

Running head: PORT-BASED AUTHENTICATION

Port-based authentication with IEEE Standard 802.1x

William J. Meador

Port based authentication

Preface

You have probably already heard of IEEE 802.1x; it is one of the security buzzwords that we hear about often. Many people associate 802.1x with securing 802.11 wireless LAN's, but 802.1x has the ability to do more, including securing IEEE 802.3 Ethernet and even IEEE 802.5 Token Ring network ports. So what exactly is the IEEE 802.1x Standard? What does 802.1x attempt to accomplish? How does it work? And can it be considered secure?

Introduction

IEEE Standard 802.1x-2001 was approved by both the Institute of Electrical and Electronics Engineers (IEEE) and the American National Standards Institute (ANSI) in 2001. It provides port-based network access control for local and metropolitan area networks:

This standard defines a mechanism for Port-based network access control that makes use of the physical access characteristics of IEEE 802 LAN infrastructures in order to provide a means of authenticating and authorizing devices attached to a LAN port that has point-to-point connection characteristics, and of preventing access to that port in case in which the authentication and authorization process fails (IEEE, 2001, p. iii).

Why would a network security administrator want to implement port-based authentication? Securing network borders is no longer confined to placing a firewall between the Internet and local network. Network borders now include publicly-accessible Ethernet ports and Wireless networks. Simple segmentation or a VLAN for such ports can impede a valid user's networking capabilities. A method was thus needed to securely authenticate users on a per-port basis. Once authentication can be established, a network administrator can be very flexible in how he/she handles authenticated and unauthenticated users. One scenario may be at a

university. The library has publicly-accessible Ethernet ports, and allows Internet-only access for (unauthenticated) guests. But authenticated users will still have access to servers and other local resources. Another scenario is a company has an 802.11b Wireless LAN. It does not want any information leaked to the public, so unauthenticated users will not be granted any access. Port-based authentication gives the network security administrator flexibility in securing the internal network.

Definitions

Several definitions are essential in order to understand IEEE Std. 802.1x-2001. The supplicant is the client that wants to authenticate to the network. The authenticator is the Wireless Access Point (802.11) or Ethernet Switch (802.3) that facilitates authentication for the supplicant. The authentication server is what determines whether a supplicant is authenticated or not; it is generally a RADIUS server but can also be a Cisco SecureACS server. Port access entity (PAE) is the authentication protocol associated with the supplicant or authenticator (IEEE, 2001).

How it Works

IEEE 802.1x provides a framework for port-based authentication, but lacks its own algorithms for authentication. Instead it relies on some variation of Extensible Authentication Protocol (EAP) to deal with authentication algorithms. EAP was defined in RFC 2284 as an authentication protocol for PPP, but is often called EAPOL (Extensible Authentication Protocol over LAN) in context to IEEE 802.1x. Many people have criticized EAP, because it was put to standard in 1998 for a protocol that is not widely used anymore. EAP has developed into several variations that do, however, include encryption.

EAP-MD5 is the simplest form of EAP. It uses a MD5 password hash for supplicant authentication but does not provide for server authentication. EAP-MD5 is not suitable for most networks, because its packets are not encrypted and can be sniffed. Also, since the authentication server cannot itself be authenticated to the client, rogue network appliances can masquerade as an authentication server and easily pickup passwords when clients authenticate to them. LEAP stands for Lightweight EAP and was developed by Cisco. It is not typically used anymore, because it uses password hashes for server and supplicant authentication, which can be picked up by a packet sniffer such as Ethereal. Its server authentication does make it a better alternative to EAP-MD5. EAP-TLS was created by Microsoft and uses Transport Layer Security for encryption. Server and supplicant both use public-key authentication. Transport Layer Security is a replacement for Secure Socket Layer. PEAP and EAP-TTLS are both closely related, as they both used public key server authentication, and offer flexible options for supplicant authentication. PEAP stands for “Protected EAP” and was developed by Microsoft, Cisco, and RSA Security. EAP-TTLS stands for “EAP-Tunneled Transport Layer Security” and was developed by Funk Software and Certicom (HackFAQ, 2004).

IEEE 802.1x's and EAPOL's functionality is much like of PPP. The supplicant physically connects to an Ethernet port for 802.3 (or attempts to connect to the access point in 802.11). The authenticator detects activity and then sends an “EAP-Request/Identity” packet to the supplicant. The supplicant sends an “EAP-Request/Identity” packet to the authenticator, which is then forwarded to the authentication server. The authentication server sends a challenge to the authenticator, which in turn forwards it to the supplicant. The supplicant responds to the challenge via the authenticator, which is forwarded to the authentication server. If the challenge response is correct, the authentication server responds to the supplicant with a success message.

The supplicant is granted full access to the LAN via the authenticator. If the challenge response is incorrect, access may not be granted, or the supplicant may be placed in a special VLAN.

EAPOL frames reside on the data link layer of the OSI model and cannot be routed. This helps in limiting network access to a supplicant before it has authenticated. So EAPOL frames are only sent from the supplicant and authenticator, and the authenticator and authentication server use some type of Layer-3 protocol such as RADIUS to encapsulate EAP messages.

EAPOL frames contain five fields: Ethernet type, protocol version, packet type, length, and body. In IEEE 802.3 and IEEE 802.11, the Ethernet type is 2 bytes long, and in IEEE 802.5 the Ethernet Type is 8 bytes long. The other fields are identical between 802.5 and 802.11 and 802.3. The protocol version field is 1 byte in length and contains an unsigned binary number indicating EAPOL version. The packet type field is also 1 byte in length and contains an unsigned binary number. The packet type can be EAP-Packet, EAPOL-Start, EAPOL-Logoff, or EAPOL-Key. The value of the packet type field is an EAP-Packet when both the supplicant and authenticator are authenticating. The EAP-Packet contains MD5-Challenge or TLS information. A packet type value of EAPOL-Start indicates the supplicant wants to start the authentication process. An EAPOL-Logoff value indicates the supplicant is ready to end the 802.1x session. A packet type value of EAPOL-Key is sent from the authenticator to the supplicant once TLS negotiation has take place between the supplicant and authentication server. The packet body length field is 2 bytes in length and indicates the length of the Packet body. The length of the packet body field is thus variable (IEEE, 2001).

Applying port-based security

Putting IEEE 802.1x into practice can be challenging. Windows XP has native support for 802.1x, but earlier Windows clients lack support for 802.1x. Fortunately Microsoft provides

a 802.1x patch for Windows 2000, but earlier versions of Windows must use a third party client. Linux, UNIX, and Mac OS also lack native support for IEEE 802.1x. Open1x, an open source IEEE 802.1x client is currently available and is actively being developed. Open1x is still very raw and is command-line only. It also lacks good documentation. A more popular solution for Linux, UNIX, and Mac is a client from Meetinghouse. The Meetinghouse client is GUI-based and has sufficient documentation but costs \$20 per client. Setting up an authentication server can also be tricky. If using a public-key authentication system, one must determine a means to distribute the key to clients. Static key distribution can also pose a threat. If someone in the organization has a laptop and decides or is forced to leave, he/she will still have a means to enter the network. For network devices that do not have 802.1x clients, special exceptions must be made to allow network access. Printers, scanners, and IP phones are just a few devices that do not have 802.1x support, plus operating systems such as Windows 98. Authenticators can be configured to allow certain MAC addresses to connect to certain ports, but this can be an administrative nightmare in large organizations.

Conclusion

Is IEEE 802.1x + EAPOL secure? Much of the answer to this question relies on which EAP variation is used and how it is setup. EAP-MD5 and LEAP both contain unencrypted authentication and should be avoided if at all possible. EAP-TLS, PEAP, and EAP-TTLS all contain encrypted mutual authentication, but this can be overridden. Even mutually authenticated PEAP or EAP-TLS can be susceptible by man-in-middle and session hijacking. One proposed solution is a per-packet authenticity and integrity of IEEE 802.11 (and 802.3) frames. I believe this would be too high of a cost. Another solution they proposed was to replace EAP-Success messages with an EAPOL-key to reduce Man in the Middle attacks. This

seems like a much more feasible solution (Mishra and Arbaugh, 2002). Another threat for 802.1x-based networks is DOS attacks. An attacker may be able to spoof EAPOL logoff frames, logging a valid user off from the authenticator. The attacker may then flood the authenticator with EAPOL start frames, rendering the authenticator unavailable (Schwartau, 2002).

IEEE Standard 802.1x-2001, along with an encrypted version on EAP, can go a long way in securing 802.11, 802.5, and 802.3 networks at a port-based level. It is not the perfect solution, but no network can ever have absolute security. IEEE 802.1x adds another layer to the “strength in layers” concept. As IEEE 802.1x gains popularity, it is inevitable that solutions will be developed to address the standard’s weaknesses.

References

- HackFAQ (2004). *What are EAP, LEAP, PEAP and EAP-TLS and EAP-TTLS?* Retrieved April 25, 2004, from <http://www.hackfaq.org/wireless-networks/eap-leap-peap-eap-tls-ttls.shtml>
- Institute of Electrical and Electronic Engineers (IEEE) LAN/MAN Standards Committee. (June 14, 2001). *IEEE Std 802.1x – IEEE Standard for Local and metropolitan area networks-Port-Based Network Access Control*. New York: Institute of Electrical and Electronics Engineers, Inc.
- Mishra, A., Arbaugh, W. (February 6, 2002). *An Initial Security Analysis of the IEEE 802.1X Standard*. Retrieved April 26, 2004, from <http://www.cs.umd.edu/~waa/1x.pdf>
- Schwartau. (September 2002). *War Driving Lessons*. Retrieved April 27, 2004, from <http://www.nwfusion.com/columnists/2002/0902schwartau.html>