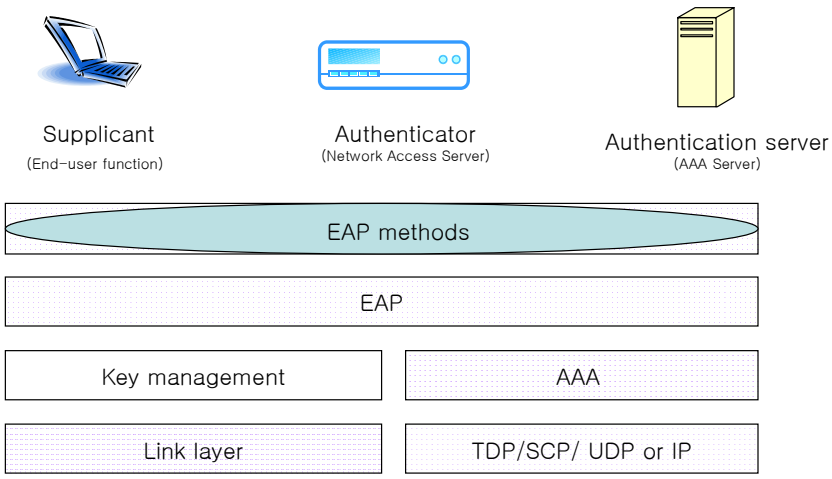

Figure 1 - EAP-based authentication and key management framework

6.2 General features of EAP

EAP should have the following features:

- **Simplicity**: Implementation should be simple, and deployment with minimal preexisting infrastructure..
- **Wide applicability**: EAP should be applicable as much as possible to any network such as wireless access network and wired access network as well as any type of access network such as that with IEEE 802 wireless LANs, 3GPP, and 3GPP2 mobile network.
- **Security**: All kinds of major attacks should be resisted, such as eavesdropping, man-in-the-middle attack, modifications, and replay attack as well as any fabrication.
- **Extensibility**: Adding to the method possible future extensions on a per-need basis should be enabled.

The following are the typical advantages of an EAP protocol:

- An EAP protocol can work with multiple authentication mechanisms. This suggests its independence from any specific authentication mechanism.
- As the authenticator, NAS (network access server) does not need to understand the details of each authentication method since it only acts as a mediator between the supplicant and the authentication server. In case a backend authentication server is used, NAS simply acts as a pass-through agent, i.e., all packets are forwarded without any modification. In some cases wherein no backend authentication server is used, a local supplicant may be authenticated by the authenticator using the supplicant's credentials as stored in the authenticator.
- The separation of the authenticator from the backend authentication server simplifies credentials management and policy decision making.

As a typical disadvantage of the EAP protocol, proving the security of the EAP protocol and key management protocol may be somewhat difficult in case the authenticator is separated from the backend authentication server.

6.3 Basic operational procedures for authentication and key management protocols

EAP authentication takes place through the following steps:

- The authenticator sends a Request packet known as Authentication Request to authenticate the supplicant.
- The supplicant sends a Response packet known as authentication response in response to a valid Request.
- The authenticator sends additional request packets, and the supplicant replies with a Response.
- The conversation continues until the authenticator can no longer authenticate the supplicant or successful authentication is deemed completed.

After a user is authenticated by the authenticator, an optional key management protocol ! mainly based on a 4-way handshake process between the supplicant and the authenticator should be executed to derive or share a common session key for subsequent communication sessions.

7 EAP protocols

7.1 Vulnerabilities in EAP

The general threat model for data communication and mobile data communication can be applied to the threat models of Recommendations X.805 and X.1121. Note, however, that there are several practical vulnerabilities associated with the EAP protocol:

- **Eavesdropping**: An attacker may try to obtain useful information by eavesdropping on authentication traffic.
- **Modification or fabrication**: This attack can be regarded as one sort of the attacks resulting from man-in-the middle attack. An attacker may try to modify or send fake EAP packets.
- **DoS**: An attacker may launch denial of service attacks by spoofing lower-layer indications or Success/Failure packets, replaying EAP packets, or generating packets with overlapping Identifiers.
- **Online dictionary attack**: In case the password-based EAP method is used, an attacker may attempt to launch an online dictionary attack by applying password of the dictionary to pass authentication verification to obtain the adequate password on the message obtained during the successful protocol being run. As a form of protection, the failed authentication trials by the server can be taken into account.
- **Offline dictionary attack**: In case the password-based EAP method is used, an attacker may attempt to recover the password by launching an offline dictionary attack on the message obtained during the previous successful protocol run.
- **Man-in-the-middle-attack**: An attacker may reside on the path between a supplicant and a server and attempt to convince the peer to be a legal peer by mounting a man-in-the-middle attack.
- **Use of weak authentication**: An attacker may attempt to disrupt EAP negotiation to cause a weak authentication method to be selected. This attack can be regarded as one sort of attacks resulting from downgrading attack and usually takes place as a result of the downgrading attack as below.

- **Weak key derivation**: An attacker may attempt to recover keys by taking advantage of weak key derivation techniques used within the EAP methods.

- **Weak cipher suites**: An attacker may attempt to take advantage of weak ciphersuites subsequently used after the EAP conversation is completed. If the conversation is completed, the attacker can exploit the weakness of the negotiated weak cipher suites to compromise the supplicant or the authentication server.

- **Downgrading attack**: An attacker may attempt to perform downgrading attacks on lower-layer ciphersuite negotiation to ensure that a weaker ciphersuite is selected subsequently for EAP authentication. An attacker acting as an authenticator may provide incorrect information to the EAP peer and/or server using out-of-band mechanisms (e.g., through AAA or lower-layer protocol). This involves impersonating another authenticator or providing inconsistent information to the peer and EAP server.

- **Identity exposure**: The attacker learns the identity of the supplicant by eavesdropping on exchanged messages during a successful protocol run. This attack can be regarded as one sort of attacks resulting from eavesdropping and usually takes place as a result of the "eavesdropping" attack.

- **Channel hijacking**: The attacker hijacks the session established between the supplicant and the authentication server.

- **Server compromised dictionary attack:** For a password-based EAP method, the attacker is unable to impersonate the supplicant by obtaining a user password even after obtaining the password file. When the attacker compromise the server, he/she can obtain the hidden password file, i.e. hashed password file, and perform the offline dictionary attack against the hidden password file to obtain the password which can be used to impersonate the supplicant. However, this kind of attack can be prevented by encrypting the hidden password file by the secret key which is stored in the external hardware token or using some sophisticated cryptographic schemes, i.e. the secret sharing scheme between the server and the hardware token. As a conclusion, this capability may be obtained by using a hardware token to store the server's secret materials.

7.2 Set of requirements for EAP

Since EAP can be performed over wired or wireless medium depending on the specific access network, several requirements for EAP methods were derived taking into account the requirements of WLAN [b-EAPKMP] as follows:

- **Secure generation of symmetric keying material**. This refers to the ability of EAP to generate keying material to protect the subsequent EAP session or subsequent data session. In other words, the supplicant and the authentication server share a common secret: top-level key. The top-level key is referred to as Master Key (MK). All cryptographic symmetric keys of lower-layer security may be derived from the Master Key.

- **Minimum key strength**. An EAP method should be capable of generating the keying material of a master key with at least 128-bit effective key strength.

- **Mutual authentication**. This pertains to an ability of the EAP method wherein an authentication server authenticates a supplicant and a supplicant authenticates an authentication server at the same time.

- **Maintenance of synchronized state between two entities**. Once the EAP method is successfully completed on the EAP peer and the server, the shared EAP method state of both sides is synchronized.

- **Seamless compatibility**. EAP can work smoothly with the existing AAA infrastructure such as RADIUS and DIAMETER infrastructure.

- **Resistance to dictionary attacks**. This refers to the immunity to dictionary attacks. There are two kinds of dictionary attacks: online dictionary attack and offline dictionary attack. When password authentication is used, passwords are commonly selected from a small set; thus raising concerns over dictionary attacks. If a password is used as a secret, a method may provide protection against dictionary attacks if it does not allow an offline attack with a work factor based on the number of passwords in an attacker's dictionary.

- **Protection against man-in-the-middle attacks**. EAP can be protected from a man-in-the-middle attack through "Cryptographic binding," "Integrity protection," "Replay protection," and "Session independence."

- **Protection against server-compromised attack**. This pertains to the ability of the EAP method to resist a server-compromised attack. Specifically, even after obtaining the password file, the attacker is not able to impersonate the supplicant without performing an exhaustive dictionary attack on the compromised password file to obtain a user password.

- **Prevention of domino effect or Denning Sacco attack**. Compromising a single authenticator is not tantamount to compromising any other part of the system including session keys and long-term secrets.

- **Replay protection**: All messages exchanged by EAP must be replay-protected.

- **Protected ciphersuite negotiation of the EAP procedure**. This refers to the ability of an EAP method to negotiate the ciphersuite used to protect the EAP conversation as well as to protect the negotiation, not the ability to negotiate the ciphersuite used to protect data. If the EAP method negotiates on the ciphersuite used to protect the EAP conversation, the "Protected ciphersuite negotiation" security claim must be supported. The protected ciphersuite negotiation should be negotiated during each EAP trial to avoid compromising a particular cryptographic algorithm.

- **Strong, fresh session keys**. Session keys may prove to be strong and fresh in all circumstances, at the same time maintaining algorithm independence.

- **Confidentiality of Master Keys**. The confidentiality of Master Keys must be maintained by the EAP peer and the authentication server. The peer can store MK using a secure hardware token such as Smart Card.

- **Authorization**. EAP peer and authenticator authorization must be performed. The authenticator can use the authorization information to provide classified services to the peer. Authorization information should be kept securely in the database.

- **User identity privacy**. This involves protecting the privacy of user identity. This can be obtained using the confidentiality algorithm and temporary ID of a user. In general, the temporary ID is exchanged through an encrypted message. Additional ciphersuite negotiation is required in maintaining confidentiality in the EAP procedure to ensure user identity privacy. The EAP method supports identity protection.

- **Unique naming and identifying**. Session keys could be uniquely named or identified.
- **Protection against server compromised dictionary attack.** This can be obtained by using a tamper-free token such as a smart card. An attacker compromising a server compromises the password file as well. In such case, the compromised password may be used to derive a password by launching a dictionary attack. Note, however, that this type of vulnerability can be protected using an EAP method wherein the password file is encrypted and the encrypting key is stored in the tamper-free module.
- **Channel binding.** This pertains to communication within an EAP method for integrity-protected channel properties such as endpoint identifiers that can be compared to values communicated via out-of-band mechanisms (e.g., through an AAA or a lower-layer protocol). It needs secure mechanisms for exchanging lower-layer EAP parameters, which enable the authenticated exchange of data. In case confidentiality is required, additional symmetric-key ciphersuite would be negotiated.
- **Fragmentation**. This refers to whether or not an EAP method supports fragmentation and reassembly. EAP methods support fragmentation and reassembly if EAP packets exceed the arbitrary length of minimum MTU (Maximum Transmission Unit), which refers to the size (in bytes) of the largest packet that can be passed onwards by a given layer of communication protocol.

7.3 Criteria for evaluating and classifying EAP methods

Based on the requirements in Section 7.2, the requirements can be classified into three categories: basic requirement, threat-related requirement, and supplemental requirement. Some criteria for classifying EAP protocols may be established as follows:

- Basic requirements
  - Secure generation of symmetric keying material
  - Minimum key strength
  - Mutual authentication
  - Strong, fresh session keys
  - Confidentiality of the Master Key
  - Maintenance of synchronized state between two entities
  - Seamless compatibility
  - Protected cipher suite negotiation of the EAP procedure

- Threat-related requirements
  - Resistance to dictionary attacks
  - Protection against man-in-the-middle attacks
  - Protection against the server-compromised attack for the password-based EAP method
  - Prevention of the domino effect
  - Replay protection
  - Protection against the server compromised dictionary attack for the password-based EAP method

- Supplemental requirements
  - Authorization
  - User identity privacy
  - Unique naming and identifying

- Channel binding
- Fragmentation

The object of the classification of EAP method in Table 1 is designed to be applicable to EAP methods developed in the future, not to existing EAP methods. The EAP method can be classified into three categories: fundamental-level EAP class, middle-level EAP class, and high-level EAP class. The network operator should use one of the three EAP classes. The system designer may use a certain level of EAP method considering the security requirements of the application. The fundamental-level EAP method satisfies all the mandatory requirements as listed in Table 1, e.g., the secure generation of symmetric keying material and seamless compatibility requirement. The middle-level EAP satisfies all the mandatory requirements of fundamental EAP method and adds three more mandatory requirements, e.g., use identity privacy, authorization, and protection against a server compromised attack and two more recommended requirements such as unique naming and protection against server compromise-based dictionary attacks. The high-level EAP satisfies all mandatory requirements of middle-level EAP method and adds two more mandatory requirements such as user naming and protection against server compromised dictionary attacks in case of strong password-based authentication or use of hard token in case of asymmetric algorithm. The difference between the fundamental-level EAP method and middle-level EAP method lies mainly in the capability of the attacker to impersonate the user compromising the server without a dictionary attack or any further effort. On the other hand, the difference between the middle-level EAP method and high-level EAP method lies mainly in the capability of the server using a hardware token to keep the secret to protect the user's authentication information. Therefore, the EAP method can be classified into one of the three EAP methods according to the capability. In Table 1, "M" refers to a mandatory requirement, "S," a recommended requirement, and "O," an optional requirement.

Table 1 - Classification of EAP methods

Criteria	Fundamental-level EAP	Middle-level EAP	High-level EAP
Secure generation of symmetric keying material	M	M	M
Minimum key strength	M	M	M
Mutual authentication	M	M	M
Maintenance of synchronized state between two entities	M	M	M
Seamless compatibility	M	M	M
Resistance to dictionary attacks	M	M	M
Protection against man-in-the-middle attacks	M	M	M
Prevention of domino effect or Denning Sacco attack	M	M	M
Replay protection	M	M	M
Strong, fresh session keys	M	M	M
Confidentiality of the Master Key	M	M	M

Criteria	Fundamental-level EAP	Middle-level EAP	High-level EAP
Protected ciphersuite negotiation of the EAP procedure	M	M	M
Authorization	S	M	M
Protection against the server-compromised attack for the password-based EAP method	S	M	M
User identity privacy	S	M	M
Unique naming and identifying	O	S	M
Protection against the server compromised dictionary attack or use of hard token for the password-based EAP method	O	S	M
Channel binding	O	O	S
Fragmentation	O	O	O

7.4 EAP method

A suitable EAP method can be selected by applying the criteria in Table 1. For example, EAP-TLS is a de facto standard for use in EAP-based authentication following the IEEE 802.1x authentication model. If some requirements of EAP are not met, a new EAP method that meets all the requirements of the application should be developed. Ideally, the EAP method should have user identity privacy, protection against the server compromised dictionary attack, and channel binding. Therefore, a specific EAP method satisfying all the above-mentioned requirements of a high-level EAP method can be regarded as high-level EAP method. The technical details of the specific EAP method are not covered by the scope of this Recommendation, however.

7.5 Evaluation of existing EAP methods

The evaluation result for the existing EAP methods is presented in Appendix I, which can be used as a guide for the network operator in selecting the adequate EAP method among the many existing EAP methods.

8 Key management

The following are the general requirements for lower-layer security:

- Considers several access networks such as IEEE 802.11, 3GPP, 3GPP2, VDSLs, and fixed network and works smoothly with them; in other words, since the access network may use a wireless or a wired medium for the access network, key management protocol should consider all kinds of transmission methods for secure key management
- Compliant with the existing authentication methods; an access network with its own authentication method supports rather than excludes the existing authentication method (in case the access network does not have its own authentication method, this specification must be applied)

8.1 Practical threats to a specific wireless access network

The general threat model for a mobile network can be applied to the threat model of Recommendation X.1121. In addition, the following are several practical threats exclusively associated with the wireless access network:

- **DoS**: An attacker may launch denial of service attacks by interfering with the frequency spectrum through an external radio frequency source or by sending several messages to the network element in the wireless network with the intention of overloading it and denying other subscribers or devices further access.
- **Man-in-the-middle-attack**: An attacker may reside in the path between a supplicant and an authenticator to convince the attacker to be a legal authenticator or supplicant by intercepting the communication.
- **Rogue network access server**: Without the authentication of the supplicant by the authenticator or the authentication server, the rogue network access server can pretend to be a legal node; thus giving rise to major security concerns.
- **Illegal supplicant**: Without proper authentication or authorization, the illegal supplicant tries to succeed in the authentication procedure and gains network access in the process.

8.2 General operational phases for key management

Similar to the wireless LAN, authentication and key management may consist of four operational phases (Figure 2): security capability discovery, EAP authentication, AAA-based key distribution, and key management of the lower layer. [b-NANCY]

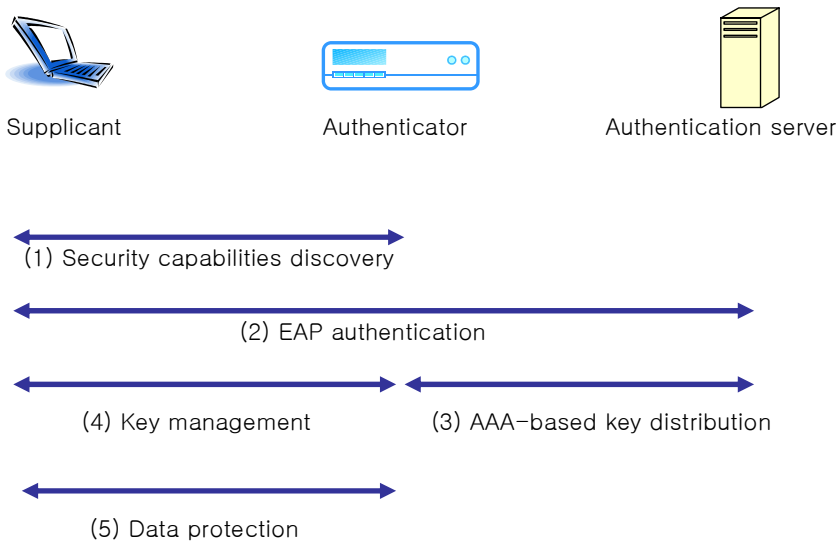

Figure 2 - Four operational phases for the authentication and key management of the lower layer

The security capability discovery phase determines the correct peer for communication, with the authenticator publishing its security capability to all supplicants periodically. At the end of the discovery phase, the supplicant is aware of the alleged network ID, alleged authentication and cipher suites the network wants to use, and correct credentials for the network, and the authenticator, aware of the types of authentication and cipher suites.

EAP authentication involves centralizing network access policy decisions at the authentication server, with the supplicant identified by the authentication server. The supplicant and the authentication server mutually authenticate each other, and an authentication server generates the master key as a side effect of authentication by using EAP method and distributes the derived master key(PMK) to the authenticator.

AAA-based key distribution involves distributing the derived master key (Pair-wise Master Key) from the authentication server to the authenticator. The detailed AAA operation is given in Appendix II.

There are two methods of sharing PMK between the supplicant and the authenticator: the pre-distribution method and the transported method. In a pre-distribution method, PMK is shared by a supplicant and an authenticator in advance. In the transported method, the Pair-wise Master Key is imported from the authentication server to the authenticator. If the pre-distribution method is used, EAP authentication and AAA-based key distribution are not required.

The key management phase involves sharing the fresh session key (Pair-wise Transient Key) from the derived master key (PMK) between the supplicant and the authenticator, proving to each other that each peer is alive and deriving all the necessary session keys (Pair-wise Transient Key) for protecting both message exchange during the key management protocol and subsequent sessions between the authenticator and the supplicant. In other words, PTK may contain cryptographic keys, e.g., keys for integrity and confidentiality, for the key management protocol.

8.3 Set of requirements for key management

The key management protocol must be executed between an authenticator and a supplicant. The EAP key management protocol in a data communication network can be said to be similar to that for WLAN in IEEE. This section describes the requirements of key management derived taking into account the requirements of WLAN [b-EAPKMP].

- **Mutual proof of possession of EAP keying material (mutual authentication)**. The supplicant and authenticator should prove possession of keying material to each other in a secure manner. For the key management protocol, the EAP peer and authenticator should prove possession of the Pair-wise Master Key transported from the backend authentication server to the authenticator to demonstrate that the peer and the authenticator have been authorized. For example, possession of keying material should be proven using the result of the hash function with the input of nonce and keying material, etc. This can protect against man-in-the-middle attacks, rogue network access server, and illegal supplicant.

- **Generation of fresh Pair-wise Transient Keys (PTKs**). The supplicant or authenticator should generate a fresh Pair-wise Transient Key from PMK for later data session in a secure manner. Ideally, PTKs should be cached in the lower layer. Deriving PTK from a portion of PMK in roaming case may result in the reuse of the shared PMK. In lower layers where the caching of EAP keying material is supported, the key management protocol should support the derivation of fresh unicast or multicast TKs even when the keying material provided by the backend authentication server is not fresh. This is typically supported via the exchange of Nonces or Counters that are then mixed with the exported keying material to generate fresh unicast session keys or even multicast session keys if possible.

- **Protection against practical threats to a specific wireless access network**. This means that there should be protection against all the threats described in section 8.1. Examples of such threats include DoS, man-in-the-middle attacks, rogue network access server, and rogue supplicants.

- **Types of PTK.** Keys in PTK can be classified into three categories: authentication key for key management protocol, encryption key for key management protocol, and encryption/authentication key (TKs) for subsequent secure traffic exchange. The authentication key can be used to ensure the integrity of messages exchanged during the implementation of the key management protocol. The encryption key for the key management protocol can be used to maintain confidentiality for specific messages, e.g., group key for subsequent data traffic.

- **Minimum key strength.** . The key management protocol may generate the keying material with 128-bit effective key strength for each key type of PTK.

- **Secure capabilities negotiation**. The supplicant and authenticator should negotiate on the capabilities in a secure manner. To protect against spoofing during the discovery phase, make sure the "best" ciphersuite is selected and protect against the forging of negotiated security parameters. The key management protocol may support secure capabilities negotiation for the key management procedure. This includes the secure negotiation of usage modes, session parameters (e.g., security association identifiers) and key lifetimes, ciphersuites, and required filters including the confirmation of security-related capabilities discovered during the key management phase.

- **Secure message protection for the key management protocol.** Messages exchanged for the key management protocol should be protected by integrity and confidentiality mechanisms. Such cryptographic services should be provided using PMK derived from MK. This can protect against man-in-the-middle attacks, rogue network access server, and illegal supplicant.

- **Key lifetime negotiation**. This features explicit key lifetime negotiation or seamless rekey. The key management protocol may handle the rekey and determination of the key lifetime. If key caching is supported, secure negotiation of key lifetimes may be required.

- **Authorization**. The authorization information of the EAP peer transport from the authentication server may be used to provide an appropriate labeled service to the peer wishing to use a specific network service. This can protect against illegal supplicant.

- **Unique entity naming**. The supplicant or authenticator should have its own Identifier. A basic feature of the key management protocol should explicitly name the parties engaged in the exchange. Without explicit identification, the parties engaged in the exchange cannot be identified.

- **Key naming and selection**. Since there are more than one key for a given key type, the key management protocol may explicitly name the keys used in the proof of possession exchange to prevent confusion when more than one set of keying material could potentially be used as basis for the exchange. To support correct processing, the key management protocol may support the naming of key management and associated transient session keys for the identification of the correct set of Pair-wise Transient Keys in processing a given packet.

- **Direct operation**. Since the key management protocol is concerned with the establishment of security associations between the EAP peer and authenticator including the derivation of PTKs, only those parties are on a "need to know" basis with PTKs. The key management protocol should operate directly between the supplicant and the authenticator; the backend authentication server should not be involved in such protocol.

- **Bidirectional operation**. While some ciphersuites only require a single set of PTKs to protect data traffic in both directions, other ciphersuites require a unique set of PTKs in each direction. The key management protocol should support the derivation of unicast temporal keys or multicast temporal keys in each direction such that two separate exchanges are not required.

- **Group key handshake protocol.** The key management protocol could be executed as an option to generate the new group key upon the completion of key management protocol. The group key generated by the authenticator can be transmitted to the supplicant from the authentication server as an option.

8.4 General flow of key management protocol

The key management protocol should be executed between the authenticator and the supplicant. By exchanging authentication information, the supplicant and the authenticator can share the extended session key derived from the pseudo-random function with the input of master session key, and random numbers generated by the authenticator and the supplicant, where a master session key is known as a PMK and extended session key a PTK. The master session key obtained after the authentication !! is transferred from the authentication server to the authenticator in a secure manner. The master session key is assumed to be known to the supplicant and the authenticator only. The 4-way handshake protocol may consist of four messages exchanged between the authenticator and the supplicant.

The authenticator begins with sending the authenticator nonce in Message 1. The supplicant selects the supplicant’s nonce and computes the extended session key, PTK, using the algorithm described in section 9.2. The PTK includes the key confirmation key, key encryption key, and pair-wise session keys. The supplicant sends the supplicant’s nonce and computes MIC (message integrity code) using the key confirmation key to enable message authentication and ensure message integrity in message 2. The authenticator can compute the PTK based on the pseudo-random function with the inputs of the authenticator’s nonce and supplicant’s nonce.

The authenticator computes MIC to enable message authentication and ensure message integrity, sending MIC and the authenticator nonce (same as authenticator nonce in Message 1) to protect against the replay attack.

The supplicant verifies MIC and computes it to ensure message integrity, sending MIC back to the authenticator in message 3. The authenticator then verifies MIC. This concludes the 4-way handshake protocol. The illustrative diagram for the four-way handshake protocol can be shown in Figure 3. In Figure 3, Info1/2/3/4 denote relevant accompanying information for each message, respectively, ANonce denotes the nonce generated by the authenticator, SNonce denotes the nonce generated by the supplicant, and MIC denotes the message integrity code for the exchanged message. After the 4-way handshake protocol, the authenticator and the supplicant may share PTK (Pair-wise Transient Key) for subsequent secure sessions between them.

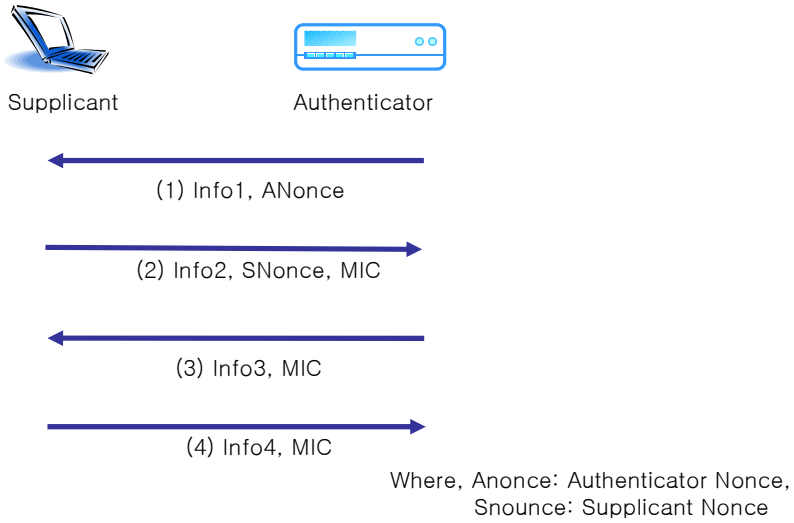

Figure 3 – Four-way handshake protocol for key management of the lower layer

8.5 Requirements classification of key management

The requirements can be classified into three categories: mandatory requirements, recommended requirements, and optional requirements. The following are the mandatory requirements of the key management protocol in a wireless access network:

- Mutual proof of possession of EAP keying material (mutual authentication)
- Generation of fresh Pair-wise Transient Keys (PTKs)
- Protection against practical threats to a specific wireless access network
- Subsequent generation of transient session key including keys for confidentiality and data integrity
- Minimum key length

- Secure capabilities negotiation
- Secure message protection for the key management protocol

The following are the recommended requirements of the key management protocol in a wireless access network:

- Unique entity naming
- Key naming and selection
- Direct operation
- Bidirectional operation
- Authorization

The following are the optional requirements of the key management protocol in a wireless access network:

- Key lifetime negotiation
- Group key handshake protocol

9 Cryptographic key for key management

9.1 General policy model

The policy decision point is defined as a logical component making policy decisions representing the access right to the access network of a data communication network with wireless network access. The policy decision can be made together with the authentication procedure by two policy decision points by exchanging EAP messages: the supplicant and the authentication server. Note that the policy decision can be represented as a policy decision token, a fresh master key which can be shared by a supplicant and the authentication server only. This token is a symmetrical key demonstrate authorization to make decision. The authentication server should distribute this token to the authenticator, where it can be used to generate the policy enforcement token representing the access right to the access network of a supplicant. Both a supplicant and the authentication server must reach the same policy decision.

The policy enforcement point is defined as a logical component enforcing policy decision by the policy decision point. The policy enforcement decision can be represented as a policy enforcement token, which is a master session key, Pair-wise Master Key. The Pair-wise Master Key can be generated by two policy enforcement points: the authentication server and the supplicant. The policy enforcement token should be shared by the authenticator and a supplicant only. In other words, the policy enforcement token is bound to this session between a supplicant and the authenticator. The policy enforcement token should be based on the policy decision token and Nonces between the authentication server and a supplicant. The possession of the policy enforcement token demonstrates authorization to access the access network of a data communication network.

Although the policy enforcement token can be derived from the policy decision token, the policy decision token should be independent from the policy enforcement token to prevent the authentication server from making access control decisions instead of the authenticator.

9.2 Possible cryptographic key hierarchy and key derivation

There are at least three levels of keys in the key hierarchy for lower-layer security in a wireless access network: Master Key (MK), Pair-wise Master Key (PMK), and Pair-wise Transient Key (PTK). The master Key (MK) is a top-level keying material shared between the supplicant and the authentication server; it can be used to derive a Pair-wise Master Key. In general, a Master Key is different from the Pair-wise Master Key. MK represents a positive access decision for a supplicant by the authentication server. The Master Key can be derived as a result of implementing the EAP protocol.

The Pair-wise Master Key (PMK) is a keying material that can be shared between the EAP peer and the server and exported to the authenticator using the EAP method. Derived from MK, PMK is at least 64 octets long. In actual implementations, an AAA server acting as an EAP server transports PMK to the authenticator. This represents the privilege given to a supplicant by an authenticator to access the lower layer of a data communication network. The Extended Pair-wise Master Key may be an additional keying material derived between the EAP supplicant and a server and can be also exported using the EAP method.

The Pair-wise Transient Key (PTK) is a keying material that can be derived from PMK along with the Nonces of the authenticator and EAP peer. PTK is used to protect both the EAP exchange and subsequent session operating in unicast mode or multicast mode. PTK contains the cryptographic key for integrity and encryption for some of the EAP messages for the key management protocol and Temporal Key (TK) for the transfer of secure messages in later sessions.

For example, the Pair-wise Master Key can be derived from the pseudo-random function with input of Master Key and several Nonces. On the other hand, the Master Key is a master secret derived from the successful completion of EAP-TLS protocol, Random 1, a random number generated at the supplicant and transferred to the authentication server, and Random 2, a random number generated at the authentication server and transferred to the supplicant. . For example, in the case of EAP-TLS, the master session key known as PMK can be derived as follows:

Pair-wise Master key (PMK) = PRF (Master Key, “Master secret” || Random 1 || Random 2)

The Pair-wise Transient Key can be derived from the pseudo-random function with inputs of PMK, supplicant nonce, authenticator nonce, authenticator’s endpoint identifier, and supplicant’s endpoint identifier. PTK is a variable length of key that can be extended to have the length required for the key between a supplicant and an authenticator.

Pair-wise Transient Key (PTK) = PRF (PMK, supplicant nonce || authenticator nonce || supplicant endpoint identifier || authenticator endpoint identifier)

The specific pseudo-random function could be a TLS-PRF defined in b-IETF RFC 2716 or other secure pseudo-random function. The Pair-wise Transient Key consists of key confirmation key, key encryption key, and temporal key. The key confirmation key and key encryption key can be used during the 4-way handshake protocol to authenticate and encrypt, respectively, the exchanged messages. The temporal key can be used to protect the message during a later data session after the 4-way handshake protocol.

Appendix I
Evaluation of existing EAP methods

(This appendix does not form an integral part of this Recommendation)

Table I.1 presents the evaluation of most well-known EAP methods based on the selection criteria in Table 1. Most well-known EAP methods are found to be noncompliant with the criteria in Table I.1. If some applications require the high-level EAP method, then new EAP methods should be developed in the future. The specific EAP method is not covered by the scope of this Recommendation, however. In Table I.1, "Y" means that the requirement is satisfied by the specific EAP method, "N," the requirement is not satisfied by the specific EAP method, and "-," the requirement is not applicable to a certain EAP.

Table I.1 - Evaluation of some of the existing EAP methods

Criteria	EAP-MD5([b-IETF RFC 3748])	EAP-TLS([b-IETF RFC 2716])	EAP-SRP([b-IETF RFC 2945])	EAP-AKA([b-IETF RFC 4187])
Secure generation of symmetric keying material	N	Y	Y	Y
Minimum key strength	N	Y	Y	Y
Mutual authentication	Y	Y	Y	Y
Maintenance of synchronized state between two entities	Y	Y	Y	Y
Resistance to dictionary attacks	Y	Y	Y	Y
Protection against man-in-the-middle attacks	Y	Y	Y	Y
Seamless compatibility	Y	Y	Y	Y
Strong, fresh session keys	N	Y	Y	Y
Prevention of domino effect or Denning Sacco attack	-	Y	Y	Y
Replay protection	Y	Y	Y	Y
Confidentiality of Master Key	-	-	-	-
Protection against server-compromised attack	-	-	Y	Y
Protected ciphersuite negotiation of the EAP procedure	-	Y	-	-
User identity privacy	N	N	N	N
Unique naming	-	-	-	-
Protection against the server compromise-based dictionary attack	-	-	N	-
Channel binding	-	-	-	-
Fragmentation	-	-	-	-

Appendix II
AAA protocol

(This appendix does not form an integral part of this Recommendation)

AAA protocol is responsible for transporting authentication messages between an authenticator and an authentication server in [b-IETF RFC 2904]. There are several proposals for transporting an authentication message: RADIUS and Diameter in [b-IETF RFC 2058] and [b-IETF RFC 3588], respectively. A possible AAA protocol must ensure the secure distribution of key material (Master key). In other words, the secure distribution of key material including a secret to derive a session key for subsequent sessions must be performed between an authenticator and an authentication server. The selection of a specific AAA protocol is not covered by the scope of this Recommendation, however. Nonetheless, the AAA protocol should be selected based on the following specific criteria.

- Protocol model
- Length of attribute field
- Type of transport layer protocol
- Session key distribution
- Error processing
- Distributed environment

AAA protocols basically provide the mechanisms for exchanging EAP packets between the authenticator and the authentication server. RADIUS is known as the most widely deployed protocol, although DIAMETER enables a high degree of flexibility that can be used to address various requirements such as transport of AAA messages, support for mobility and roaming, and enhanced security features.

RADIUS has been known to have many problems and lack features for supporting mobility and roaming requirements, i.e., scalability problems and security problems in untrusted proxy environments. This is because this protocol only supports weak hop-by-hop security; it does not define data-object security mechanisms. Moreover, RADIUS was originally designed to support a small network with a few end-users and a specific set of access control mechanisms.
On the other hand, DIAMETER was designed to support roaming and mobility; it was based on the scalability and security principle, i.e., explicit support for agents by ensuring scalability and strong hop-by-hop security based on IPSec and reliable transport based on TCP.
Even though the selection of a specific AAA protocol is not covered by the scope of this Recommendation, the use of DIAMETER as AAA protocol for a data communication network is recommended.

Bibliography

[b-NANCY] Nancy Cam-Winget, Tim Moore, Dorothy Stanley, and Jesse Walker. "IEEE 802.11i Overview," WLAN workshop, 2004.

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

[b-IETF RFC 2716] IETF RFC 2716 (1999), *PPP EAP TLS Authentication Protocol.*

[b-IETF RFC 2904] IETF RFC 2904 (2000), *AAA Authorization Framework.*

[b-IETF RFC 2945] IETF RFC 2945 (2000) *SRP Authentication and Key Exchange System.*

[b-IETF RFC 3748] IETF RFC 3748 (2004), *Extensible Authentication Protocol (EAP*).

[b-IETF RFC 4187] IETF RFC 4187 (2006) *Extensible Authentication Protocol Method for 3rd-Generation Authentication and Key Agreement (EAP-AKA).*

[b-IETF RFC 2058] IETF RFC 2058 (1997) *Remote Authentication Dial-In User Service (RADIUS).*

[b-IETF RFC 3588] IETF RFC 3588 (2003), *Diameter Base Protocol.*

[b-EAPKMP] B. Aboba, Dan Simon, J. Akko, P. Eronen, H. Levkowetz. *Extensible Authentication Protocol (EAP) Key Management Framework*, draft-ietf-eap-keying-21.txt, October 2007.