

# Choosing the right flavor of 802.1X

10 Apr 2006 | SearchSecurity.com

By **Lisa Phifer**

802.1X provides an extensible framework for controlling WLAN usage. But 802.1X is merely an envelope that carries some type of Extensible Authentication Protocol ([EAP](#)). Which of the nearly 50 defined EAP Types would work best in your WLAN? In this tip, we compare the most popular EAP Types used with 802.1X, authentication methods that can be supported by each, known vulnerabilities and suitable usage environments.

## **EAP-MD5 (Message Digest #5)**

This EAP type provides one-way client authentication. The server sends the client a random challenge. The client proves its identity by hashing the challenge and its password with MD5. Because a man in the middle could see both the challenge and response, EAP-MD5 is vulnerable to dictionary attack when used over an open medium. Since there is no server authentication, it's also vulnerable to spoofing. Finally, EAP-MD5 cannot deliver keys. As a result, EAP-MD5 may be used over Ethernet but should never be used over a WLAN.

## **LEAP (Lightweight EAP)**

Also known as EAP-Cisco Wireless, this EAP type provides mutual client and server authentication over Cisco WLANs. As with EAP-MD5, a LEAP server sends the client a random challenge, which the client uses to return a hashed password. The authenticated client challenges the server for its password, followed by a key exchange. Because LEAP is a proprietary protocol, it can only be used in corporate WLANs with Cisco APs and Cisco-compatible cards. Like EAP-MD5, LEAP is vulnerable to dictionary attack. Today, many [attack tools](#) can be used to crack LEAP-authenticated passwords. New WLANs should therefore avoid LEAP. If your WLAN already uses LEAP, ensure that every client and server uses long random passwords, and upgrade to a stronger EAP type as soon as possible.

## **EAP-TLS (Transport Layer Security)**

This EAP type is generally regarded as the strongest available and the most expensive to deploy. It provides mutual certificate authentication between client and server, using the standard [TLS](#) protocol (a descendant of the [SSL](#) protocol used to secure most Web transactions.) The server uses TLS to demonstrate that it holds a digital certificate, requesting the same from the client. The client uses its certificate to prove its identity and keying material is exchanged. The TLS tunnel ends once authentication has been completed, but the keys delivered by EAP-TLS can be used to encrypt data with [AES](#), [TKIP](#) or [WEP](#). EAP-TLS is a good fit in WLANs where clients already have digital certificates or where high security needs justify investment in a [public key infrastructure](#) to manage those certificates.

## **EAP-TTLS (Tunneled TLS)**

This EAP type balances security vs. deployment cost by replacing client-side certificates with legacy password authentication methods like [PAP](#), [CHAP](#) and MSCHAPv2. EAP-TTLS requires the server to authenticate itself by certificate and establish a TLS tunnel through which to challenge the client. Even when a cleartext password is returned, the client's response is obscured by the TLS tunnel. To avoid exposing the client's name, EAP-TTLS should be configured to send an "anonymous" identity when 802.1X starts, then send the actual identity through the TLS tunnel. That tunnel ends when authentication is completed and keys are delivered. EAP-TTLS is a good fit for WLANs that wish to reuse legacy user authentication databases (e.g., LDAP, Active Directory) in a secure fashion.

## **PEAP (Protected EAP)**

PEAP is very similar to EAP-TTLS but uses different client authentication protocols. Like EAP-TTLS, PEAP provides mutual authentication, using server certificates, a TLS tunnel and client authentication through that encrypted tunnel. Unlike EAP-TTLS, PEAP requires the client to use another EAP type, like EAP-MSCHAPv2 or EAP-GTC (see below). Although the same user credentials can be used with EAP-TTLS, a PEAP authentication server must be able to parse both EAP and the contained legacy authentication protocols.

*Note: It is critical to use the same version of PEAP on clients and servers. PEAPv0/EAP-MSCHAPv2 requires 802.1X supplicant (client) software included in Windows XP SP2 and 2000 SP4. PEAPv1/EAP-GTC requires another 802.1X supplicant, like the one installed with Cisco's Aironet Client Utility. These supplicants are mutually exclusive -- installing a PEAPv1 client replaces any existing PEAPv0 client.*

## **EAP-MSCHAPv2 (Microsoft Challenge Handshake Protocol)**

This EAP type can be used inside the TLS tunnel created by Protected EAP. EAP-MSCHAPv2 wraps Microsoft's Challenge Handshake

Protocol inside the Extensible Authentication Protocol. It is a good fit for companies that want to reuse Microsoft user credentials and servers (e.g., NT Domain Controllers, Windows Active Directories) for wireless authentication. Similar goals can also be accomplished with EAP-TTLS/MSCHAPv2.

### **EAP-GTC (Generic Token Card)**

This EAP type can be used inside the TLS tunnel created by Protected EAP. EAP-GTC defines an EAP envelope to carry "one time passwords" generated by token cards like RSA SecurID. It is a good fit for companies that use two-factor authentication to avoid common password compromises (e.g., passwords shared with others, written on sticky notes, stored on stolen laptops) – especially where such tokens are already in use by a remote access VPN. Those starting a WLAN from scratch must decide whether token deployment cost is justified.

### **EAP-SIM (Subscriber Identity Module)**

This EAP type provides mutual authentication, based on the SIM card found in cellular telephones sold by GSM carriers. A SIM may be a small chip inserted into a dual-mode phone, a wireless data card or a USB stick. That smartcard implements the authentication algorithm normally used by cellular handsets to authenticate to GSM telephone networks. 802.1X requests carrying EAP-SIM are relayed through the carrier's roaming gateway to a GSM authentication server. This type can be used to authenticate devices like smartphones that roam between commercial 802.11 hotspots and GSM networks.

### **EAP-AKA (Authentication and Key Agreement)**

EAP-AKA is similar to EAP-SIM, but meets the needs of non-GSM carriers by using the User Service Identity Module (USIM) employed by Universal Mobile Telecommunications System (UMTS) networks. Although your carrier's network determines which type your smartphone must use, the permanent authentication keys used by EAP-AKA are considered stronger than the derived authentication keys used by EAP-SIM.

### **EAP-FAST (Flexible Authentication via Secure Tunneling)**

This EAP type was created by Cisco as a replacement for LEAP; it is available today in some Cisco APs and Cisco-compatible wireless cards. Like PEAP and EAP-TTLS, FAST provides tunneled mutual authentication. However, EAP-FAST does not require the server to authenticate itself with a digital certificate. Instead, a one-time provisioning exchange establishes a shared secret, called a Protected Access Credential (PAC) Key. That PAC Key is used for all subsequent authentications. EAP-FAST caters to small footprint clients, like VoWiFi handsets, that would be noticeably slowed by digital certificate signature verification. Currently, EAP-FAST is limited to use in Cisco-based WLANs.

### **Wi-Fi Alliance Certification**

The Wi-Fi Alliance currently tests the following EAP types: EAP-TLS, EAP-TTLS, PEAPv0, PEAPv1 and EAP-SIM. To determine which EAP types your products support, see the Wi-Fi Alliance's Certified Products page. Products that have not been certified may still interoperate, but it is wise to check for EAP type compatibility when deploying 802.1X in multi-vendor WLANs.