

What is 802.1X?

Understanding what IEEE 802.1X is, its relationship to NAC, and why you should care about it means understanding three separate concepts: EAP (Extensible Authentication Protocol), IEEE 802.1X itself, and Tunnelled Authentication.

What is EAP and what does it have to do with 802.1X?

Most people are familiar with PPP, the point-to-point protocol. It's most commonly used for dial-up Internet access. PPP is also used by some ISPs for DSL and cable modem authentication, in the form of PPPoE (PPP over Ethernet). PPP is part of L2TP, a core part of Microsoft's secure remote access solution for Windows 2000.

PPP has gone far beyond its original use as a dial-up access method and is now used all over the Internet. Although PPP has many parts that make it useful in different networking environments, the part that we care about in 802.1X is the authentication piece. Before anything at Layer 3 (like IP) is established, PPP goes through an authentication phase at Layer 2. As PPP use grew, people quickly found its limitations, both in flexibility and in level of security, in the initial simple built-in authentication methods, such as PAP and CHAP.

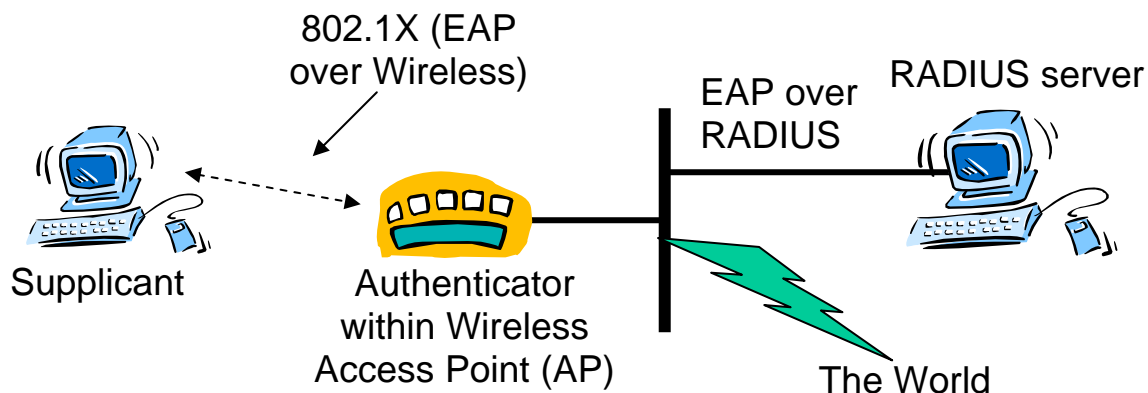
Most corporate networks want to do more than simple usernames and passwords for secure access, so a new authentication protocol, called the Extensible Authentication Protocol (EAP) was designed. EAP sits inside PPP's authentication protocol and provides a generalized framework for all sorts of authentication methods. Rather than keep changing PPP, the idea was to simply have a tunnel through the remote access server for a more powerful protocol between the user and the real authentication server. By pulling EAP out into a separate protocol, it then has the option of re-use in other environments---like 802.1X. EAP is supposed to head off proprietary authentication systems and let everything from passwords to challenge-response tokens and PKI certificates work smoothly.

What is IEEE 802.1X?

IEEE 802.1X is simply a standard for passing EAP over a wired or wireless LAN. With 802.1X, you package EAP messages in Ethernet frames. It's authentication, and nothing more.

In the wireless environment, 802.1X also describes a way for the access point and the wireless user to share and change encryption keys, and adds some messages which help smooth operations over wireless. The key change messages help solve the major security vulnerability in 802.11, the management of WEP keys. With 802.1X, WEP is brought up to an acceptable level of security for most companies.

IEEE 802.1X uses three terms that you must know. Because a wired or wireless LAN authentication has three parties involved, 802.1X created three labels for them. The user or client that wants to be authenticated is a **supplicant**. The actual server doing the authentication, typically a RADIUS server, is called **the authentication server**. And the device in between these two elements, such as a wireless access point, is called the **authenticator**. One of the key points of 802.1X is that the authenticator can be simple and dumb---all of the brains have to be in the supplicant and the authentication server. This makes 802.1X ideal for wireless access points, which typically have little memory and processing power.



How does 802.1X work, with and without NAC?

The protocol in 802.1X is called EAPOL (EAP encapsulation over LANs). It is currently defined for Ethernet-like LANs including 802.11 wireless, as well as token ring LANs like FDDI. EAPOL is not particularly sophisticated. There are a number of modes of operation, but the most common case would look something like this:

- 1) The Authenticator sends an "EAP-Request/Identity" packet to the Supplicant as soon as it detects that the link is active (*e.g.*, the supplicant system has associated with the access point).
- 2) The Supplicant sends an "EAP-Response/Identity" packet with their identity in it to the Authenticator, which is then passed on to the Authentication (RADIUS) Server encapsulated in the RADIUS protocol. When NAC is involved, these early messages will also include end-point security assessment information from the client (access requestor) to be evaluated by the Authentication Server (called Policy Decision Point in NAC).
- 3) The Authentication Server sends back a challenge to the Authenticator, such as with a token password system. The Authenticator unpacks this from RADIUS and re-packs it into EAPOL and sends it to the Supplicant. Different authentication methods will vary this message and the total number of messages. EAP supports both client-only authentication and strong mutual authentication. Only mutual authentication is considered appropriate for the wireless case. If NAC is involved, all of these messages might also help in the end-point security assessment control and communications chain.
- 4) The Supplicant responds to the challenge via the Authenticator, which passes the response onto the Authentication Server.
- 5) If the supplicant provides proper credentials, the Authentication Server responds with a success message, which is passed on to the Supplicant. The Authenticator now allows access to the LAN. In the case of NAC, access is restricted based on attributes that came back from the Authentication Server. For example, the Authenticator might switch the Supplicant to a particular VLAN using standard RADIUS attributes (as is used in the TCG-TNC and Microsoft-NAP strategies) or using private RADIUS attributes (as is used in Cisco-NAC).

EAPOL (EAP over LAN) has other message types as well. For example, when the Supplicant is finished, it can send an explicit "LOGOFF" notification to the Authenticator. 802.1X also defines a re-authentication timer, which can be used to periodically require the Supplicant to re-authenticate.

What is Tunneled Authentication?

EAP has many authentication protocols defined, but the two most important for NAC and 802.1X are TTLS (Tunneled TLS) and PEAP (Protected EAP). These differ slightly in their design, but they have a common strategy. With TTLS and PEAP, authentication is asymmetric. Both TTLS and PEAP authentication use certificates and the TLS (SSL) protocol to authenticate the server side of the 802.1X transaction and establish an encrypted tunnel. Then, within that tunnel, the client side authenticates to the server, typically using a simpler method, such as username/password or token card.

TTLS and PEAP accomplish this by packing another authentication protocol inside of the TLS tunnel when it comes time to authenticate the user. That's where the "Tunneled" in "Tunneled TLS" comes from. With TTLS, the network manager has the option of using a very simple tunneled authentication protocol, such as clear-text passwords or challenge-response passwords, or a more advanced technique, such as token-based authentication. With PEAP, there are fewer options: the tunneled authentication method is EAP itself, meaning that you can only use an EAP-defined method for authentication.

In the simple world of authentication and 802.1X, TTLS and PEAP are the preferred authentication methods because they give strong server authentication and convenient user authentication. NAC architects saw this and came up with another idea: how about using the TLS tunnel between the client and the authentication server to pass end-point security posture information? This is exactly what NAC does over an 802.1X switch or wireless access point: it uses the tunnel established for authentication as a bidirectional channel to securely carry posture information.

At least one vendor, Cisco, also has a mechanism for encapsulation of EAP over UDP (rather than in Ethernet frames), which lets you use the same EAP NAC scheme both in an 802.1X environment and one where the enforcement point is not running 802.1X or where the enforcement point is closer to the core of the network.