An EAP Authentication Method using One Time Identity

M. Ait Hemad, M. A. El Kiram, A. Lazrek,

Cadi Ayyad University, Faculty of Science, Department of Computer Science, Marrakech, Morocco

Summary
Recently, several EAP (Extensible Authentication Protocol) authentication methods have been proposed to protect mobile communications. The objective of these methods is to ensure authentication in 802.11 wireless networks. However, they do not provide other security services, especially the identity protection. In this paper, we present a new EAP authentication method called EAP-OTI (OTI for One Time Identity) that takes into account the increased need to protect the identity of the client. In addition, it optimizes the computation time required to authenticate both clients and authentication servers.

Key words:
Security, authentication, mobile communication, wireless local area network, identity protection, 802.1X, EAP-TLS, EAP-MD5, PEAP.

1. Introduction

The wireless local area networks (WLANs) [13] are experiencing a growing success. This success is mainly due to their ease and speed of implementation. But, it is threatened by weak security.
To strengthen security, IEEE proposed 802.1X architecture [18] [14] [13] as a basis for authentication, access control and key management in 802.11 wireless networks. The 802.1X architecture involves three main actors: the system to authenticate (client or supplicant), which is the mobile station requesting access to the network, the access point (authenticator) and the authentication server. Often, the authentication server is RADIUS server (Remote Authentication Dial In User Service) [2]. 802.1X uses EAP (Extensible Authentication Protocol) [3] [4] for ensuring authentication. This protocol specifies a generic framework for multiple authentication methods. These methods define authentication schemes and key distribution.
So far, dozens of EAP methods exist, but most of them use personal information such as ID (Identity) and non-anonymous digital certificates, to authenticate the client. This allows any malicious user to eavesdrop the network and then knows who has access, when, where it accesses from and the service accessed to.
By accumulating this collected information, the client behavior can be deduced which constitutes a violation of the privacy of client.

Manuscript received July 5, 2010
Manuscript revised July 20, 2010

2. EAP authentication methods

The 802.1X architecture does not specify a particular authentication method. For this reason, several authentication methods are proposed including EAP-MD5 [3], EAP-TLS [1], EAP-TTLS [12], EAP-LEAP [7] [8] and PEAP [16].
The EAP messages exchanged between the mobile station and access point are carried in EAPOL (EAP over LAN) frames. And the ones exchanged between the access point and authentication server are carried in EAPOR (EAP over RADIUS) frames.
In the rest of this section, we study the three following authentication methods: EAP-MD5 that is fast and easy to implement, EAP-TLS that is the high standard for the moment and PEAP that protects the identity of the client.

2.1 EAP-MD5

The EAP-MD5 [3] [8] is based on CHAP (Challenge Handshake Authentication Protocol), which uses the principle of challenge-response to authenticate the client. EAP-MD5 requires a shared key between client and authentication server. This key is usually a password associated with a user name or identity (e.g. IP address or MAC).

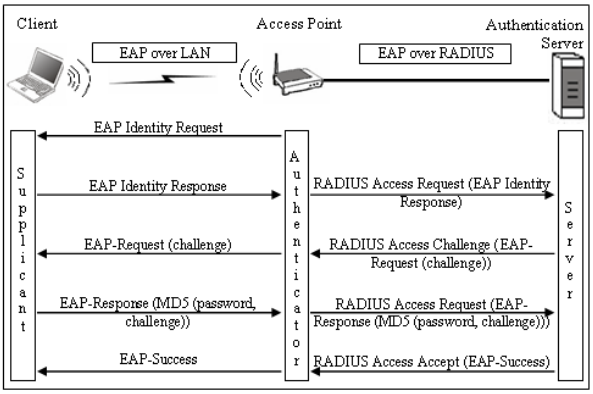

Fig. 1 EAP-MD5 exchange.

As illustrated in Figure 1, after the request of the authenticator, the client shows his identity in an EAP message to authenticator which relays it to the

authentication server. Moreover, during all exchanges EAP, the role of access point is limited to relay EAP messages between the client and authentication server. Then, the authentication server sends a random challenge value to the client who calculates a hash value by encrypting the challenge and its password, via the MD5 hashing algorithm [17]. The hash value is returned in an EAP message. The authentication server performs the same calculation of hash value as the client and compares the two hash values. Two identical hash values means the client possesses the right password, which leads to the success of the authentication and emission of an acceptance message. Otherwise, authentication fails and the server rejects the request. On this basis, the authenticator allows or denies the client accessing the network.

2.2 EAP-TLS

EAP-TLS [1] [8] is an authentication method standardized by the IETF (Internet Engineering Task Force). It comes from the TLS (Transport Layer Security) [9] which is considered an effective solution for securing exchanges.
EAP-TLS relies on the digital certificate of both the client and the server to perform mutual authentication. Figure 2 illustrates the EAP-TLS exchange.
The EAP-TLS session begins by detecting the presence of a new mobile station. The access point sends an identification request to station detected. The station responds with the client's identity (the machine name or login). This message is relayed by the access point to the authentication server. Then, the authentication server initiates the authentication process by sending EAP-TLS/start packet. The client responds with the EAP-TLS/client_hello packet that principally contains: the version of the TLS client, a random number, a session identifier and the types of encryption algorithms supported by the client. After the server responds with an EAP-Request packet containing the message server_hello, mentioning the algorithm chosen from those offered by the client, followed by the certificate server, a request from the client certificate. Next, the client checks the certificate received and responds with his own certificate. Both the client and authentication server calculate the session key. Messages change_cipher_spec trigger encryption of communications. The message finished ends the authentication phase of TLS (TLS handshake). Then, the client sends an EAP-Response packet whose data field is empty and the server responds with the message EAP-Success.

2.3 PEAP

The PEAP (Protected EAP) [16] [8] uses the same principle as that of EAP-TLS to authenticate the authentication server. Once the TLS tunnel is established, another EAP exchange (e.g. EAP-MD5 or another authentication mechanism such as PAP or CHAP) will take place within the tunnel to authenticate the client to the authentication server. Through the use of TLS tunnel, the client's identity is protected during the exchange. Figure 3 illustrates the operating principle of PEAP.

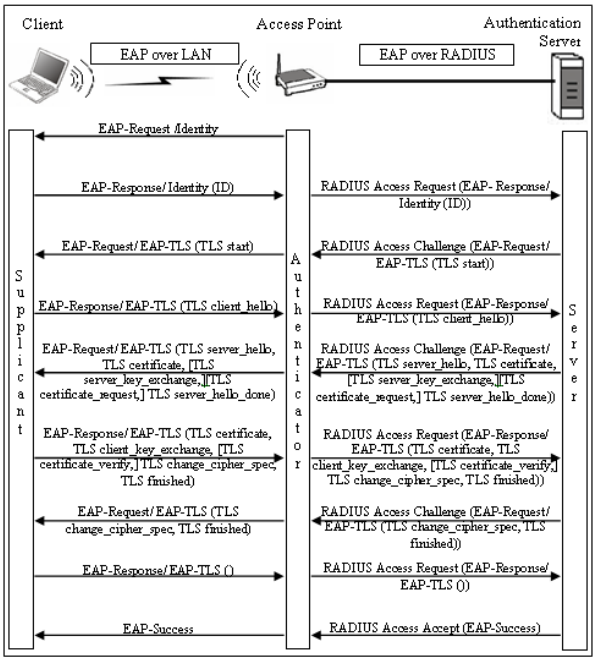

Fig. 2 EAP-TLS exchange.

.

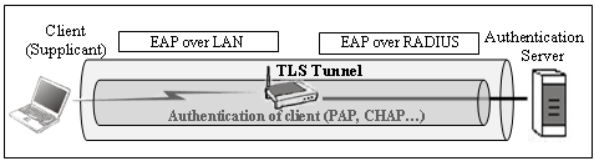

Fig. 3 Operating principle of PEAP.

2.4 Analysis of EAP authentication methods

The EAP-MD5 method offers the advantage of not requiring a lot of resources for treatment. Moreover it does not require public keys infrastructure (as required by EAP-TLS). However, it is not used today because it is recognized as vulnerable to dictionary attacks and brute force attacks. Finally, EAP-MD5 only provides one-way authentication. The server authenticates the client, but the client can not authenticate the server. Hence, it is not possible with this method to detect false servers and rogue access points (controlled by hackers).
EAP-TLS is a secure authentication method, but it is not simple to implement. Indeed, each entity must have a digital certificate, hence the necessity of establishing a Public-Key Infrastructure (PKI). In addition, in the EAP-

TLS, the client can not see the list of certificates that are no longer valid. Indeed, the checking certificates for revocation need access to the network. As the client will not be connected before the accomplishment of authentication, he takes the risk of accepting certificates without being sure of their validity. Another downside of EAP-TLS is that during authentication, the certificates are transmitted in clear text over the network. So EAP-TLS does not protect the identity.

PEAP is one of the most secure authentication methods. It is based on using only server-side certificates which greatly reduces the complexity caused by the management of PKI. In addition, the transmission of the client's identity, in a secure TLS tunnel, provides protection of this identity. Its main drawback is the high cryptographic load, while in wireless networks, mobile devices often have limited computing capacity.

3. EAP-OTI Authentication Method

The EAP-OTI (EAP-One Time Identity), we propose, ensures mutual authentication and the protection of client identity while reducing the cryptographic load required by certain methods such as EAP-TLS, EAP-TTLS and PEAP. EAP-OTI uses only one secret key PSK (Pre-Shared Key) shared between the client and authentication server. This key is used to derive two different keys K1 and K2, like in EAP-EHash [6]. The client and the authentication server must have these two keys to prove their identities. Thus, it enhances security. Indeed, to impersonate a legitimate entity, the intruder must have two keys and not just one.

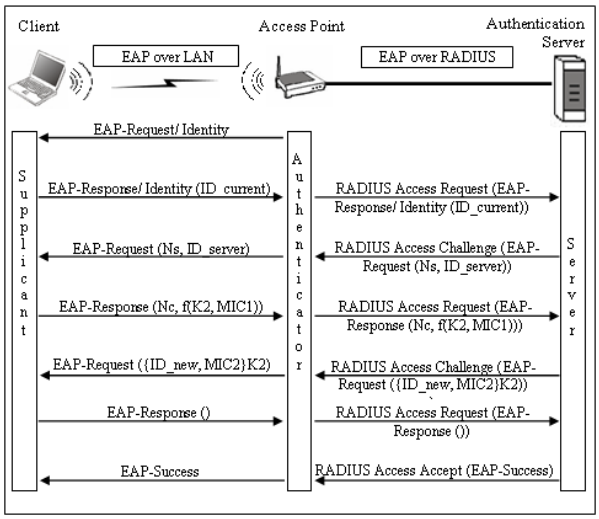

Fig. 4 EAP-OTI exchange.

To ensure the protection of client identity without using digital certificates, which are not very suitable for wireless networks. For this reason, we introduce in the EAP-OTI method a new mechanism that is based on the use of disposable identities. Indeed, each client has an identity that will be used only once, a different identity is used at each login. Figure 4 illustrates the operation of EAP-OTI.

After receiving the identification request from the access point, the server sends its current identity (ID_current). Then, it sends the client a random number Ns combined with its identity (ID_server). The client responds with a message that contains a random number Nc and the hash value of MIC1 using the key K2 and a one-way hash function f. The MIC1 (Message Integrity Check) is a hash value calculated as follows: MIC1 = f (K1, ID_server & Ns & Nc) such:

- f denotes a one-way hash function (such as HMAC-SHA-1 or HMAC-MD5).
- K1 and K2 are two session keys derived from PSK as K1 = f(PSK, ID_server & Ns & Nc) and K2 = f(PSK, ID_current & Nc & Ns) where & denotes the concatenation.

Then the server calculates the same keys as the client then checks the MIC1 received. In case of success, the client is authenticated to the server. Then, the server assigns a new identity to the client (ID_new) and sends to the client the result of encryption of MIC2 combined with ID_new, using the key K2. The MIC2 is calculated as follows: MIC2 = f (K1, ID_new & Nc).

In his turn, the client calculates the MIC2 and compares it with the one received. If they match, the server is authenticated to the client.

Note that the MIC1 and MIC2 serve to prove respectively the identity of client and server. To make dictionary attack more difficult, the MIC1 and MIC2 are respectively hashed and encrypted with another key K2.

3.1. Management of identity

Since the private key is often stored in an encrypted file, only the user knows the password to decrypt this file and then to recover its private key and use it. In our method EAP-OTI, at each session the new identity received from the server must also be stored in this file.

It would be preferable to consider the use of smart cards to eliminate vulnerabilities of the storage of private key and the complexity of storage and management of identity. Indeed, a smart card provides data security and mobility.

3.2. Comparison between EAP authentication methods

The Table 1 gives a comparison of aforementioned methods (EAP-MD5, EAP-TLS and PEAP) [15] [8] and the proposed method. This comparison highlights some

performances of EAP-OTI like: its easiness of deployment without the need for PKI unlike EAP-TLS and PEAP, its execution speed vis-à-vis EAP-TLS and PEAP. This speed is due to the use of symmetric cryptography and one way hash functions and therefore requires less processing. What's more, its assurance to authenticate both client and server unlike EAP-MD5 which authenticate only client and its guarantee to connect without revealing the client's identity.

Table 1: Comparison between methods

EAP method **Propriety**	EAP-MD5	EAP-TLS	PEAP	EAP-OTI
Mutual authentication	No	Yes	Yes	Yes
No computational burden	Yes	No	No	Yes
PKI required	No	Yes	Yes	No
Protection of client identity	No	No	Yes	Yes
Security	Weak	Strong	Strong	Strong

4. Security Analysis of EAP-OTI

In this section, we discuss the main possible attacks that may be led against EAP-OTI, and we will show that our method resists these attacks.

Eavesdropping attack This attack consists of intercepting all messages exchanged between client and server. Thus, an attacker can extract from the intercepted traffic a secret, as a password that circulates in the clear. It is impossible to carry out this attack against the EAP-OTI since no secrets are transmitted in the clear over the network.

Replay attack This attack consists of sending in a communication, messages intercepted during an earlier communication. EAP-OTI is robust against this attack. Indeed, the random number Nc and Ns ensure the freshness of exchanged messages.

Dictionary attack This attack consisting of eavesdropping the network to have access to both plaintext and corresponding ciphertext and then testing a set of keys, one after another, hoping to discover the key used. EAP-OTI is robust against this type of attack. Indeed, the corresponding hash value of the plaintext is itself hashed or encrypted with another key K2. In addition, the pre-shared key is not used directly, it serves only to generate two different keys at each login session. These two keys will be used during authentication.

Known key attack Even if the session keys K1 and K2 are compromised, the known key attack still fails because the intruder has to know the pre-shared key to compute the new session keys K1 and K2.

5. Conclusion

Given that EAP authentication methods are widely used in wired and wireless networks and that most of these methods do not offer the protection of client identity, which has become a highly required additional service, we propose an EAP authentication method called EAP-OTI, which meets this need. Indeed, EAP-OTI is based on the use of disposable identities; in each session, the client connects with a different identity that prevents an intruder who eavesdrops on the network to discover his identity.

EAP-OTI has other advantages such as its execution speed which is due to the use of symmetric cryptography, its mutual authentication and resistance against attacks such as replay and dictionary attacks.

In a following phase, we proceed to a more formal analysis of EAP-OTI, using for example CASPER [10], EVA [11] or AVISPA [5], in order to more prove its security properties.

References

[1] B. Aboba and D. Simon, "PPP EAP TLS Authentication Protocol," RFC 2716, October 1999.
[2] B. Aboba and P. Calhoun, "RADIUS (Remote Authentication Dial In User Service) Support For Extensible Authentication Protocol (EAP)," IETF RFC 3579, September 2003.
[3] B. Aboba, L. Blunk, J. Vollbrecht, J. Carlson and H. Levkowetz, "Extensible Authentication Protocol (EAP)," IETF RFC 3748, June 2004.
[4] B. Aboba, D. Simon and P. Eronen, "Extensible Authentication Protocol (EAP) Key Management Framework," IETF RFC 5247, August 2008.
[5] AVISPA project, http://avispa-project.org, 2006.
[6] O. Cheikhrouhou, M. Laurent, A. Ben Abdallah and M. Ben Jemaa, "An EAP-EHash authentication method adapted to resource constrained terminals," Annals of telecommunications, DOI 10.1007/s12243-009-0135-9, November 2009.
[7] S. Convery, D. Miller and S. Sundaralingam, "Cisco SAFE: Wireless LAN security in depth," Cisco Systems, 2003.
[8] R. Dantu, G. Clothier and A. Atri, "EAP methods for wireless networks," Computer Standards & Interfaces, vol.29, no.3, pp. 289-301, March 2007.
[9] T. Dierks and C. Allen, "The TLS protocol version 1.0," IETF RFC 2246, January 1999.
[10] B. Donovan, P. Norris and G. Lowe, "Analyzing a library of security protocols using Casper and FDR," In Proceedings of the Workshop on Formal Methods and Security Protocols, Trento, Italy, 1999.
[11] EVA RNTL project, Explication et Vérification Automatique de protocoles cryptographiques, http://www-eva.imag.fr, 2001.
[12] P. Funk and S. Blake-Wilson, "EAP Tunneled TLS Authentication Protocol (EAP-TTLS)," draft-ietf-pppext-eap-ttls-05.txt, internet draft, July 2004.
[13] M. S. Gast, 802.11 wireless network, O'Reilly, 2005.

[14] IEEE 802.1X-2004, "IEEE Standard for Local and Metropolitan Area Networks Port-Based Network Access Control", IEEE, Piscataway, 2004.
[15] J. Lei, X. Fu, D. Hogrefe and J. Tan, "Comparative Studies on Authentication and Key Exchange Methods for 802.11 Wireless LAN," Computers & Security, Elsevier, ISSN 0167-4048, vol.26, no.5, pp. 401-409, August 2007.
[16] A. Palekar, D. Simon, G. Zorn, J. Salowey, H. Zhou and S. Josefsson, Protected EAP Protocol (PEAP) version 2, draft-josefsson-pppext-eap-tls-eap-07.txt, internet draft, October 2003.
[17] B. Schneier, Applied Cryptography, Second edition, John Wiley & Sons, 1996.
[18] J. Snyder, What is 802.1X?, Network World Global Test Alliance, www.networkworld.com, May 2002.

Miloud Ait Hemad received the B. S. in the Department of Computer Science, Cadi Ayyad University in 2001. He has his master in the field of networks and telecommunication at University the Chouaib Doukkali at El Jadida in 2005. He is currently a Ph.D. candidate of the Department of Computer Science at Cadi Ayyad University, Marrakech, Morocco. His main field of research interest is the authentication in wireless networks.

Moulay Ahmed El Kiram is research professor at the Faculty of Science Semlalia, Cadi Ayyad University of Marrakech. He received his DES in Computer Science in 1997 at Mohammed V University of Rabat. EL KIRAM specializes in Security and network communication. His areas of interest include Authentication, particularly in multicast environment.

Azzeddine Lazrek is full Professor in Computer Science at Cadi Ayyad University in Marrakesh. He holds a Ph.D. in Computer Science from Lorraine Polytechnic National Institute in France, awarded in 1988, and a State Doctorate Morocco awarded in 2002. Prof. Lazrek specializes in communication through multilingual multimedia e-documents. His areas of interest include multimedia information processing and its applications, particularly, to electronic publishing, digital typography, Arabic processing, and history of sciences. He is in charge of the Information Systems and Communication Networks Research Team and the Multilingual Scientific E-Document Processing Research Group. He is an Invited Expert at W3C. He leads a multilingual e-document composition project with some international organizations. He contributes to scientific journals and is a member of several national and international scientific associations. Email: lazrek@ucam.ac.ma