Authentication Overview

William H. Wolfe II **(14.5 mins)**

Strong Authentication and Encryption.

In 802.11, in order to set up a secure 802.11 link, we first have to go through an authentication process. This authentication process actually authenticates a user/client to the SSID. One thing we want to ensure is that we have a strong mutual authentication, and the recommendation is actually to use a technique called 802.1x-Extensible Authentication Protocol. We're going to spend some additional time in a subsequent module on 'Central Authentication' which ties in to 802.1x EAP.

Following authentication, we have association which is where we establish the virtual port link between the client and the access point that transmits the RF data. That uses an encryption algorithm to make sure that all of the RF propagated packets and frames are in fact secure and confidential.

We'll cover some of the encryption algorithms and their standards and weaknesses as we go through this module.

Should we have a secure SSID or an open SSID? Needless to say, a secure SSID is much preferred, but in some situations, you want to have an open SSID for something like your guest network or possibly a WiFi hotspot. Secure SSIDs, once they are configured cannot fall back to an open authentication. If a user does not support 802.1x, it cannot fall back to any type of open authentication. Pre-shared keys that are derived from an 802.1x environment cannot coexist with secure SSID. However, we can combine both techniques- Secure SSID and open SSID as long as we have multiple SSIDs. Multiple SSIDs are generally configured in cooperation with wireless VLANs and unique subnets that correspond directly to those multiple SSIDs.

During the authentication evolution from the beginning, the first was a MAC Address Authentication process. That moved its way into Wired Equivalent Privacy which we're going to touch on a little bit in a moment. Then we moved into the 802.1x with a dynamic WEP. This gives us a little bit of additional security because we ended up using the 802.1x method, which allow us to authenticate a user against a RADIUS server. Finally, we ended up with WiFi Protected Access (WPA) and WiFi Protected Access Version Two (WPA2), which by the way is the current recommended minimum standard.

WPA2 was also ratified as an IEEE standard - the 802.11i standard. The WiFi Protected Access nomenclature is actually something that was developed by the WiFi Alliance, which is the interoperability body.

WPA2, and we'll talk about this momentarily, uses the Advanced Encryption Standard encryption algorithms. WPA originally used Temporal Key Integrity Protocol encryption which we'll see, had some vulnerabilities. The authentication mechanisms can either be a pre-shared key, or as some people will call it, the personal mode. That's generally for home and small-to-medium business use.

The 802.1x with Extensible Authentication Protocol is for enterprise, or large office, or corporate use. You'll generally see these WPA2 options in configuration of wireless technology, whether it be on a home wireless router, or on a standalone access point, or in a controller-based access point environment, or even a cloud-based wireless environment.

You'll see the pre-shared key being called the personal mode and the 802.1x EAP mode called the enterprise mode.

Let's talk a little bit about open authentication and what generally happens when we authenticate to an SSID, keeping in mind that this has absolutely no authentication credentials, and there is also no encryption algorithm used to encrypt the data being propagated via RF. This is typical for a WiFi hotspot.

Because clients today use active scanning versus passive scanning, we end up having a client to send a probe request, the access point then sends a probe response and the client evaluates the access points responses and select the access point that has the strongest signal and meets all of the requirements for connectivity. At that time, the client then sends an authentication request to authenticate itself to the SSID. In this case, because it's open authentication, the access point automatically confirms that authentication and then registers the client in the client database with inside the access point. The client then will associate itself with that specific access point that it had already determined was the best one from the initial probe response, and the association continues and fully registers the client as a connected device to that access point and traffic begins to flow

With pre-shared key WEP (Wired Equivalent Privacy) which as everyone knows, has a tremendous amount of vulnerabilities and for all practical purposes, has been deprecated from use in the industry. There's a couple of changes that happen when we go to WEP. In step 4, the access point sends the authentication response back to the client containing an unencrypted challenge text. The client then encrypts that challenge text with the WEP key that has been configured and sends it back to the access point. The access point then compares that encrypted challenge text with its copy of that encrypted challenge text. If they're the same, the client is then properly authenticated.

Needless to say, part of the problem and the issues associated with WEP is that, these initialisation vectors are randomly generated. They're used to start the encryption process of the data and there's a limited bit length for the keys, and that generates some limitations. There's no per packet authentication, or message integrity check. Hackers can easily obtain that initial challenge phrase because it was not encrypted, so they can crack the WEP key and get easy access to that particular SSID.

As mentioned earlier, extremely weak encryption of the data and those initialization factors are repeated. As a result, it is actually very insecure as a authentication method, as well as weak encryption, and generally considered less secure than even using open authentication and applying a user authentication and dynamic encryption method to it such as incorporating 802.1x with dynamic WEP.

WiFi Protected Access or WPA, this is version 1, was introduced in 2003. Literally WPA incorporated 802.1x with EAP authentication for enterprise and PSK for personal edition. It was considered more secure, but it did still use an initialisation vector and it use the Temporal Key Integrity Protocol, which was considered a less secure encryption method.

These are the comparisons between the enterprise and the personal mode. Generally, the major difference between enterprise and personal is that with enterprise you're going to have a RADIUS server used within the environment, and its going to have centralised access control. We'll be covering a little bit more in a follow-up session.

The encryption methods are TKIP or AES. TKIP was the standard, and generally what you will see applied to the WPA authentication.

In June of 2004, so roughly a year later, the IEEE standardised 802.11i, which again, is the equivalent to the WiFi Alliance's WPA2. It change the encryption method from TKIP to AES, which is considered the strongest encryption algorithm, and was actually ratified and standardized by NIST and the FIPs 197 document. It is the government's recommended standard encryption algorithm. We'll speak more about the encryption algorithms in a little while.

The AES encryption is basically a CBC-Mac or counter mode authentication method. It's generally known as AES-CCMP and it actually uses 128-bit key. AES is also implemented in hardware and the NIST, FIPS team actually goes through an annual review of vendors to confirm that their products are adhering to and are using the proper AES standardised algorithms. We have full interoperability between vendors.

Some of the comparisons between WPA, WPA2 and 802.11i. As you can see, there's basically in the IEEE standard, no pre-shared keys, so officially it does not support personal mode it is strictly enterprise mode only. Lastly, there is an option for doing some level of security by using matched filtering, but generally not consider a primary method or even an additional method for wireless because of the fact that MAC addresses have to be maintained at a local database, and they can always be spoofed.