

# 802.1X

Ronny Haryanto <ronny@haryan.to>

October 2004

## Abstract

The ever increasing complexity of today's applications and services demands better security design and implementations. One of the most popular new technologies today is wireless LAN. It has different characteristics compared to the wired LAN so it requires new ways to provide proper security and authentication for wireless solutions. There are some early workarounds that tried to address wireless LAN security, one of the most popular was Wired Equivalent Privacy (WEP) which proved to be inadequate. 802.1X is the authentication part of the solution to the overall security problem.

The IEEE 802.1X is a standard that tries to address authentication issues by utilising an authentication protocol called Extensible Authentication Protocol (EAP) encapsulated over LANs, or EAPOL for short, that in turn utilizes many existing protocols for authentication such as PPP, MD5, TLS, CHAP, and RADIUS. 802.1X can be used for wireless as well as wired LANs.

This paper will discuss EAP and 802.1X in more details, what it is, how it works, what components are needed and how they interact with each other, where it can be used, some examples, and more. This paper could also provide some understanding of the 802.1X and EAP technology to network designers so that they can utilize it in their wired and/or wireless solutions.

# Contents

<b>1</b>	<b>Introduction to 802.1X</b>	<b>4</b>
<b>2</b>	<b>802.1X Components</b>	<b>6</b>
2.1	Supplicant . . . . .	6
2.2	Authenticator . . . . .	7
2.3	Authentication Server . . . . .	7
<b>3</b>	<b>EAP and EAPOL</b>	<b>8</b>
<b>4</b>	<b>How 802.1X Works</b>	<b>10</b>
<b>5</b>	<b>802.1X EAP Types</b>	<b>13</b>
5.1	EAP-MD5 . . . . .	14
5.2	Cisco LEAP . . . . .	15
5.3	EAP-TLS . . . . .	15
5.4	EAP-TTLS . . . . .	16
5.5	PEAP . . . . .	16
5.6	Other Types . . . . .	17
<b>6</b>	<b>802.1X Flaws and Solutions</b>	<b>17</b>
<b>7</b>	<b>Conclusion</b>	<b>18</b>
<b>8</b>	<b>Glossary</b>	<b>19</b>
	<b>References</b>	<b>21</b>

## List of Figures

1	Basic 802.1X Components . . . . .	6
2	EAP Packet Format [ILJ98, Pet04] . . . . .	9
3	EAPOL Frame Format for 802.3 Ethernet [Pet04, IEE01] . . . . .	9
4	EAPOL and EAP conversation boundaries . . . . .	10
5	Typical Successful 802.1X Transaction using OTP . . . . .	11
6	Typical Failed 802.1X Transaction using OTP . . . . .	13

# 1 Introduction to 802.1X

The IEEE 802.1X [IEE01] is a standard for Port-Based Network Access Control by means of layer 2 authentication. It can be applied wherever the notion of a port can be abstracted in a IEEE 802 (Ethernet) network. The most common examples of port-based network access are access to wireless LANs via wireless access points, and access to wired LANs via a workgroup switch. By default the ports are in a “closed” or “unauthorized” state which means that no access are allowed to go through even though the physical connection has been established. Only after the user or device requesting access has authenticated themselves then the port state is changed to “open” or “authorized” which means that normal traffic are allowed to go through the port.

Before 802.1X, other techniques such as MAC-based access control and shared keys (WEP) [Wik04] were used to manage access control. Some people do not worry about the lower level security and leave it up to the higher layer protocols to handle it, such as using VPN and SSL. Some techniques are reactionary in nature, allow everything (or most) by default while monitoring for intrusions with an IDS then take actions when an intrusion happens. [Bru02] The most important problems [Bru02, Ste02] with one or more of these methods are:

1. manageability: cumbersome to administer and manage,
2. scalability: not scalable, and
3. security: provide little protection thus giving users a false sense of security.

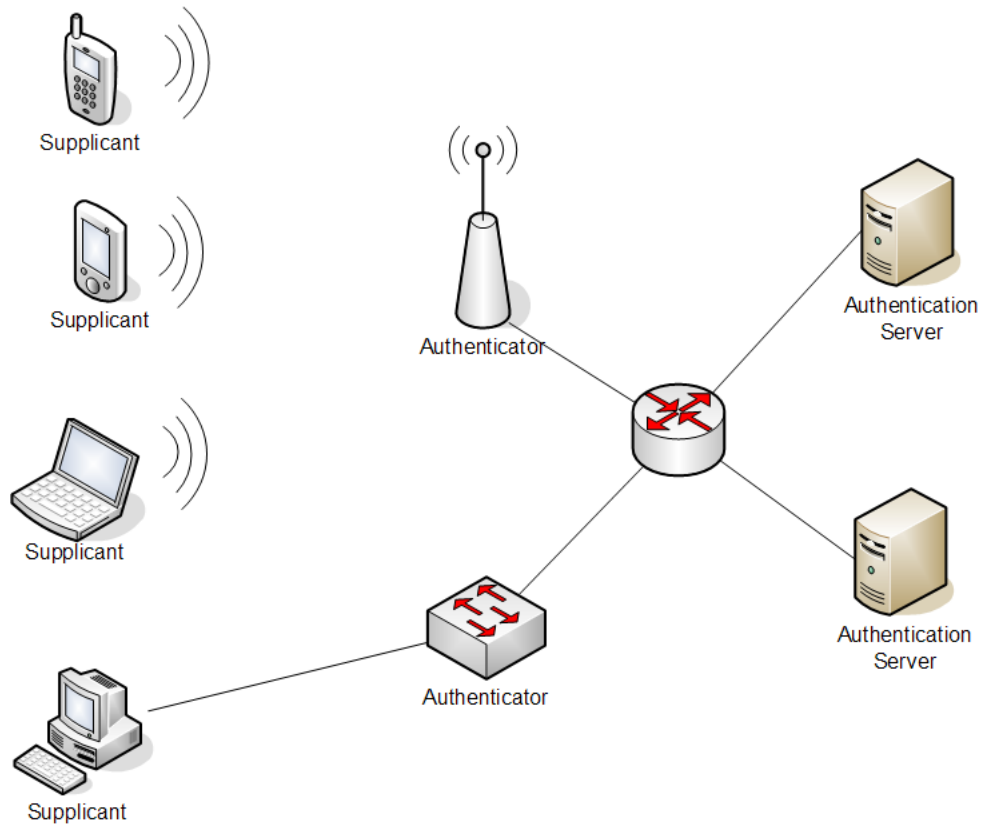
802.1X is an effort to address these problems by providing a modular, scalable and centralized design for port-based network access control. 802.1X,

together with WPA and AES, is an integral part of the recently finalized [Eri04] 802.11i standard dubbed “WPA2” which is an effort to address most (if not all) of the known security issues for 802.11 wireless LANs.

When 802.1X was originally designed, it was intended for use over wired LANs. Only recently 802.1X has gained popularity in the wireless community. There are several additional requirements for 802.1X to be used in wireless LAN environments. There need to be some mechanism to prevent eavesdropping of sensitive authentication traffic (such as clear-text passwords), and also a mutual authentication mechanism to ensure that the user can be certain that she is connecting to the right network and, vice versa, the network can be sure that the user is, in fact, a valid and legitimate user. Although security is mentioned, it is important to note that 802.1X only defines *authentication*, it does not define anything about the security of the data traffic *after* the authentication phase is successful. However, 802.1X allows [IEE01, section 8.4.9] for optional key exchange between the supplicant (see 2.1 on the following page) and the authenticator (see 2.2 on page 7) for further traffic encryption, for example the WEP or WPA key used in 802.11 wireless LANs. This capability proves useful for WLAN security because the keys can be generated dynamically and different for each session, therefore, it minimizes the security risks caused by the key being compromised. The discussion on that subject is, however, outside the scope of this document.

## 2 802.1X Components

Figure 1: Basic 802.1X Components



### 2.1 Supplicant

The 802.1X standard [IEE01] defines supplicant as “an entity at one end of a point-to-point LAN segment that is being authenticated by an authenticator attached to the other end of that link”. It can be a wireless (802.11) laptop, 802.11-capable PDA, or a desktop/workstation computer. The supplicant is the device that needs to gain access to the network, the

subject of the 802.1X authentication process. The supplicant is directly connected to the authenticator but normally not directly connected to the authentication server.

## **2.2 Authenticator**

As depicted in Figure 1 on the preceding page, commonly an authenticator is a wireless access point or a (managed) switching hub. The authenticator is the device that offers the services to the end device (supplicant) and “facilitates authentication of the supplicant” according to the 802.1X standard [IEE01].

It simply acts as an intermediary device or a proxy, passing authentication information traffic back and forth between the supplicant and the authentication server on the supplicant’s behalf. [Jim03] Hence the authenticator does not need to be powerful because all the processing happens in the supplicant and the authentication server. [Joe02]

Some authenticator devices could be configured to have multiple authentication servers, it will talk to one authentication server normally and will fall back to one of the other authentication servers when the primary server goes down. This is useful to provide higher availability, however, normally the authentication servers must be synchronized to each other.

## **2.3 Authentication Server**

The authentication server is the device that will do the actual authentication, authorization and accounting (AAA). It “determines, from the credentials provided by the supplicant, whether the supplicant is authorized to access the services provided by the authenticator.” [IEE01]

The 802.1X standard does not specify which authentication server must be used. It is typically a RADIUS server. Other type of AAA server protocol is Diameter (<http://www.diameter.org>). It has access to some credential (e.g. user/password, certificates) database(s), normally supports several authentication mechanisms such as PAP and CHAP, and supports management mechanisms such as SNMP. More information about RADIUS could be obtained from <http://en.wikipedia.org/wiki/RADIUS>.

### 3 EAP and EAPOL

Extensible Authentication Protocol, or EAP for short, is an authentication protocol originally designed as an improvement for PPP. EAP is defined in RFC 2284 [ILJ98]. EAP supports multiple authentication mechanisms (see Section 5 on page 13) such as passwords, challenge response, One-Time Password (OTP), Generic Token Card, and public-key infrastructure certificates. PPP is a point-to-point protocol, therefore, EAP is probably more suited for point-to-point communication.

Figure 2 on the following page shows the EAP Packet Format according to RFC 2284 [ILJ98]. The predefined values for the code field are:

1. Request
2. Response
3. Success
4. Failure

When the code is 1 or 2, then the first byte (8 bits) of the data field must indicate the EAP authentication type (see Section 5 on page 13).

Figure 2: EAP Packet Format [ILJ98, Pet04]

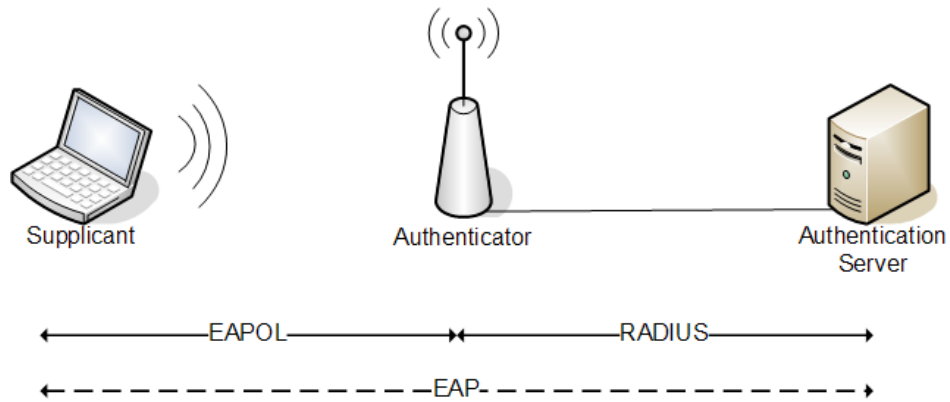
Code 1 byte	ID 1 byte	Length 2 bytes	Data 0+ bytes
----------------	--------------	-------------------	------------------

EAP over LANs, or EAPOL for short, is defined and used in the 802.1X standard [IEE01] as an encapsulation of EAP messages in Ethernet frames (layer 2) for transport over wired or wireless Ethernet-like (including token ring) LANs. 802.1X “borrows” the EAP concept and protocol from PPP but it does not use PPP at all. [Joe02] EAP is one of the possible payload data units encapsulated in EAPOL frames; indicated by a value of 0 in the Packet Type field of the EAPOL frame (see Figure 3). Simply put, EAPOL is a layer 2 wrapper to transport EAP information between the authenticator and the supplicant. If the authentication server is a RADIUS server, then the authenticator will encapsulate EAP messages in RADIUS (according to RFC 3579) to converse with the authentication server. The use of RADIUS as the protocol between the authenticator and the authentication server is optional and not mandated by the 802.1X standard [IEE01, section 8.4.7]; however, RADIUS is the *de facto* standard. The Diameter protocol could be used in place of RADIUS. See Figure 4 on the next page for illustration of the conversation boundaries.

Figure 3: EAPOL Frame Format for 802.3 Ethernet [Pet04, IEE01]

Destination MAC 6 bytes	Source MAC 6 bytes	Ethernet Type Code 2 bytes	Protocol Version 1 byte	Packet Type 1 byte	Body Length 2 bytes	Packet Body 0+ bytes
-------------------------------	--------------------------	----------------------------------	-------------------------------	--------------------------	---------------------------	-------------------------

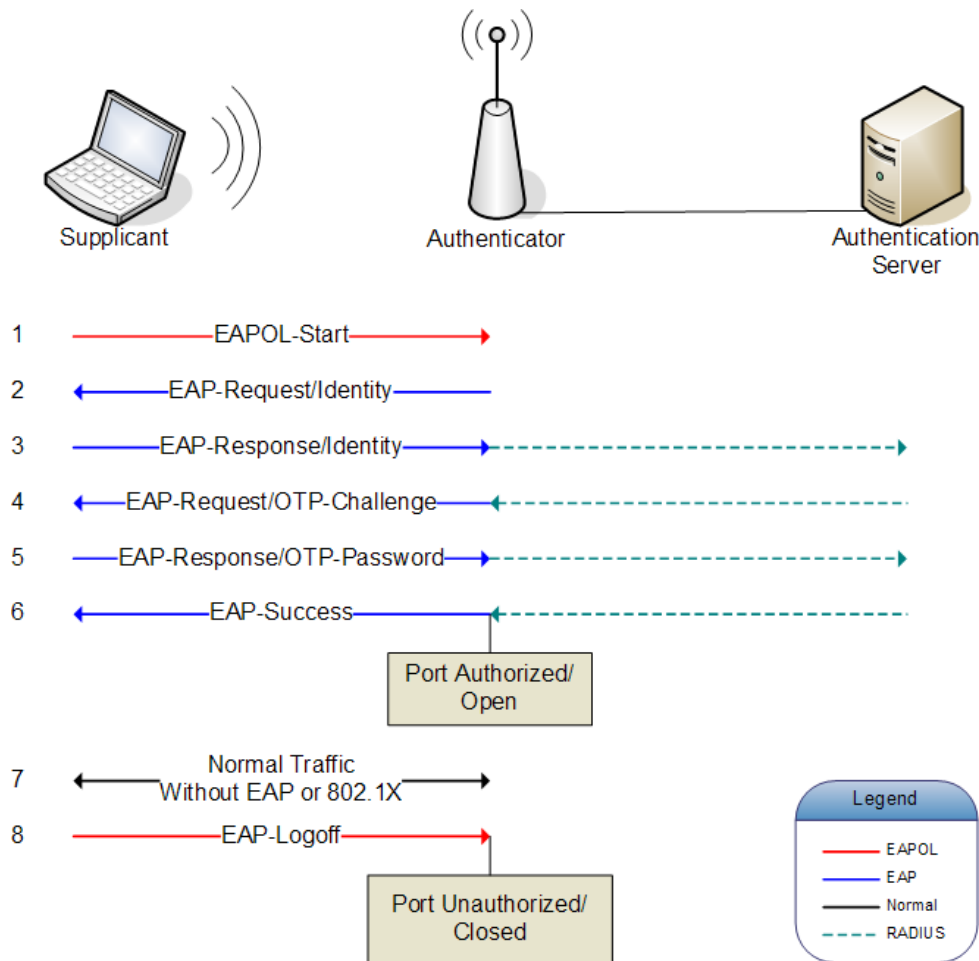
Figure 4: EAPOL and EAP conversation boundaries



## 4 How 802.1X Works

Basically the bulk of the 802.1X authentication process is exchanging EAP messages. The 802.1X standard [IEEE01, section 8.4.8] provides several examples of typical scenarios of 802.1X EAP exchanges. Figure 5 on the following page illustrates a successful authentication resulting in the port state being changed from unauthorized/closed to authorized/open and normal traffic flows through the port. The 802.1X process can be initiated either by the supplicant (by sending a EAPOL-Start message to the authenticator) or by the authenticator (by sending a EAP-Request/Identity message to the supplicant).

Figure 5: Typical Successful 802.1X Transaction using OTP

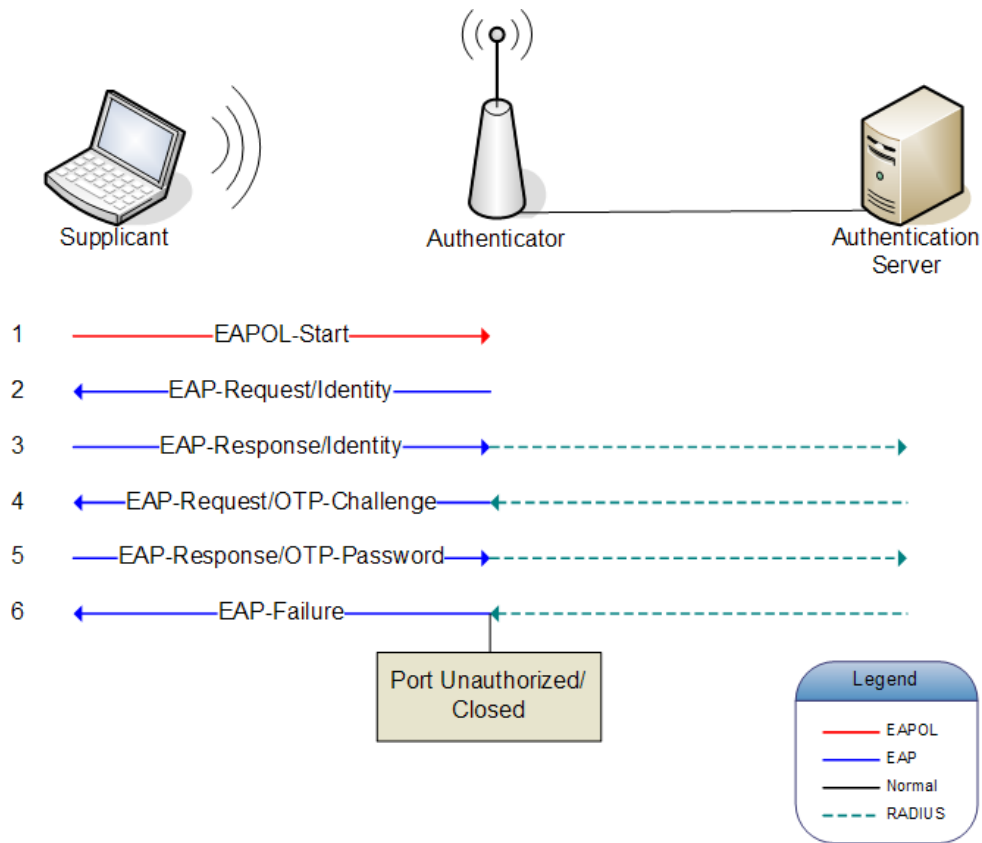


The first line (1) in Figure 5 shows that the supplicant initiated 802.1X transaction by sending a EAPOL-Start message to the authenticator. This step is omitted when the authenticator initiated the transaction by sending the EAP-Request/Identity (line 2) directly to the supplicant. The supplicant then replies with its identity (line 3) to the supplicant, the supplicant will then re-encapsulate the contained EAP-Response/Identity into a suitable format (e.g. RADIUS) to be sent to the authentication server.

The dotted lines in the diagram shows this re-encapsulated EAP packets being sent to and received from the authentication server. In this example the one-time password (OTP) authentication mechanism is configured and used to perform the actual authentication (line 4 and 5), therefore, it might differ for other types of authentication. After some processing the authentication server determines whether the supplicant is granted access or not. In this case the authentication is successful, therefore, the authentication server sends a EAP-Success message back (line 6) to the supplicant (via the authenticator). The authenticator monitors for EAP-Success and EAP-Failure messages and change the port states accordingly (line 6). In our example above, it receives a EAP-Success message then changes the port state to authorized/open for the corresponding supplicant, re-encapsulates the EAP-Success message into EAPOL and then sends it to the supplicant. The supplicant receives the EAP-Success message and assumes that the authenticator has enabled the port for it to access, therefore normal traffic can go through without 802.1X nor EAP at all (line 7). The port state is changed back to unauthorized/closed when the authenticator receives a EAP-Logoff message from the supplicant (line 8).

If the authentication server sends a EAP-Failure instead of EAP-Success in line 6, then the authenticator will force the port state to unauthorized/closed, and normal traffic should not pass through the authenticator. The supplicant should not be sending any normal traffic anyway because it receives the EAP-Failure message. This is illustrated in Figure 6 on the following page.

Figure 6: Typical Failed 802.1X Transaction using OTP



## 5 802.1X EAP Types

The original EAP specification [ILJ98, section 3] only defines several types of EAP authentication including MD5-Challenge (type 4), One-Time Password (OTP; type 5), and Generic Token Card (type 6). There are several other types of EAP, which will be discussed shortly, that are available at the time of this writing. Not all vendors support every single one of them, therefore it is important to make sure that all the devices participating in

the 802.1X process support the same EAP authentication type that will be used.

## 5.1 EAP-MD5

EAP-MD5 is a UserID/Password-based authentication method, RFC 2284 specifies that it is the same as PPP CHAP protocol (RFC 1994) with MD5 selected as the hashing algorithm. CHAP is a challenge-response handshake protocol. The basic operation is as follows. The client identifies itself to the server by providing a username, the server then randomly generates a challenge string to the client. The client calculates a response to the challenge by computing the hash of the user's password combined with the challenge, then the resulting hash (i.e. the challenge response) is sent back to the server. Note that the password itself is never sent. The server maintains a database of user passwords. The server does the same process using the challenge it just sent to the client and the claimed user's password from its database, then the resulting hash is compared with the response it received. If they are the same then the handshake is complete and the authentication is successful, otherwise the client is not authenticated.

One major drawback to these types of challenge response protocol is that the passwords must be stored in plain text format in the server. Normally passwords are never stored in clear text, only the hash of the password is stored. The hashing process is irreversible or one-way; it is mathematically impossible to obtain the password from the hash.

Another drawback of password-based authentication methods is that they are prone to dictionary attack. It relies heavily on users choosing strong (not easily guessed) passwords.

EAP-MD5 is also prone to man-in-the-middle attack and session hijacking

if used in wireless LANs because it does not provide dynamic key management and it does not provide mutual authentication. See Section 6 on page 17 for more information. As a side note, Windows XP originally allowed EAP-MD5 in its 802.1X supplicant software for both wired and wireless LANs, however since Service Pack 1 it disallows EAP-MD5 to be used in wireless LANs. [Mic04]

## 5.2 Cisco LEAP

Lightweight-EAP (LEAP) from Cisco Systems (<http://www.cisco.com>) is basically an improvement to EAP-MD5 (it still uses CHAP MD5). Cisco LEAP supports dynamic key management and mutual authentication. Since LEAP is based on EAP-MD5 then it is also prone to dictionary attacks. If it is of major concern Cisco recommends [Cis04a, Cis04b] the implementation of strong password policy or using other types of EAP authentication that are not vulnerable to dictionary attacks, such as EAP-FAST (Cisco), EAP-TLS or PEAP.

## 5.3 EAP-TLS

Transport Layer Security (TLS) is a cryptographic protocol that provides a secure layer above TCP and for higher layer protocols (e.g. HTTP, SMTP, NNTP) to be transported in a secure manner. EAP-TLS is based on TLS and is defined in RFC 2716. It is not a password-based authentication method like EAP-MD5 or LEAP; it is a certificate-based authentication method. EAP-TLS requires *both* server *and* client certificates for mutual authentication. This implies that PKI must already be in place before EAP-TLS can be used. Another potential drawback of EAP-TLS is that the identity exchange process happens in clear text before the exchange of client and server certificates, so anyone that can listen to the traffic can

also learn about the identity of the users (e.g. usernames), [Mat02] and that EAP-TLS does not support fast session reconnect/re-authentication. EAP-TLS, and other TLS based EAPs, provide keying material which can be used to generate dynamic WEP keys for wireless (802.11) LAN security.

## **5.4 EAP-TTLS**

EAP-TTLS (Tunneled TLS) is an extension of EAP-TLS. It eliminates the PKI barrier of EAP-TLS by making the client certificates optional while still retaining the benefits of TLS. The TTLS operation basically happens in two stages. First, a TLS tunnel is established and the server authenticates itself to the client. Once the secure tunnel is established then the client authentication process can begin. This process, that happens inside the secure tunnel, can be any of the legacy protocols (PAP, CHAP, MSCHAP, MSCHAPv2) or even other EAP.

## **5.5 PEAP**

Protected EAP (PEAP) is a very similar method to EAP-TTLS, designed by Cisco and Microsoft. The only difference, technically, between PEAP and EAP-TTLS is that PEAP only allows other EAP process in the secure tunnel where EAP-TTLS allows legacy protocols such as plain PAP and CHAP. However, PEAP allows the EAP variant of the legacy protocols, such as EAP-MD5 and EAP-MSCHAPv2.

Most people would probably want to use PEAP or EAP-TTLS for their 802.1X EAP type because they are both less prone to the vulnerabilities previously explained in other EAP types, they are relatively easier to implement than EAP-TLS, and nowadays they are widely supported by most of the big vendors/OS/devices: Windows (native), Mac OS X (native),

Linux and other UNIX (Xsupplicant), Cisco, Microsoft IAS RADIUS, FreeRADIUS, and so on.

## 5.6 Other Types

There are other less-popular EAP types such as EAP-FAST from Cisco, EAP-SIM (Subscriber Identity Module) based, SecureID based, and EAP-AKA. They either have limited adoption by vendors, or are very domain-specific, or still very new.

## 6 802.1X Flaws and Solutions

About one year after the 802.1X standard came out, Mishra and Arbaugh from the University of Maryland published a paper [AW02] on security of 802.1X. It addresses some design flaws in 802.1X and proposed some solutions to the problems. The design flaws according to Mishra and Arbaugh are:

1. the absence of mutual authentication, which leads to Man-in-the-Middle attacks and rogue gateways, and
2. session hijacking, only relevant in shared-medium networking, for example in 802.11 wireless LANs an (802.1X) authenticated client can be disconnected from the access point by an adversary pretending to be the access point by sending a 802.11 MAC Disassociate message to the authenticated client, then the adversary pretends to be the authenticated client and sends network traffic to the actual access point.

There are other relatively minor issues as well, as pointed out by Bruce Potter [Bru02]:

1. roaming, the authentication process could take time, this could cause disruptions while moving from cell to cell in a wireless environment.
2. potentially single point of failure if only one authentication server back-end is used.

Cisco Systems responded [Cis03] to the Mishra and Arbaugh paper by providing its proposed solutions, which can be generalized to non-Cisco hardware, software and protocols. Some of the more sophisticated EAP types, such as LEAP and all the TLS based EAP: EAP-TLS, EAP-TTLS and PEAP, provide mutual authentication and dynamic keying material derivation/generation. This could be combined together with dynamic WEP key, per-packet keying, and message integrity checking in the 802.11 space to address all the issues.

The 802.1X-2001 standard is currently undergoing a revision and the resulting amendment is the upcoming 802.1aa standard from IEEE which could be obtained from <http://www.ieee802.org/1/pages/802.1aa.html> if you are a member of the IEEE or the EAP working group. This amendment should supposedly address all of the aforementioned issues with 802.1X.

## 7 Conclusion

From its original design, 802.1X was more directed towards wired LANs. However, 802.1X is becoming more popular and more important today because of the widespread use and deployments of 802.11 wireless networks and the demand for better security for wireless networks.

Due to this demand, many vendors have started adopting and supporting 802.1X in their products. To network designers and administrator this

means that 802.1X is available now. 802.1X is a key part of the new (at the time of writing) 802.11i standard for the next generation 802.11 security. A thorough understanding of how 802.1X and the various EAP types work is very crucial to the overall security of networks where 802.1X is used, especially in the 802.11 wireless world.

## 8 Glossary

More detailed explanations about the following terms (and others used throughout the document) could be obtained from search engines, e.g. Google (<http://www.google.com>), or from online encyclopedias, e.g. Wikipedia (<http://en.wikipedia.org>). RFCs could be obtained from <http://www.ietf.org/rfc/>.

**CHAP** Challenge-Handshake Authentication Protocol; defined in RFC 1334, 1994 and RFC 2794 (Microsoft CHAP v2).

**EAP** Extensible Authentication Protocol; defined in RFC 2284 and RFC 2716 for EAP TLS.

**IDS** Intrusion Detection System, e.g. Snort, Tripwire.

**IEEE** Institute of Electrical and Electronics Engineers, <http://www.ieee.org>.

**MAC Address** Media Access Control Address; every Ethernet device has this layer 2 address.

**Managed Switch** A highly configurable switch that can be configured via a management interface.

**MD5** An arbitrary-length one-way hashing/digest algorithm (128 bits) defined in RFC 1321.

**Mutual Authentication** A two-way authentication; one party authenticates itself to the other, and vice versa.

**OTP** One-Time Password; defined in RFC 1938.

**PAP** Password Authentication Protocol; defined in RFC 1334.

**PKI** Public Key Infrastructure.

**RADIUS** Remote Authentication Dial In User Service; defined in RFC 2865-2869 and RFC 3579 for EAP support in RADIUS.

**SNMP** Simple Network Management Protocol; defined in RFC 1157, RFC 1905 (v2), and RFC 3410-3418 (v3).

**SSL** Secure Sockets Layer; an encapsulation method for transporting clear-text data in an encrypted tunnel.

**TLS** Transport Layer Security; based on SSLv3; defined in RFC 2246 (v1.0).

**VPN** Virtual Private Network; a virtual private network running on top of a public network such as the Internet.

**WEP** Wired Equivalent Privacy, superseded by WPA.

**WPA** Wi-Fi Protected Access; see also 802.11i (WPA2).

## References

- [AW02] Arunesh Mishra and William A. Arbaugh. An Initial Security Analysis of the IEEE 802.1X Standard. <http://www.cs.umd.edu/~waa/1x.pdf>, 6 February 2002.
- [Bru02] Bruce Potter. 802.1x What it is, How it's broken, and How to fix it. <http://www.shmoo.com/1x/>, July 2002.
- [Cis03] Cisco Systems. Response to University of Maryland's Security Analysis. [http://www.cisco.com/en/US/products/hw/wireless/ps430/prod\\_bulletin09186a00800a9e74.html](http://www.cisco.com/en/US/products/hw/wireless/ps430/prod_bulletin09186a00800a9e74.html), 23 January 2003.
- [Cis04a] Cisco Systems. Cisco Response to Dictionary Attacks on Cisco LEAP. [http://www.cisco.com/en/US/products/hw/wireless/ps430/prod\\_bulletin09186a00801cc901.html](http://www.cisco.com/en/US/products/hw/wireless/ps430/prod_bulletin09186a00801cc901.html), 30 April 2004.
- [Cis04b] Cisco Systems. Cisco Security Notice: Dictionary Attack on Cisco LEAP Vulnerability. <http://www.cisco.com/warp/public/707/cisco-sn-20030802-leap.shtml>, 19 July 2004.
- [Eri04] Eric Griffith. 802.11i Security Specification Finalized. <http://www.wi-fiplanet.com/news/article.php/3373441>, 25 June 2004.
- [IEE01] IEEE. 802.1X Port-Based Network Access Control. <http://www.ieee802.org/1/pages/802.1x.html>, 2001.
- [ILJ98] IETF, L. Blunk, and J.Vollbrecht. RFC 2284 - PPP Extensible Authentication Protocol (EAP). <http://www.ietf.org/rfc/rfc2284.txt>, March 1998.

- [Jim03] Jim Burns. How 802.1x authentication works. <http://www.computerworld.com/mobiletopics/mobile/story/0,10801,79995,00.html>, 3 April 2003.
- [Joe02] Joel Snyder. What is 802.1x? <http://www.opus1.com/www/jms/0506whatisit.html>, 6 May 2002.
- [Mat02] Matthew Gast. A Technical Comparison of TTLS and PEAP. <http://www.oreillynet.com/pub/a/wireless/2002/10/17/peap.html>, 17 October 2002.
- [Mic04] Microsoft. Windows XP Wireless Deployment Technology and Component Overview. <https://www.microsoft.com/technet/prodtechnol/winxppro/maintain/wificomp.mspx>, 4 August 2004.
- [Pet04] Peter J. Welcher. Examining 802.1x and EAP. <http://www.enterprisenetworksandservers.com/monthly/art.php/696>, May 2004.
- [Ste02] Steve McQuerry - Cisco Press. IEEE 802.1X: Practical Port Control for Switches. <http://www.ciscopress.com/articles/article.asp?p=29600>, 4 October 2002.
- [Wik04] Wikipedia. Wired Equivalent Privacy. <http://en.wikipedia.org/wiki/WEP>, 2004.