IEICE TRANS. INF. & SYST., VOL.E92–D, NO.5 MAY 2009

INVITED PAPER *Special Section on Information and Communication System Security*

Extensible Authentication Protocol Overview and Its Applications

Heung Youl YOUM[†a)], ***Nonmember***

SUMMARY The Extensible Authentication Protocol (EAP) is an authentication framework that supports multiple authentication mechanisms [38] between a peer and an authentication server in a data communication network. EAP is used as a useful tool for enabling user authentication and distribution of session keys. There are numerous EAP methods that have been developed by global SDOs such as IETF, IEEE, ITU-T, and 3GPP. In this paper, we analyze the most widely deployed EAP methods ranging from the EAP-TLS [27] to the EAP-PSK [25]. In addition, we derive the security requirements of EAP methods meet, evaluate the typical EAP methods in terms of the security requirements, and discuss the features of the existing widely-deployed EAP methods. In addition, we identify two typical use cases for the EAP methods. Finally, recent global standardization activities in this area are reviewed.

key words: *EAP, IKEv2, EAP-MD5, EAP-TLS, PEAP, TLS*

1. Introduction

EAP was introduced as an extension to PPP to allow for the flexible development of arbitrary network access authentication methods [1]–[3], [13]. EAP is considered as a framework for transporting authentication protocols, rather than as an authentication protocol itself. EAP is used as a basic tool for enabling user authentication and distribution of session keys. EAP was designed for use in network access authentication, where IP layer connectivity may not be available.

EAP method is basically carried out between the peer and the authentication server. However, several networks use three entities-based EAP model [38], for example, a wireless LAN, an IEEE 802.16e MAN, or a 3G cellular network [4], [5]. The abstract model for three-entities-based model consists of three entities: a peer, an authenticator, and an authentication server [1]. The peer refers to the entity that is wishing to access to the network or the end of the link that responds to the authenticator. The authenticator refers to the end of the link that initiates the EAP authentication. The authentication server refers to the entity that terminates the EAP method with the peer. The EAP messages exchanged between the authenticator and authentication server are encapsulated by the AAA protocol such as RADIUS [6] and Diameter protocol [50]. In case that the authenticator acts in pass-through mode, that is, the authenticator only relays the EAP packets from the peer or the authentication server, the authentication server can be part of the authenticator. Therefore, after the parties complete negotiating and choosing a specific EAP method among many candidates of the EAP methods, EAP allows for an exchange of messages between the peer and the authentication server. The conversations consist of requests and responses for exchanging authentication information.

The EAP methods can be classified into a pre-shared secret based EAP method, a public key based EAP method, and an EAP method based on both credentials according to the type of a credential. In a pre-shared secret based EAP method, the peer and the authentication server share the secret key in advance. However, in a public key based EAP method, they don't share any secret in advance, but use the public key to authenticate each other. In an EAP method using above two credentials, they authenticate themselves using either a pre-shared secret or a public key. Moreover, EAP methods are classified into a tunnel-based EAP method and a non-tunnel based EAP method. In a tunnel-based EAP method, once a secure tunnel is established, for instance, using TLS [10] or IKE [11], any authentication protocols can be used to authenticate a peer within a secure the tunnel.

The rest of the paper is organized as follows. In Sect. 2, we address a background of an EAP method including a threats model and desired security requirements. In Sect. 3, we describe numerous widely deployed EAP methods together with presenting the security features and evaluate the existing well-known EAP methods. In Sect. 4, we explore typical use cases of EAP methods, for example, a use case in Wireless LAN or a use case in the 3G cellular. In Sect. 5, several issues for future standardization of EAP methods are described. Finally, we conclude this paper.

2. Threats and Security Requirements

2.1 Security Attacks

There are a lot of threats or attacks that are related to security of the EAP methods. These attacks can be grouped into logical attacks and physical attacks as follows. The following sub-sections describe details of every possible attack.

2.1.1 Logical Attacks

The attacker with the capability to access the lower link layer may perform many types of logical attacks which are identified in [7], [38] as shown in Table 1.

Manuscript received December 1, 2008.
Manuscript revised February 24, 2009.

[†]The author is with Dept. of Information Security Engineering, Soonchunhyang University, Korea.
a) E-mail: hyyoum@sch.ac.kr

DOI: 10.1587/transinf.E92.D.766

Table 1 Various logical attacks.

Type of logical attacks	description
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.
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 peer and a server and attempt to convince the peer to be a legal peer by mounting a man-in-the-middle(MITM) 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
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 peer 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 peer and the authentication server.
Server compromised dictionary attack	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 that can be used to impersonate the peer. However, this kind of attack can be prevented by encrypting the hidden password file by the secret key that 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. In summary, this capability may be obtained by using a hardware token to store the server's secret materials.

2.1.2 Physical Attacks

There are various physical attacks that are related to security of EAP methods in the wireless LAN, which are identified in [4], [5]. They are mainly caused by a rogue access point (AP), improperly located APs, and AP with a broader coverage due to strong transmission power.

2.2 Desired Security Requirements

Considering that an EAP is performed over wired or wireless medium depending on the specific access network, there are several security requirements of EAP methods which are derived from [7], [38], [51]:

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 peer 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.

Strong, fresh session keys: Session keys may prove to be strong and fresh in all circumstances.

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

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

Integrity protection: The capability provides data origin authentication and protection of unauthorized modification for EAP packets exchanged during the EAP procedure.

Confidentiality of EAP procedure: The capability provides encryption of EAP packets during the EAP procedure. It can be used usefully in case where identity protection is needed.

Key derivation: This refers to the ability of the EAP method to derive exportable keying materials, such as the MSK (Master Session Key) and the EMSK (Extended

Master Session Key).

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

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 credential, 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 MITM attacks: EAP can be protected from the MITM 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 peer without performing an exhaustive dictionary attack on the compromised password file to obtain a user password.

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 conversations, 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" requirement must be supported. The protected ciphersuite negotiation should be negotiated during every EAP to avoid compromising a particular cryptographic algorithm.

Session independence: This refers to the demonstration that the passive attacks (such as eavesdropping on the EAP conversation) or the active attacks (including compromise of the MSK and the EMSK) does not enable compromise of subsequent or prior MSKs (Master session key) or EMSKs (extended master session key) described in Sect. 2.4.

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.

Cryptographic binding: This capability provides the validity of the EAP peer to the EAP server that a single entity has acted as the EAP peer for all methods executed within a tunnel-based EAP method. This requirement can serve to mitigate MITM attacks when the tunnel-based EAP methods are supported.

Fast reconnect: This capability is to create a new security association more efficiently by using a previously established security association. It allows the mobile terminal to facilitate a fast roaming capability in case where the roaming is supported.

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.

User identity privacy: This involves protecting the privacy of user identity. This can be obtained using the confidentiality algorithm and temporary Identifier of a user. In general, the temporary Identifier is exchanged through encrypted messages. An additional ciphersuite negotiation is required in maintaining confidentiality in the EAP procedure to ensure user identity privacy. The EAP method supports identity protection.

2.3 Primitive Cryptographic Protocols

This section describes two primitive cryptographic protocols, i.e. TLS and IKEv2, which are used to build the tunnel-based EAP methods such as EAP-TTLS [33], EAP-IKEv2 [32], and PEAP [36].

TLS protocol The primary goal of the TLS protocol is to provide privacy, message authentication, and data integrity between two parties [8]–[10]. TLS consists of four kinds of protocols: a Record protocol, a Handshake protocol, a Change-CipherSpec protocol and an Alert protocol. The TLS Handshake Protocol allows the server and the client to authenticate each other, share a master key between them, and negotiate a cryptographic algorithm and cryptographic keys before the application protocol exchanges its first byte of data. The TLS Record protocol encapsulates various higher-level protocols including Handshake protocol itself. In general, the tunnel-based EAPs use only the TLS Handshake protocol.

IKEv2 Protocol IKEv2 protocol is used by IPSec for user authentication and key exchange [49]. IKEv2 messages are exchanged between two parties, that is, an Initiator and a responder [11]. IKEv2 consists of two phases: a phase for establishing an IKE-SA and a phase for establishing further CHILD-SAs. As EAP-IKEv2 only uses the first phase of IKEv2, only the first phase of IKEv2 is related to the EAP methods.

2.4 Key Derivation for EAP

The fundamental goal of an EAP method is to authenticate a peer or an authentication server. However, as a side effect, the most recently-proposed EAP methods are able to provide a top-level keying material (known as pre-master secret) shared between a peer and an authentication server from the successful completion of an EAP method run, which are used to produce a set of necessary cryptographic keys that are used to protect a subsequent data sessions [28]. The keying materials can be derived from the long term-credential called a long-term secret. In case of the EAP method based a pre-shared secret, the long-term credential is the pre-shared secret shared between the peer and the

authentication server.

During the EAP conversations, two kinds of keys are derived as follows;

- Keys that are used only locally by EAP method itself but that are not exported. They are used to ensure confidentiality or integrity of the exchanged EAP messages. They are called TEKs (Transient EAP Keys) that are used as session keys that are used to establish a protected channel between the peer and authentication server during the EAP authentication exchange.
- Keys that are exported by the EAP method, namely a Master session key (MSK), an Extended master session key (EMSK) and an Initialization Vector (IV). All three keys must have at least 64 byte length. Derivation of MSK and EMSK is mandatory, while derivation of IV is optional.

There are two kinds of keying materials that are exported by the EAP method: master session key (MSK) and extended master session key (EMSK). In a practical EAP method, there are at least three levels of keying materials: a pre-master secret, a master secret (MS), and a master session key (MSK)/extended master session key (EMSK). The pre-master secret is generated as a result of the EAP method run. The Master secret is derived from the pre-master secret, in turn, the MSK and the EMSK are derived form the MS.

For example, in case of EAP-TLS [30], various keying materials are derived from the pre-master secret that is shared between the peer and the authentication server after a successful EAP-TLS completes. The pre-master secret is used to derive the master secret (MS), i.e. second level of keying materials, as follows;

- Master secret (MS) = TLS-PRF-48 (pre-master secret, "master secret", client random || server random)[1...47]

A client random and a server random are Nonces generated by the peer and the server and exchanged during the TLS handshake protocol, respectively. In addition, TLS-PRF-48 is a pseudo-random function specified in [8] with the length of 48 bytes. The master session key (MSK) and extended master session key (EMSK) are derived from the master secret (MS) as follows;

- MSK = TLS-PRF-64 (master secret, "client EAP encryption", client random || server random)[0...63]
- EMSK = TLS-PRF-64 (master secret, "client EAP encryption", client random || server random)[64...127]

The MSK and EMSK are exported to entities outside EAP method. The pseudo-random function (PRF) could be a TLS-PRF defined in [27] or any other pseudo-random function [12].

3. Analysis of EAP Methods

The EAP methods are classified into the EAP method based on shared secret, the EAP method based on public key, the EAP methods based either secret key or public key according to type of credentials that are used. In this section, several features of the well-known EAP methods are provided. Further, evaluations are provided in terms of the desired security requirements for the EAP methods described in Sect. 2.2.

3.1 Pre-Shared Secret Based EAP Methods

EAP-MD5 EAP-MD5 is a mandatory-to-implement EAP method of RFC 2284 [13] and a typical example of the EAP methods based on the shared secret. It is considered as one of the simplest EAP methods. The peer and the EAP authentication server share the password in advance. The one-way hash algorithm, MD5 [14] is used together with a pre-shared secret and a challenge to compute the hashed value in order to prove that the peer knows the shared secret.

It does not provide mutual authentication, that is, the authentication server only authenticates the peer. It does not generate any keying materials as a side effect. Furthermore, it is vulnerable to dictionary attacks and the MITM attack. In summary, EAP-MD5 is inherently insecure and does not support the most of the security requirements for EAP methods described in [7].

LEAP The Lightweight Extensible Authentication Protocol (LEAP), also known as Cisco wireless EAP [15], [16], was developed by Cisco system that provides the password-based authentication protocol between the peer and the authentication server. It is considered the challenge-response protocol based on a pre-shared secret or password between the peer and the authentication server.

In contrast to EAP-MD5, it supports mutual authentication and the session key derivation. However, it does not support the identity privacy and is vulnerable to the dictionary attack.

EAP-AKA The EAP-AKA [19] is developed by Ericsson and Nokia for the 3G cellular network [17], [18]. It is an EAP method that uses the existing AKA (Authentication and Key Agreement) mechanism that was developed for authentication and key exchange in the 3G cellular network. The AKA is used for mutual authentication and the session key derivation based on the shared symmetric key, which can be used to protect the data session in the air interface in the 3G cellular networks [19], [20]. On the peer side, it runs in a Subscriber Identity Module, which is either a UMTS Subscriber Identity Module (USIM) or a (Removable) User Identity Module ((R)UIM), similar to a smart card. In the 3G context, an entity called HLR (Home Location Register) acts as the authentication server, an entity called VLR (visitor location register) acts as the authenticator, and a mobile station (MS) acts as the peer, respectively [17].

Basically, EAP-AKA incorporates the AKA into EAP method to perform the authentication and the session key derivation as well as optional identity privacy support, optional result indications, and an optional fast re-authentication procedure. In addition, it is assumed that the peer has access to the subscriber's USIM, where the shared

secret K is kept and the actual AKA protocol is carried out. The master key (MK) is computed from IK (Integrity key) and CK (Cipher key) computed during the EAP-AKA method run. The MK is used to compute the transient EAP session key (TEKs), MSK and EMSK.

EAP-SRP The EAP-SRP [23] is based on the SRP (Secure Remote Password), proposed by T. Wu [21]. This scheme is known as one of the typical examples of "Strong Password Protocol" that resists dictionary attacks. Most of the pre-shared secret based EAP methods are known to be vulnerable to dictionary attacks. However, EAP-SRP is able to resist the dictionary attacks. Basically, the SRP scheme is considered as a variant of DH key exchange scheme [22] allowing two entities to agree on the common secret key by using public key cryptography.

Basically, EAP-SRP incorporates SRP into EAP method to perform the authentication and the session key derivation. However, EAP-SRP is still in draft document of IETF [23]. In summary, EAP-SRP supports mutual authentication and resists dictionary attacks. Though the Internet draft of EAP-SRP mentions the possibility to provide the identity privacy via a hidden pseudonym, it is also described that it is unable to support the strong identity privacy. In addition, EAP-SRP can support the limited fast reconnect.

EAP-PSK The EAP-PSK is proposed by both France Telecom and Siemans AG in January 2004 [25]. The PSK stands for "pre-shared Key". It provides mutual authentication based on a 16-byte pre-shared secret between the peer and the EAP server. It is mainly designed to apply to context with the restricted computational resources, especially for the mobile terminal in wireless networks. It uses only one primitive cryptographic algorithm, namely the AES algorithm [26]. There are two types of EAP-PSK method: standard EAP-PSK and extended EAP-PSK method. The standard EAP-PSK method uses the protected channel to transmit a protected result indication, while the extended EAP-PSK uses the protected tunnel to transmit the arbitrary information in variable length. It is regarded as a typical challenge/response protocol, in that two parties exchange their Nonces, their identities, and a proof of knowledge of the secret. The authentication can be achieved by sending a MAC computed with the pre-shared key over the Nonces and identities exchanged in the previous conversation.

It is based on the AKEP2 (Authenticated key exchange protocol 2) [24]. It is assumed that two parties should have shared two keys as a prerequisite, a1 and a2, where a1 is used for authentication purposes and a2 is used for session key derivation.

It supports mutual authentication, key derivation, and dictionary attack resistance. However, it does not support identity protection, fast reconnect, and the protected ciphersuite negotiation.

3.2 EAP Methods Based on Public Key

EAP-TLS EAP-TLS was developed by Microsoft [27]. It is firstly published as RFC 2716 in October 1999, which was replaced by RFC 5216 in March 2008 [30]. It is considered as a mature, stable, and widely deployed EAP method. It relies on the Transport Layer Security [8].

EAP-TLS uses a TLS Handshake phase to authenticate the peer and the authentication server. Although TLS Handshake protocol actually sets up a secure tunnel between the peer and the authentication server, this tunnel is not used in the subsequent data session. Instead, as some keying materials are sent to the authenticator, the peer and the authenticator use them to protect the subsequent data session. In EAP-TLS, certificates are used to authenticate the EAP authentication server to the peer, and, optionally, to authenticate the peer to the authentication server. Therefore, it provides mutual authentication based on X.509 certificates, which results in protecting against the MITM attacks and use of a rogue network access server. It also generates the symmetric keying material that can be used to protect the subsequent data session. After EAP-TLS is completed, the authentication server and the peer are able to share the pre-master secret. The pre-master secret is used to generate the master secret (MS), which is in turn used to generate the MSK, EMSK using the pseudo-random function as described in Sect. 2.4 [12], [28], [29].

EAP-TLS can be considered as a secure EAP method, so that it is now being widely deployed in many applications. It supports fast reconnect since new security association can be generated by using the existing security association efficiently and fast. In summary, it supports most requirements except the channel binding and identity protection. Since EAP-TLS uses certificates, it inherits all the certificated-related problems: a problem from unencrypted certificates, a problem of postponed verification of the certificate [27]. The first problem arises from that certificates are sent unencrypted. It results in revealing identity that is contained in certificate to the attacker who is able to eavesdrop on the conversation. The second problem arises from that the peer is unable to verify the signature or the certificate chain. Furthermore, the peer is unable to verify whether the certificate of the authentication server has been revoked in the meantime. Therefore, there is no other means to avoid the problem except postponing the verification.

3.3 EAP Methods Based on Both Credentials

This section describes the EAP methods based on either the public key or shared secret.

EAP-FAST EAP-FAST [31] was proposed by Cisco System as an alternative EAP method, LEAP, that is known to be vulnerable to dictionary attacks. It was originally proposed to reduce the workload of small wireless devices. FAST stands for "Flexible Authentication via Secure Tunneling". The primary design goals of EAP-FAST include the mutual authentication, resistance to brute-force dictionary attacks, immunity to the MITM attack, large support for existing user database containing credentials. In general, EAP-FAST uses the TLS handshake protocol to establish a mutually authenticated tunnel between the peer and the

authentication server. However, in contrast to EAP-TTLS, the secure tunnel can be established using either the public key similar to EAP-TLS or a pre-shared symmetric key known as PAC (Protected Access Credential). The PAC can be considered as a security token provided to the peer by the server to establish a secure tunnel for future optimized network authentication. EAP-FAST consists of two phases. In the first phase, the peer uses PAC to establish the secure TLS tunnel. If the peer does not have the corresponding PAC, the server requests the peer to initiate the full TLS Handshake. In subsequence to this full TLS Handshake, the peer requests the server to issue the PAC that can be used to establish the TLS tunnel later. In the second phase, EAP-TLS like authentication or legacy authentications may be used to authenticate the peer within the secure tunnel. PAC consists of three components: a shared secret, an opaque element, and optional other information. The shared secret is used to establish the secure tunnel. The opaque element is provided to the peer and presented to the server when the peer wishes to obtain access to the network resource. The opaque element may include the PAC and the peer's identity. The server uses a strong cryptographic algorithm to protect the opaque element in order to recover the necessary information for the server to identify and authenticate the peer. The other information may contain to provide the integrity of the PAC issuer.

There are three kinds of authentication methods: a certificate-based authentication that is used in EAP-TLS, a combined authentication that is used in EAP-TTLS, or a PAC (Protected Access Credential) based authentication. In a certificate-based authentication, the peer and the authentication server use the certificates to authenticate each other. In a PAC-based authentication, the peer uses the PAC to establish a TLS tunnel. Therefore, EAP-FAST is considered as an efficient EAP method that combines the features of EAP-TLS and EAP-TTLS and adopts the idea to use EAP-TLS with pre-shared key. In summary, the EAP-FAST is a very flexible EAP method that is intended for the constricted mobile device as it supports authenticate each other by using a pre-shared key. It meets most of the requirements described in [7].

EAP-IKEv2 The EAP-IKEv2 [32] was proposed by Simense AG, France Telecom R&D and Toshiba in March 2003. It was adopted as RFC 5106 in February 2008. This EAP method is based on mechanisms and payloads of IKEv2 [11]. It provides mutual authentication and session key establishment between an EAP server and an EAP peer. In order to authenticate each other, it supports various authentication techniques according to the types of credentials: asymmetric key pairs, symmetric keys, and a combination of both. It is possible to use a different authentication credential in each direction. For instance, the EAP server authenticates itself using the public key pairs, while the peer authenticates itself using symmetric key.

It supports most of the requirements described in [38] except channel binding.

3.4 Tunnel-Based EAP Methods

This section describes two types of the tunnel-based EAP methods; EAP-TTLS and PEAP.

EAP-TTLS EAP-TTLS is in RFC 5281 [33]. EAP-TTLS is an EAP (Extensible Authentication Protocol) method based on TLS (Transport Layer Security) protocol. TTLS stands for "Tunnel Transport Layer Security". EAP-TTLS is considered as an extension to EAP-TLS. The authentication in EAP-TLS is typically mutual, that is, the authentication server and the peer authenticate each other. It uses the certificate to authenticate the authentication server and simpler authentication method to authenticate the peer. It consists of two phase: the TLS Handshake phase and TLS tunnel phase. In the first phase, the authentication server is authenticated to the peer using X.509 certificate of the server. After the first phase is completed, the secure tunnel is established. In the second phase, all communications are protected by this secure channel. The client is authenticated to the authentication server by using the legacy authentication methods, such as clear-text password or challenge-response password, or a more advanced authentication mechanism, such as token-based authentication. EAP-TTLS supports the identity protection since an attacker can not see the user identity, as the identity can be sent in the second phase. However, EAP-TTLS is known to be vulnerable to the MITM attack as follows. The tunneled protocols require the session key derived from the first phase, which is used to provide a secure tunnel. In a certain environment, a peer is allowed to skip the first phase and to proceed directly to the second phase. In this case, the active MITM attack may take place if the attacker can hijack a valid authentication session [34]. However, a cryptographic binding scheme [35] was proposed to protect the MITM attack in tunnel based EAP method. Therefore, EAP-TTLS can be considered to be secure if a cryptographic binding is applied. In addition, IETF EMU (EAP methods update) working group [54] has been developing the Internet draft on "requirements of tunnel-based EAP method" [37], as of December 2008.

PEAP PEAP (Protected Extensible Authentication Protocol) is a proprietary protocol that was developed by Microsoft, Cisco and RSA Security [36], [37]. It provides an encrypted and authenticated tunnel using TLS Handshake protocol, which encapsulates further authentication mechanisms for the peer. It uses TLS to protect against rogue authenticators, protect against various attacks on the confidentiality and integrity of the inner EAP method exchange and provide EAP peer identity privacy. It also provides support for chaining multiple EAP mechanisms, cryptographic binding between authentications performed by inner EAP mechanisms and the tunnel, exchange of arbitrary parameters, and fragmentation and reassembly. PEAP uses the public key cryptography for authentication and negotiation of key that can be used to encrypt data. PEAP also uses TLS for server authentication and encryption, but avoid the need for user certificates by using a second authentication

Table 2 Evaluation of well-known EAP methods based on the secret key.

	EAP-MD5	LEAP	EAP-AKA	EAP-PSK	EAP-SRP
Authentication mechanism	Pre-shared key	Pre-shared key	Pre-shared key	Pre-shared key	Pre-shared key
Ciphersuite negotiation	No	No	No	No	No
Mutual authentication	No	Yes	Yes	Yes	Yes
Integrity protection	No	No	Yes	Yes	No
Replay protection	No	No	Yes	Yes	Yes
Confidentiality	No	No	Yes	No	No
Key derivation	No	No	Yes	Yes	Yes
Key strength	N/A	N/A	128-bits	128-bits	N/A
Dictionary attack resistance	No	No	N/A	Yes	Yes
Fast reconnect	No	No	Yes	No	No
Cryptographic binding	N/A	N/A	N/A	N/A	N/A
Session independence	No	No	Yes	Yes	No
Fragmentation	No	No	No	No	No
Channel binding	No	No	No	No	No
Identity protection	No	No	Limited (using temporal ID)	No	Limited(not strong)

Table 3 Evaluation of well-known EAP methods based on public key and other credentials.

	EAP-TLS	EAP-FAST	EAP-IKEv2	EAP-TTLS	PEAP
Authentication mechanism	Certificate	Certificate, shared secret	Certificate, shared secret	Certificate	Certificate
Ciphersuite negotiation	Yes	Yes	Yes	Yes	Yes
Mutual authentication	Yes	Yes	Yes	Yes	Yes
Integrity protection	Yes	Yes	Yes	Yes	Yes
Replay protection	Yes	Yes	Yes	Yes	Yes
Confidentiality	Yes	Yes	Yes	Yes	Yes
Key derivation	Yes	Yes	Yes	Yes	Yes
Key strength	2048-bits	128-bits, 2048bits	128-bits, 2048bits	2048bits	2048-bits
Dictionary attack resistance	N/A	Yes	Yes	N/A	N/A
Fast reconnect	Yes	Yes	Yes	Yes	Yes
Cryptographic binding	N/A	Yes	N/A	No	Yes
Session independence	Yes	Yes	Yes	Yes	Yes
Fragmentation	Yes	Yes	Yes	Yes	Yes
Channel binding	No	Yes	No	No	No
Identity protection	No	Yes	No	Yes	Yes

protocol between the peer and the server, which is protected by the TLS encryption. The basic principle of EAP-TTLS and PEAP is nearly identical. The main difference between them is in that PEAP can only use legacy authentication methods such as ID/Password-based authentication in the second phase, whereas EAP-TTLS can use either other EAP methods or any legacy authentication methods. However, it is still in the Internet draft of IETF as of December 2008.

3.5 Evaluation and Comparison of EAP Methods

RFC 3748 [38] presents the security requirements that the EAP methods meets and requests all EAP methods to declare whether it meets the security requirements such as protected ciphersuite negotiation, mutual authentication, integrity protection, replay protection, confidentiality, key

Table 4 Operational aspects of the EAP methods based on shared secret.

	EAP-MD5	LEAP	EAP-AKA	EAP-PSK	EAP-SRP
Server authentication	None	Hashed value of Password	Public key (certificate)	MAC based on Shared secret	Hashed value of shared secret
Peer authentication	Hashed value of Password	Hashed value of Password	MAC	MAC based on Shared secret	Hashed value of shared secret
Ease of deployment	Easy	Easy	Easy	Easy	Easy
Overall security strength	Poor	Poor	Good	Good	Good

Table 5 Operational aspects of the EAP methods based on public key and other credentials.

	EAP-TLS	EAP-FAST	EAP-IKEv2	EAP-TTLS	PEAP
Server authentication	Public key(Certificate)	Public key(certificate), MAC based shared secret	Public key(certificate), MAC based shared secret	Public key(Certificate)	Pre-shared key
Peer authentication	Public key(Certificate)	Any legacy authentication	Public key(certificate), MAC based on shared secret	Any legacy authentication like EAP-MD5, LEAP, ID/Password	Any EAP methods, any legacy authentication
Ease of deployment	Hard	Moderate	Moderate	Moderate	Moderate
Overall security strength	Good	Good	Good	Good	Good

derivation, key strength, dictionary attack resistance, fast reconnect, cryptographic binding, session independence, fragmentation, and channel binding or not. In this section, the most widely deployed EAP methods are compared in terms of these security requirements presented in only [38]. Table 2 represents the evaluation of typical EAP methods based on shared secret and Table 3 represents the evaluation of typical EAP methods based on public key and other credentials. In the Tables, "Yes" means that the requirement is satisfied by the specific EAP method, "No," the requirement is not satisfied by the specific EAP method, and "N/A," the requirement is not applicable to a certain EAP. Tables 4 and 5 represent the evaluation results of various EAP methods in terms of operational aspects. In above two Tables, in terms of "ease of deployment", it is evaluated as "hard" in case there is a need for additional infrastructure such as public key infrastructure, as "ease" in case there is no need, and as "moderate" in case there is alternative other than additional infrastructure, respectively, taking into account the server side as well as the client side. In terms of overall security strength, it was evaluated as "poor" in case there is obvious security vulnerability such as dictionary attack and evaluated and as "good" in case there are no known security vulnerabilities. The comparison results can be used as selection criteria of suitable EAP method for the network designer.

4. Use Cases for EAP Methods

This section presents two typical use cases of EAP methods [41], [44]–[47]: a use case for IEEE 802.11i WLAN [41], a use case of 3G cellular network [17]. The other use case that is not addressed in this paper is IEEE 802.16e MAN [44]–[46].

4.1 Use Case of EAP in IEEE 802.11 WLAN

The path between the peer and the authenticator in LAN context 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 peer and authentication server using the EAP transport mechanism via the authenticator [40]–[42]. When operating in pass-through mode, the authenticator only relays EAP messages from the peer to the authentication server or vice versa. The backend protocol that forwards the authentication messages from the authenticator to the authentication server should use the existing AAA protocol such as RADIUS [6] and Diameter [50].

There are three generations for protecting wired or wireless LAN standardized by IEEE [40]–[42]. The first generation is called the Wired Equivalent Privacy (WEP) released in 1999, the second generation is called the WPA (Wi-Fi Protected Access) released 2001, the third generation is called RSN (Robust Security Network) ratified in June 2004 [43]. WEP is used to protect the communication between the peer and the AP. However, WEP is still considered weak since there are so many known weaknesses that are found by many researchers [4], [5]. WPA provides an intermediate solution and mitigates the known weaknesses of WEP. It is based on TKIP (Temporal Key Integrity Protocol) and 802.1x port-based access control protocol, while being compatible with the legacy hardware based on RC4 algorithm [5]. RSN in IEEE 802.11i is new complete security architecture to provide complete security solutions for a wired or wireless LAN.

The authentication process for WPA and RSN adopted

the three-entity model in IEEE 802.1x. The three entities are known as a peer, an authentication server, and NAS (network access server) or AP (access point). The AP (Access Point) acts as the authenticator and the AAA server acts as the authentication server in a LAN context. The authentication and key management are based on the IEEE 802.1x and IEEE 802.11i. The IEEE 802.1x is a port-based network access control protocol to achieve mutual authentication and efficient key exchange between the peer and EAP server in wired or wireless LANs. It provides a mechanism to authenticate the peer to the EAP server and optionally authenticate AP to prevent rogue AP attack. It uses the EAP methods using messages to exchange the authentication requests and responses. The typical EAP methods being now used for wireless or wired LAN include EAP-TLS, EAP-TTLS and PEAP.

The operation of IEEE 802.1x in wireless LAN environments is as follows. It starts first ignoring all the packets to the AP except EAP traffic generated from the peer. The EAP messages are exchanged via so called "uncontrolled port", while secure communications take place via the controlled port, but the controlled port is blocked till the AP authenticate the peer using several EAP methods.

The operation of authentication and key management is performed among a USIM acting as a peer, an AP (access point) acting as an authenticator, and an AS (authentication server). Initially the peer captures the signals from AP. Then the peer associate with AP. When the association is completed, they authenticate themselves using the IEEE 802.1x authentication. Furthermore, the peer and the EAP server exchange the EAP messages to derive PMK (pre-master key). In case the peer and the EAP server share a secret key, PSK, the value of PSK is taken by PMK. Therefore, EAP methods are used to authenticate themselves and derive the PMK. Next, the AP and the peer performs the 4-way handshake protocol to derive the PTK (pairwise transient key) and GTK (group transient key). All the message exchanges take place through the IEEE 802.1x uncontrolled port, while 802.1x controlled port is blocked till the authentication process is completed. When the 4-way handshake completes, data is ready to be sent through the 802.1x controlled port.

4.2 Use Case of EAP in 3G Cellular Network

The 3G cellular network provides wide coverage, support roaming and attractive combination of bandwidth and quality-of-service, which makes the technique suitable for broadband applications. The EAP methods are used to authenticate the parties and derive the keying material. In this context, the USIM (universal subscriber identity module) acts as the peer, the VLR (visitor location register) acts as the authenticator, and the HLR (home location register) acts as authentication server [47]. Basically, the operation of EAP method is performed between the USIM in mobile station and authentication server in home network. After EAP method completes between USIM and HLR, they authenticate themselves and share the common key, known as cryptographic keys (CKs), which are transferred to Authenticator from HLR for protecting the wireless radio or wireless links. The CKs are used to provide the secure link for protecting data packets. The typical EAP types that are being used for 3G cellular network include EAP-AKA [19] and EAP-SIM [20].

Table 6 Available EAP methods for different network access.

Type of network access	Available EAP methods
Dial-up access	EAP-MD5, EAP-TLS
VPN remote access connections	EAP-MD5, EAP-TLS, PEAP-TLS
VPN site-to-site connections	EAP-MD45, EAP-TLS
IEEE 802.1x authentication to a switch	EAP-MD5, EAP-TLS, PEAP-TLS
IEEE 802.1x to a wireless AP	PEAP-MS-CHAP v2, EAP-TLS, PEAP-TLS

4.3 EAP Methods for Different Types of Network Access

The EAP methods are used for many types of network access such as IEEE 802.1x authentication to an switch (Wired), IEEE 802.1x authentication to a wireless AP, VPN (virtual private network) site-to-site connections, VPN remote access connections, and dial-up remote access. Table 6 lists the available EAP methods for the different types of access. PEAP-TLS means that inner authentication protocol within the secure tunnel is based on the TLS protocol in PEAP.

5. Issues for Future Standardization of EAP Methods

The activities for standardizing the EAP methods have been leaded by IETF [54]. IETF established an EMU working group, called "EAP methods update" at 64th IETF meeting in November 2005 [48]. The main goal of this working group was to development and standardization of EAP methods suited for current and upcoming network access technologies. Since then, there were numerous successful achievements made by this working group, including the update of EAP-TLS and development of various EAP methods including EAP-FAST [31] and EAP-IKEv2 [32]. Especially, they are now under development of EAP-GPSK (Generalized pre-shared key) [53] and the requirement of tunnel-based EAP method [37] as of December 2008.

However, there are still open issues in the standardization of EAP method identified in [2]. They include EAP method performance analysis, revision of EAP method requirements, and effective key strength. For the EAP method performance, it still lacks a performance analysis on the computational complexity and message size of transmitted data of EAP methods. In addition, for the revision of EAP method's requirement, the following three requirements should be seriously considered [2];

Negotiation of cryptographic algorithms: Considering

there have been a lot of controversies over the security of MD5 and SHA-1 since 2005, it is desirable to negotiate and select more secure crypto algorithm by a negotiation procedure in EAP methods in order to avoid a fragile algorithm being chosen. As you can see in Table 2 and Table 3, the most tunnel-based EAP methods satisfy this requirement, however, the rest of the EAP methods do not meet it.

Identity protection: There are two kinds of identity protections: active identity protection and passive identity protection. Active identity protection is stronger concept to the passive identity protection. The passive identity protection is typically achieved by sending the identity encrypted over the network, so that an eavesdropper is unable to identify it. Furthermore, active identity protection means that an attacker reveals its identity only if it has to talk to the relevant party. This requirement is very helpful for protecting the identity of the peer. This can be achieved when the peer's identity is transmitted encrypted over the link after the authentication server authenticates himself to the peer. Especially, EAP-IKEv2 supports this feature. As you can see in Table 2 and Table 3, the most tunnel-based EAP methods satisfy this requirement.

Perfect Forward Secrecy: PFS refers to the confidence that the compromise of long-term secret key does not result in compromising session keys of earlier session. The future EAP methods should meet this requirement.

For an effective key strength, all recently proposed EAP methods support effective key strength of 128 bits. The comparable asymmetric key strength is known to be 3,000-bits for RSA/DH [52]. However, public key operations in this key size are considered to be very expensive and lead to a significantly increased latency. The question raised is whether every application should have an effective key strength equivalent to 128-bits or not. The key strength of EAP methods should be flexible depending on a specific application or a context. In addition, there is another challenge, that is, IPR issue. Considering that there exists no strong password authentication without claiming IPR [2], the EAP method should be designed or standardized without imposing any critical IPR from the point of view of the service operator.

6. Conclusions

As Wireless access networks grow in application area and are used more frequently, the need for authentication and key exchange becomes inevitable and vital. Many innovative EAP methods have been developed to meet these requirements. In this paper, we present various threats and requirements for the EAP methods, analyze several widely-deployed EAP methods, evaluate them in terms of various requirements described in [38], and discuss several issues for future standardization of the EAP methods.

References

[1] Extensible Authentication Protocol Overview, Microsoft, at http://www.microsoft.com/technet/network/eap/eap.mspx

[2] T. Otto, "Extensible network access authentication," July 2006, available at http://www-public.tu-bs.de:8080/~y0013790/thesis-otto-eapmethods.pdf

[3] W. Simpson, "The point-to-point protocol (PPP), STD 51," RFC 1661, July 1994.

[4] A.M. Al Naamany, A.A. Shidhani, and H. Bourdoucen, "IEEE 802.11 wireless LAN security overview," International Journal of Computer Science and Network Security, vol.6, no.5B, pp.138–186, May 2006.

[5] K.H. Baek, S.W. Smith, and D. Kotz, "A survey of WPA and 802.11i RSN authentication protocols," Dartmouth College, Technical Report TR2004-524, Nov. 2004.

[6] B. Aboba and P.R. Calhoun, "RADIUS (remote authentication dial in user service) support for extensible authentication protocol (EAP)," RFC 3579, Category: Informational, Updates RFC 2869, Sept. 2003.

[7] ITU-T, Guideline on Extensible Authentication Protocol based Authentication and Key Management in a Data Communication Network, ITU-T X.1034, April 2008.

[8] T. Dierks and E. Rescorla, "The transport layer security (TLS) protocol version 1.2," RFC 5246, Aug. 2008.

[9] T. Dierks and C. Allen, "The TLS protocol version 1.0," RFC 2246, Jan. 1999.

[10] T. Dierks and E. Rescorla, "The transport layer security (TLS) protocol version 1.1," RFC 4346, April 2006.

[11] C. Kaufman, "Internet key exchange (IKEv2) protocol," ITEF RFC 4306, Dec. 2005.

[12] L. Chen, Recommendation for Key Derivation Using Pseudorandom Functions, Draft NIST Special Publication 800-108, April 2008.

[13] L. Blunk and J. Vollbrecht, "PPP extensible authentication protocol (EAP)," IETF RFC 2284, March 1998.

[14] R. Rivest, "The MD5 message-digest algorithm," RFC 1321, April 1992.

[15] Cisco, Dictionary attack of Cisco LEAP, Technical note, available at http://www.ciscosystems.ch/warp/public/707/cisco-sn-20030802-leap.pdf, July 2004.

[16] Cisco, Wireless LAN Security White Paper, http://www.ciscosystems.com/warp/public/cc/pd/witc/ao1200ap/prodlit/wswpf_wp.htm

[17] 3rd Generation Partnership Project, 3GPP Technical Specification 3GPP TS 33.102 V5.1.0: Technical Specification Group Services and System Aspects; 3G Security; Security Architecture (Release 5), Dec. 2002.

[18] 3rd Generation Partnership Project 2, 3GPP2 Enhanced Cryptographic Algorithms, Sept. 2003.

[19] J. Arkko and H. Haverinen, "Extensible authentication protocol method for 3rd generation authentication and key agreement (EAP-AKA)," IETF RFC 4187, Jan. 2006.

[20] H. Haverinen and J. Salowey, eds., "Extensible authentication protocol method for global system for mobile communications (GSM) subscriber identity modules (EAP-SIM)," RFC 4186, Jan. 2006.

[21] T. Wu, "The SRP authentication and key exchange system," IETF RFC 2945, Sept. 2000.

[22] W. Diffie and M.E. Hellman, "New directions in cryptography," IEEE Trans. Inf. Theory, vol.19, no.3, pp.644–654, 1976.

[23] J. Carson, B. Aboba, and H. Haverinen, "EAP SRP-SHA-1 authentication protocol," IETF Draft, draft-ietf-pppext-eap-srp-03.txt, July 2001.

[24] M. Bellare and P. Rogaway, "Entity authentication and key distribution," Crypto'93, Aug. 1993.

[25] F. Bersani and H. Tschofenig, "The EAP-PSK protocol: A pre-shared key extensible authentication protocol (EAP) method," RFC 4764, Jan. 2007.

[26] "Federal information processing standards (FIPS) publication 197, "Advanced encryption standard (AES)"," Nov. 2001, http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf

[27] B. Aboba and D. Simon, "PPP EAP TLS authentication protocol," RFC 2716, Oct. 1999.
[28] B. Aboba, D. Simon, and P. Eronen, "Extensible authentication protocol (EAP) key management framework," RFC 5247, Aug. 2008.
[29] H. Krawczyk, M. Bellare, and R. Canetti, "HMAC: Keyed-hashing for message authentication," RFC 2104, Feb. 1997.
[30] D. Simon, B. Aboba, and R. Hurst, "The EAP-TLS authentication protocol," RFC 5216, March 2008.
[31] N. Cam-Winget, D. McGrew, J. Salowey, and H. Zhou, "The flexible authentication via secure tunneling extensible authentication protocol method (EAP-FAST)," IETF RFC 4851, May 2007.
[32] H. Tschofenig, D. Kroeselberg, A. Pashalidis, Y. Ohba, and F. Bersani, "The extensible authentication protocol—Internet key exchange protocol version 2 (EAP-IKEv2) method," RFC 5106, Feb. 2008.
[33] P. Funk and S. Blake-Wilson, "Extensible authentication protocol tunneled transport layer security authenticated protocol version 0 (EAP-TTLSv0)," RFC 5281, Aug. 2008.
[34] N. Asokan, V. Niemi, and K. Nyberg, "Man-in-the-middle in tunneled authentication protocols," IACR ePrint Archive Report 2002/163, Oct. 2002, http://eprint.iacr.org/2002/163
[35] J. Puthenkulam, V. Lortz, A. Paleker, and D. Simon, "The compound authentication binding problem," Work in Progress, draft-puthenkulam-eap-binding-04.txt, Oct. 2003.
[36] A. Palekar, D. Simon, J. Salowey, H. Zhou, G. Zorn, and S. Josefsson, "Protected EAP protocol (PEAP) version 2," draft-josefsson-pppext-eap-tls-eap-10.txt, Oct. 2004.
[37] K. Hoeper, S. Hanna, and J. Salowey, ed., "Requirements for an tunnel based EAP method," draft-ietf-emu-eaptunnel-req-01.txt, Oct. 2008.
[38] B. Aboba, L. Blunk, J. Vollbrecht, and J. Carlson, "Extensible authentication protocol (EAP)," RFC 3748, June 2004.
[39] L. Han, "A threat analysis of the extensible authentication protocol," Honors Project Report, Carleton University, April 2006.
[40] IEEE Standard for Local and Metropolitan Area Networks, "Wireless LAN mediun access control (MAC) and physical layer specification," ANSI/IEEE Std 802.11, 1999 Edition (R2003), 2003.
[41] IEEE Standard for Local and Metropolitan Area Networks, "Wireless LAN medium access control (MAC) and physical layer specification, medium access control (MAC) security enhancements," ANSI/IEEE Std 802.11i, 2004 Edition, 2004.
[42] IEEE Standard for Local and Metropolitan Area Networks, "Port-based network access control," ANSI/IEEE Std 802.1x, 2001 Edition (R2004), 2004.
[43] Wi-Fi Alliance, Wi-Fi Protected Access, version 2.0, April 2003.
[44] IEEE P802.16e/D12, Oct. 2005. Draft IEEE Standard for Local and metropolitan area networks. Part 16: Air Interface for Fixed and Mobile Broadband Wireless Access Systems. Amendment for Physical and Medium Access Control Layers for Combined Fixed and Mobile Operation in Licensed Bands.
[45] IEEE Std 802.16-2001. IEEE Standard for Local and Metropolitan Area Networks. Part 16: Air Interface for Fixed Broadband Wireless Access Systems. http://standards.ieee.org/getieee802/download/802.16-2001.pdf
[46] IEEE Std 802.16-2004. Draft IEEE Standard for Local and Metropolitan Area Networks. Part 16: Air Interface for Fixed Broadband Wireless Access Systems. http://standards.ieee.org/getieee802/download/802.16-2004.pdf
[47] 3rd Generation Partnership Project, "3GPP technical specification 3GPP TS 33.105 4.1.0: Technical specification group services and system aspects; 3G security; Cryptographic algorithm requirements (Release 4)," June 2001.
[48] EAP Method Update Working Group. Official website. http://www1.ietf.org/html.charters/emu-charter.html
[49] W. Stalling, Cryptography and Network Security, Fourth ed., Prentice Hall, 2005.
[50] IETF RFC 3588 (2003), Diameter Base Protocol, 2003.
[51] IETF RFC 4017 (2005), Requirements of the Extensible Authentication Protocol (EAP) Method for Wireless LANs, 2005.
[52] R. Rivest, A. Shamir, and L. Adleman, "A method for obtaining digital signatures and public-key cryptosystems," Commun. ACM, vol.21, no.2, pp.120–126, 1978. http://theory.lcs.mit.edu/~rivest/rsapaper.pdf
[53] T. Clancy and H. Tschofenig, "EAP generalized pre-shared key (EAP-GPSK) method," IETF draft-ietf-emu-eap-gpsk-17, Nov. 2008.
[54] IETF Web site, www.ietf.org

Heung Youl Youm received his PhD's degree in Electronics Engineering from Hanyang University, Seoul, Korea in 1990. He received his Bachelor's and Master's degree in Electronics Engineering from Hanyang University, Seoul Korea, in 1981 and 1983, respectively. Currently, he is working as a Professor for the Department of Information Security Engineering of SoonChunHyang University, Korea. He has been working for the former MIC (Ministry of Information and Communication), Korea as a Project Manager for information security, since November, 2006. His current interest includes theoretical and practical study on various security technologies/protocols such as IPTV/USN/NGN security. He has now served as vice-president and an editor-in-chief for the KIISC Journal for KIISC (Korea Institute of Information Security and Cryptology) since 2007 and 2008, respctively. He had served as a Rapporteur of ITU-T SG17 Question 9 from 2005 to 2008, and recently he was elected as a vice-chairman of ITU-T SG17 at WTSA'08 held in Johannesburg, South Africa. Since 2005, he has contributed to ITU-T by serving as a editor of four approved ITU-T Recommendations such as X.1034 (Guideline on extensible authentication protocol based authentication and key management in a data communication network), X.1111 and X.1151, X.1191 and six ITU-T draft Recommendations under development including ITU-T X.iptvsc-3 and ITU-T X.usnsec-1.