Extensible Authentication Protocol (EAP) Security Issues

Samuel Sotillo, *Dept. of Technology Systems, East Carolina University*

***Abstract*—This document describes the Extensible Authentication Protocol and several of its best-known security issues. It introduces the basic functionality of EAP as well as of several of its implementations. It discusses several vulnerabilities that affect EAP methods.**

***Index Terms*—EAP, Network, Security, Wireless, PPP, Authentication, WLAN, WPA, WPA2, TLS, TTLS, EAP-TLS, EAP-TTLS, LEAP, SEAPv0, SEAPv1, CHAP, EAP-FAST, EAP-PSK**

I. INTRODUCTION

THIS document presents an overview on some security issues that affect the Extensible Authentication Protocol as defined by the IETF RFC 3748 [1].

This document introduces some basic concepts about EAP, its basic architecture and functionality. Also, it describes some of the many implementations of EAP. Finally, it discusses some security issues associated to diverse EAP implementations.

II. EXTENSIBLE AUTHENTICATION PROTOCOL OVERVIEW

A. Origen

The Extensible Authentication Protocol (EAP) is an Internet standard that provides an infrastructure for network access clients and authentication servers. It is described in the RFC 3748 [1].

EAP is not and does not specify any specific authentication mechanism. Instead, EAP procures a framework that provides some common functions and a negotiation of the desired authentication mechanism [2].

Originally, EAP was created as an extension to PPP that allows for the development of arbitrary plug-in modules for current and future authentication methods and technologies [3]. Today, EAP is most often used in wireless LANs [2]. Particularly, two wireless standards, WPA and WPA2, which have officially adopted several EAP methods as their main authentication mechanisms [2].

B. EAP Basics

Since EAP was originally developed for PPP, the best way to explain its operation is using a PPP-based example. Figure 1 shows a typical stack for EAP authentication of PPP-based communication. As shown in Figure 1, with EAP, PPP peers do not choose a specific authentication mechanism during the link establishment phase of the PPP connection; instead, they negotiate to perform EAP during the connection authentication phase. Once the connection authentication phase is completed, the peers negotiate the use of a specific EAP authentication method. At this point, the conversation between the peers consists mostly of requests and responses for authentication information (in Figure 1, the conversation is shown as a solid line connecting the EAP boxes on each corresponding stack). After that, EAP allows for an open-ended exchange of information between the access client and the authenticating server that varies depending on the connection parameters involved (in Figure 1, the authenticating server uses the RADIUS protocol). In essence, the EAP method determines the length and details of the

Manuscript completed November 29, 2007. This work was completed as a project requirement for the course ICTN 4010, section 601.

Samuel Sotillo is a senior student at East Carolina University, Dept. of Technology, College of Technology and Computer Science.

conversation between authenticating peers [1] [3].

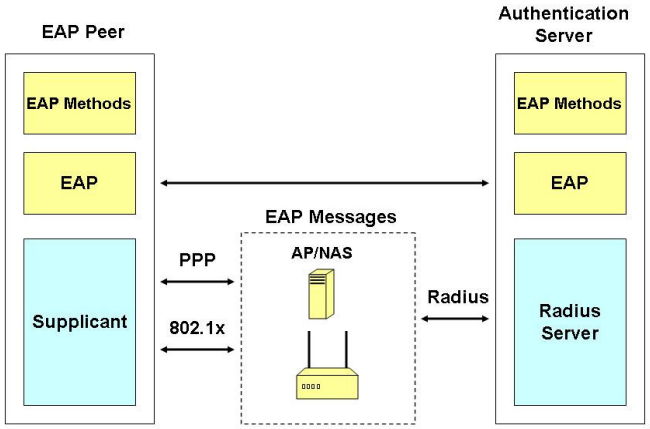

Fig. 1. Typical stack for EAP authentication of PPP-based communication. PPP peers do not choose a specific authentication mechanism during the link establishment phase of the PPP connection; instead, they negotiate to perform EAP during the connection authentication phase.

The main components of EAP, as shown in Figure 1, are the following:

--*EAP peer/Access Clients*: Computers attempting to access network resources.

--*EAP authenticator*: An access point (AP) or network access server (NAS) requiring EAP authentication before granting access to a network resource.

--*Authentication server*: A server computer that negotiates the use of a specific EAP method with an EAP Access Client. It also validates EAP peers' credentials and authorizes access to network resources. In Figure 1, the authentication server is a Remote Authentication Dial-In User Service (RADIUS) server. [3]

Notice from Figure 1 that the EAP peer and, in the case of wireless LANs, the EAP *authenticator* both send EAP messages using a *supplicant* and a data link layer transport protocol such as PPP—or, for wireless LANs, the IEEE 802.1X infrastructure.

TABLE I
EAP VARIANTS

PROTOCOL	DESCRIPTION
Lightweight EAP (LEAP)	Cisco proprietary—it is a modified version of MS-CHAP
EAP-TLS	Based on Transport Layer Security—which is based on the Public Key Infrastructure (PKI)
EAP-MD5	Based on MD5 hash
EAP-PSK	Based on pre-shared keys (PSK)
EAP-TTLS	Based on Tunneled Transport Layer Security (TTLS)—most widely used
EAP-IKE2	Based on Internet Key Exchange Protocol version 2 (IKEv2)—it uses asymmetric/symmetric key pairs and passwords
PEAPv0/EAP-MSCHAPv2	Similar in design to EAP-TTLS—however, it only requires a server-side PKI certificate—second most used
PEAPv1/EAP-GTC	Cisco variant—based on Generic Token Card (GTC) authentication
EAP-FAST	Cisco proprietary replacement for LEAP—based on Flexible Authentication via Secure Tunneling (FAST)
EAP-SIM	For Global System for Mobile Communications (GSM)—based on Subscriber Identity Module (SIM), a variant of PEAP for GSM
EAP-AKA	Based on Universal Mobile Telecommunications System (UMTS) Subscriber Identity Module (USIM) Authentication and Key Agreement (AKA)
EAP-GSS	Based on Generic Security Service (GSS)—it uses Kerberos

See [2] for more details.

A supplicant is a software component that uses EAP to authenticate network access but that handles the actual data exchange [3]. In the example shown in Figure 1, both the EAP authenticator and the authentication server send EAP messages using RADIUS. As a result, EAP messages are actually exchanged between the EAP components on the EAP client and the authentication server.

In a word, EAP provides the highest flexibility because it allows vendors to create more secure

authentication schemes that can be plugged in later on, as required [1] [2].

C. EAP Implementations

There are quite a few implementations available of EAP. Table 1 summarizes some of the most important.

In wireless LANs, on one hand, Wi-Fi Protected Access (WPA) originally recommended EAP-PSK—mainly, because home/small office applications were not required to support IEEE 802.1x authentication [4]. EAP-PSK is based on pre-shared keys—where a *shared secret key* is shared in advance between the two parties, using some secure channel [5]. EAP-PSK is a very lightweight protocol—it only requires four messages to complete the authentication stage [4]. Regardless of EAP-PSK simplicity and economy, WPA later recommended using EAP-TLS and EAP-TTLS for increased security [4].

On the other hand, IEEE 802.11i (also known as WPA2) requires enterprise-level security. Therefore, in addition to EAP-TLS/TTLS, WPA2 devices also support PEAPv0, PEAPv1 and other open standards [4] [6].

Currently, the two most used EAP implementations are EAP-TTLS and PEAPv0. Both are extensively supported by most commonly used operating systems (Microsoft, Mac OS, and Linux) as well as by most network appliance vendors [4]. Table 2 summarizes the most common EAP methods used by the different types of network access.

TABLE 2
EAP METHODS FOR DIFFERENT TYPES OF NETWORK ACCESS

TYPE OF NETWORK ACCESS	AVAILABLE EAP METHODS
Dial-up remote access or site-to-site connections	EAP-MD5 CHAP, EAP-TLS
Virtual private network (VPN) remote access or site-to-site connections	EAP-MD5 CHAP, EAP-TLS
802.1X authentication to an authenticating switch (wired)	EAP-MD5 CHAP, PEAP-MS-CHAP v2, EAP-TLS, PEAP-TLS, EAP-FAST
802.1X authentication to a wireless access point (AP)	PEAP-MS-CHAP v2, EAP-TLS, PEAP-TLS, EAP-TTLS

See [2] for more details.

III. SECURITY ISSUES

As mentioned before, EAP is a standard that provides an infrastructure for network access clients and authentication servers. EAP does not specify the authentication mechanism itself but the way it is negotiated by the communicating parties. Consequently, EAP has no security issues in itself.

In contrast, each EAP implementation stipulates some specific valid authentication methods. Consequently, EAP implementations may furnish with security vulnerabilities. The following paragraphs summarize some of the most common security issues associated to the different EAP implementations.

A. Dictionary Attacks

A dictionary attack is a technique for defeating a code or authentication mechanism by trying each word from a dictionary—a list of common words—and encoding it the same way the original passphrase was encoded [6] [7]. Dictionary attacks differ from brute-force attack on the fact that only the most likely words are tried [7].

Several EAP implementations are vulnerable to dictionary attacks. For instance, Cisco's Lightweight EAP (LEAP) relies on a shared secret—the user's logon password. Systems lacking strong password policies can easily be compromised with dictionary attacks [8]. As a consequence of this vulnerability, Cisco developed EAP-FAST to provide better protection against dictionary attacks. However, EAP-FAST is vulnerable to MitM attacks

as explained below [2].

EAP-GSS is another example of an EAP implementation vulnerable to dictionary attacks. GSS relies on the Kerberos protocol, which is itself vulnerable to dictionary attacks [9].

B. Plaintext Attacks

EAP implementations that rely on clear-text authentication using RADIUS (even within a protected tunnel) are vulnerable to known-plaintext attacks [2]. In a known-plaintext attack (KPA), the attacker uses samples of both the plaintext and its encrypted version to reveal further secret information such as the secret encryption key [10].

EAP-IKE2 and EAP-TTLS are examples of EAP implementations that may use password-based authentication (PAP) and therefore are vulnerable to this type of attacks [11]. In PAP-based authentication, passwords are transmitted unencrypted.

C. Man-in-the-middle Attacks (MitM)

Tunneling protocols such as TLS and TTLS offer a server-authenticated tunnel that secures both the authentication method and the user's identity [12]. Unfortunately, original implementations of EAP that are based on these protocols may also be vulnerable to man-in-the-middle (MitM) attacks. In a MitM attack, a rogue client assumes the identities of both the client and the server in order to intercept communication from one device and send a tainted one to the other device [6].

The main reasons for these vulnerabilities are:

--Re-using of legacy client authentication protocols that run inside the authenticated tunnel.

--Clients cannot or do not properly authenticate the server—even when the authentication protocol is used within a supposedly server-authenticated tunnel. [12]

In a nutshell, the problem arises when the client re-use a legacy authentication mechanism (for instance, plain EAP) vulnerable to MitM within the secured tunnel. Also, if the client does not support mutual authentication or some form of session key agreement, then the backend server cannot be sure that the identity of the client using the legacy authentication protocol and the identity of the client endpoint are the same [12].

The same MitM vulnerability has also been found on the first PEAPv0 implementation for Microsoft Windows XP (SP1). The problem was later corrected for SP2 [11].

Another EAP implementation vulnerable to MitM attacks is EAP-FAST, the protocol Cisco developed as replacement for LEAP. EAP-FAST was designed to address the vulnerabilities of LEAP (see previous discussion on dictionary attacks) while keeping its simplicity and economy. However, the (automatic) private key provisioning mechanism reuse legacy authentication methods that makes the authentication model vulnerable to the same MitM attacks all other tunneled implementations are exposed to [2].

Fortunately, several solutions to the problem of MitM attacks have been suggested since the original research on this issue was first published in 2003 [11].

D. Ciphertext attacks

Theoretically, EAP-SIM improves the original GSM security model—based on a pre-shared key and challenge-response mechanism. The original GSM standard uses A5/1 and A5/2 stream ciphers with key length of 64 bits [13] [15]. EAP-SIM improves the original GSM standard by increasing the key length to 128 bits. Unfortunately, the way the new 128-bit key is generated has been shown to be defective [14]. Rather than being 128-bit long, the resulting key has an effective key length of 64 bits only.

The probability of decrypting a 64-bit long key is approx.:

$\frac{1}{2^{64}} = 5.4210 \times 10^{-20}$, which is considerably larger than the probability for a 128-bit long key: $\frac{1}{2^{128}} = 2.9387 \times 10^{-39}$.

It is well known that the larger the probability, the lower the time complexity of the attack; which means that a smaller amount of time is needed for the attacker being able to expose significant information [15].

In a known-plaintext attack, the attacker uses samples of both the plaintext and its ciphertext (encrypted version) to disclose further secret information such as a secret key [13]. A weak key makes the disclosing process a lot easier to accomplish. Today, attackers have access to increasing computing power. Consequently, the pressure on weak cryptosystems such as GSM increases as well. 64-bit long systems are difficult to justify today, the same way 128-bit long systems might be likely difficult to justify in the near future.

IV. Conclusion

The Extensible Authentication Protocol (EAP) is an Internet standard that provides an infrastructure for network clients and authentication servers. In a nutshell, EAP provides a flexible mechanism for hosting authenticating plug-in modules for current and future authentication methods.

EAP has been implemented based on several well-known authentication technologies. For instance, there are versions of EAP built on top of PSK, TLS, TTLS, GSM, AKA, among many others.

Unfortunately, some of these implementations present significant security vulnerabilities such as exposure to dictionary attacks, plaintext attacks, known-ciphertext attacks, and man-in-the-middle attacks.

To conclude, EAP is a highly flexible infrastructure for secure network access authentication. Thus far, it has been implemented in many flavors and colors, based on well known authentication schemes, some of them with important security weaknesses.

References

[1] B. Aboda, L. Blunk, J. Vollbrecht, J. Carlson, and H. Levkowetz. (2004, June). *Extensible Authentication Protocol (EAP)*. [Online] Available: http://www.ietf.org/rfc/rfc3748.txt. IETF RFC 3748.

[2] "Extensible Authentication Protocol." (2007, November 18). In *Wikipedia, The Free Encyclopedia*. [Online] Available: http://en.wikipedia.org/w/index.php?title=Extensible_Authentication_Protocol&oldid=174317627.

[3] "Extensible Authentication Protocol Overview." (n.d). In *Microsoft TechNet*. [Online] Available: http://www.microsoft.com/technet/network/eap/eap.mspx.

[4] "Wi-Fi Protected Access." (2007, November 19). In *Wikipedia, The Free Encyclopedia*. [Online] Available: http://en.wikipedia.org/wiki/Wi-Fi_Protected_Access.

[5] "Pre-shared key." (2007, August 29). In *Wikipedia, The Free Encyclopedia*. [Online] Available: http://en.wikipedia.org/wiki/Pre-shared_key.

[6] M. Ciampa. (2006). *CWNA Guide to Wireless LANs*. Boston: Thomson.

[7] "Dictionary attack." (2007, September 27). In *Wikipedia, The Free Encyclopedia*. [Online] Available: http://en.wikipedia.org/wiki/Dictionary_attack.

[8] "Cisco Security Notice: Dictionary Attack on Cisco LEAP Vulnerability." (2004, July 19). In *Cisco Security Notice*. [Online] Available: http://www.cisco.com/warp/public/707/cisco-sn-20030802-leap.pdf.

[9] T. Wu. (1999). *A Real-World Analysis of Kerberos Password Security*. In *Symposium on Network and Distributed Systems Security (NDSS '99), San Diego, CA*. [Online] Available: http://www.isoc.org/isoc/conferences/ndss/99/proceedings/papers/wu.pdf.

[10] "Known-plaintext attack." (2007, November 1). In *Wikipedia, The Free Encyclopedia*. [Online] Available: http://en.wikipedia.org/wiki/Known-plaintext_attack.

[11] B. Aboda. (2006, June 13). *The Unofficial 802.11 Security Web Page*. [Online] Available: http://www.drizzle.com/~aboda/IEEE/.

[12] N. Asokan, V. Niemi, and K. Nyberg. (2003, June). "Man-in-the-Middle in Tunneled Authentication," In *11th Security Protocols Workshop, Cambridge, United Kingdom*. [Online] Available: http://www.saunalahti.fi/~asokan/research/tunnel_extab_final.pdf.

[13] "GSM." (2007, November 10). In *Wikipedia, The Free Encyclopedia*. [Online] Available: http://en.wikipedia.org/wiki/GSM.

[14] S. Patel. (2003, May 29). "Analysis of EAP-SIM Session Key agreement." In *www.Drizzle.com*. [Online] Available: http://www.drizzle.com/~aboda/EAP/AnalyisOfEAP.pdf.

[15] E. Barkan, E. Biham, N. Keller. (2003). "Instant Ciphertext-only Cryptanalysis of GSM Encrypted Communication." In *Crypto2003*. [Online] Available: http://www.springerlink.com/index/ythkwv4gfq0fr5j4.pdf.