Wireless Networking

Overview

Hard to believe there was once a time when a laptop or PC had to be connected to an outlet to access the Internet, isn't it? Wireless is becoming a larger and larger part of everyday life, to the point where people expect to be able to access the Net or connect to their network while eating lunch.

Wireless networks are created by configuring Wireless Access Points (WAP or AP, depending on documentation). If you're connecting to the Internet or your company's network from a hotel or restaurant, you're connected to a *lily pad network.*

Unlike the networks we've discussed previously, the WAPs in a lily pad network can be owned by different companies. The WAPs create hotspots where Internet access is available to anyone with a wireless host - and hopefully, a username and password is required as well!

WAPs are not required to create a wireless network. In an *ad hoc WLAN* ("wireless LAN"), the wireless devices communicate with no WAP involved.

A much more common wireless configuration is an *infrastructure WLAN*, where a WAP is used to allow multiple devices to connect to the wired network.

WAPs make it relatively simple to extend the availability of your network as the company grows. As you add personnel and expand your office, just add WAPs to connect these users to the network!

CSMA/CD vs. CSMA/CA

Ethernet has CSMA/CD, and wireless networking has CSMA/CA, Carrier Sense Multiple Access with Collision *Avoidance.* CSMA/CA works much the same as CSMA/CD...

- a host that wants to transmit must listen first to see if another host is transmitting
- if the channel is idle, the host can transmit
- if the channel is busy, the host can't transmit, and must invoke a random backoff timer

So what's the real difference between CSMA/CD and CSMA/CA? CA is used on wireless networks, and jam signals will not be sent over a wireless network. Collisions are not *detected* on a wireless network, they can only be *avoided,* so we use CSMA/CA instead of CD.

The Distributed Coordination Function (DCF) vs. WiFi Multimedia (WMM)

The IEEE 802.11 WLAN standard (*802.11e*) uses the DCF to implement the DIFS interval mentioned above. With normal data transfer, the DIFS interval doesn't cause much trouble - except with our delay-sensitive traffic, voice and video!

That's where *WiFi Multimedia* comes in! WMM is actually QoS for our WLAN traffic, since priority is given to delay-sensitive traffic while making regular data wait its turn. The WMM standard makes a point of mentioning that absolute QoS is not guaranteed, but it's a major step forward over DCF.

Since WMM uses an enhanced version of DCF, that version is called - you guessed it! - *Enhanced DCF* (EDCF). WMM and EDCF allow for true QoS over a wireless connection, although it is not guaranteed.

WMM has four preset priority levels:

- Platinum (for voice)
- Gold (for video)
- Silver (for everything else - best-effort)
- Bronze (background traffic)

Just as PQ uses the third queue down (the Normal queue) for traffic that has not been specifically been assigned to another queue, WMM uses the Silver queue for its default queue.

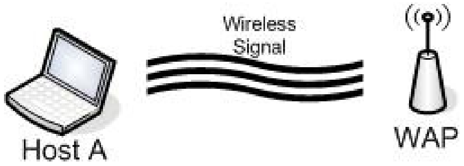
When it comes to wireless standards, we're in a hurry. 802.11e was actually in the process of being ratified when the Wi-Fi Alliance (http://www.wi-fi.org/) released WMM.

While WMM has four priority levels, 802.11e has *eight*. Here's how these levels map to each other:

- Platinum (Voice) - 802.11e Priority Level of 6 or 7
- Gold (Video) - 802.11e Priority Level of 4 or 5
- Silver (Best-Effort) - 802.11e Priority Level of 0 or 3
- Background (Bronze) - 802.11e Priority Level of 1 or 2

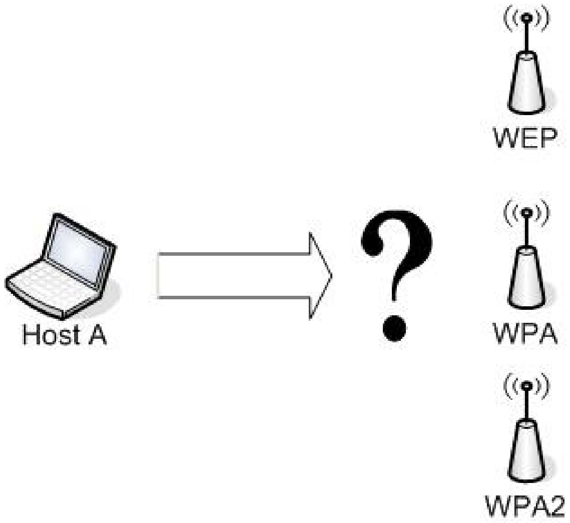
Lightweight Access Points

Many WLANs start small and end up, well, not so small! At first, centralizing your security policies doesn't seem like such a big deal, especially when you've only got one access point.

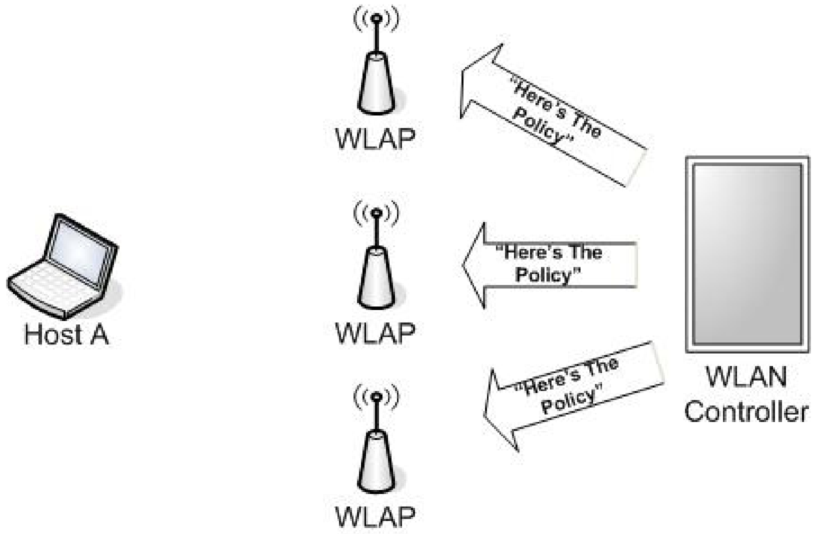


As your network grows larger and more access points are added, having a central policy does become more important. The more WAPs you have, the bigger the chance of security policies differing between them - and the bigger the chance of a security breach.

Let's say you add two WAPs to the WLAN network shown above. Maybe they're configured months apart; maybe they're configured by different people - but the result can be a radically different set of security standards.



We've now got three very different WLAN security protocols in place, and the difference between the three is huge, as you'll soon see. Depending on which WAP the laptop uses to authenticate to the WLAN, we could have a very secure connection - or a very insecure connection. This simple example shows us the importance of a standard security policy, and that's made possible through the use of *Wireless Lightweight Access Points* (WLAPs) and *Wireless LAN Controllers* (WLCs).



Configuring the access points as WLAPs allows us to configure a central device, the WLAN Controller, to tell the WLAPs what the security policy is. The protocol used to do so, the aptly-named *Wireless LAN Access Point Protocol* (WLAPP), detects rogue (fake) access points as well.

While having a centralized QoS policy isn't as important as having a single security policy, it doesn't hurt, either! The WLAN controller is capable of handling QoS tasks as well as security tasks, and the primary task is mapping one QoS value to another. With an end-to-end communication, we will have three different values in play:

- Layer 2 - 802.1p
- Layer 3 - DSCP
- WMM priority values

It's the WLAN controller that will handle mapping one value to another when necessary.

As always, there's a tradeoff with any benefit! Having centralized WLAN controllers does help to standardize security and QoS policies, but we can't have the controller handle all AP operations. The *split MAC architecture* does just what it sounds like - it splits the MAC layer processing between the WLAN controller and the APs.

"War Driving"

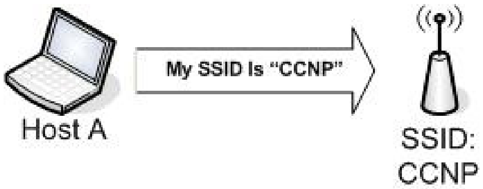
Many of us (ahem) have done this without even knowing it had such a dramatic-sounding name! The term "war driving" refers to the process of driving around a neighborhood or business district in hopes of finding a non-secured WLAN. ("war driving" is derived from "war dialing", a term from the film *WarGames*).

There's one very sad fact about many of today's WLANs: The WLAN devices have basic security features that are easy to configure - and many users just don't take the time to configure them.

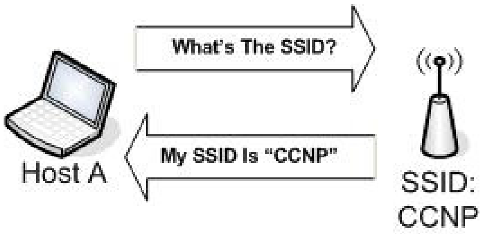
Regardless of what it's called, as WLAN security admins, we need to stop it! Here are a few methods of doing so.

Service Set Identifier (SSID)

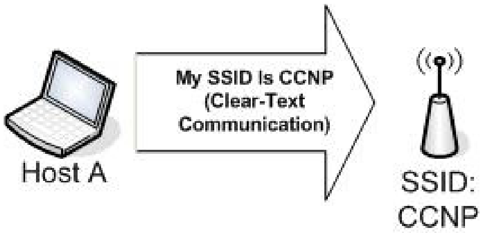
When you configure a name for your WLAN, you've just configured a SSID. The SSID theory is simple enough - if the wireless client's SSID matches that of the access point, communication can proceed. The SSID is case-sensitive and it has a maximum length of 32 characters.



A laptop can be configured with a null SSID, resulting in the client basically asking the AP for its SSID; if the AP is configured to broadcast its SSID, it will answer and communication can proceed.



A classic "gotcha" with SSIDs is to configured the AP to not broadcast its SSID. This would seem to be a great move for your WLAN's security ... but is it?



As you've already guessed, this is not an effective security measure, because the SSID sent by the client is not encrypted. It's quite easy to steal, and obviously no unencryption is needed!

MAC Address Authentication

During your CCNA studies, you learned about a Cisco switch feature called port-based authentication. This authentication scheme allowed a device to successfully authenticate only if its MAC address was considered secure for that particular port. There are WLANs set up to use MAC addresses in a similar fashion.

Basically, the AP keeps a list of secure MAC addresses; devices with a secure MAC address can authenticate successfully, while those with a non-secure MAC cannot.

If this strikes you as fine for a switchport but not fine for a WLAN, well, I agree with you! It's pretty easy to spoof a MAC address, especially when there is no physical connection between the client and the access point.

WEP, WPA, And WPA2

These three WLAN security standards are the result of two evolutions:

- WEP came first
- WPA evolved from WEP
- WPA2 evolved from WPA

There are significant differences between the three, so let's take a look at each and compare them.

Wired Equivalent Privacy (WEP) has some real problems:

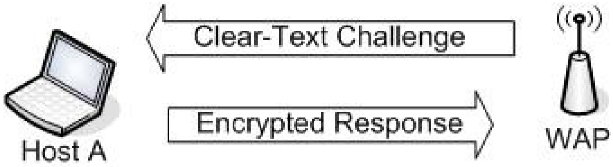
- Clear-text keys
- Static keys (makes passwords easier to guess)
- One-way authentication (client does not authenticate AP, making it easier for rogue access points to infiltrate the WLAN)
- Encryption scheme is very easily broken in a matter of seconds

Other than that, it's pretty good. :)

WEP supports two forms of authentication, *open* and *shared key*. Open authentication is pretty much what it sounds like - the virtual door to an AP is wide open. Any device can authenticate and then open communication with the AP.

According to Cisco's website, if both devices are using WEP but the key on the client does not match that of the AP, authentication will succeed but data cannot be successfully passed.

With shared key, the AP will send a clear-text challenge; the PC will encrypt its answer.



I hope the phrase "clear-text challenge" sets off alarm bells! This clear-text transmission results in shared key authentication actually being considered *less secure* than open authentication.

WEP was ratified by the IEEE in 1999, and things have changed just a bit since then. WEP can be broken by easily-obtained software programs in a matter of seconds, so it's a good idea to avoid WEP unless it's the only option you have. And if it is the only option you have, buy something that *does* give you more options!

WPA

The next step in WLAN security was *Wi-Fi Protected Access* (WPA). WPA works with all wireless NICs, but you may have trouble running it on legacy (old) APs. If you can't run WPA on your APs, it's time to get some new APs.

WPA's strengths:

- Two-way authentication - AP authenticates the client, client authenticates the AP
- Dynamic keys and a stronger encryption scheme through use of *Temporal Key Integrity Protocol* (TKIP, "tee-kip")
- WPA uses an 8-byte *Message Integrity Check* (MIC), sometimes called "Michael", to protect against replay attacks, spoofing, and man-in-the-middle attacks.
- WPA uses 802.1x or pre-shared keys (PSK) for authentication

NOTE: Some Cisco documentation puts a "C" in front of TKIP and MIC - "CTKIP" and "CMIC". Watch for that on exam day and in future studies. They're generally referred to as TKIP and MIC and that's how they'll be referred to in the rest of this section.

Some additional details regarding TKIP:

- WEP-only devices can be used to run "WEP with TKIP", and WEP devices running TKIP have backwards compatibility with WEP-only devices.
- Both WEP and TKIP use the RC4 stream cipher for encryption, but TKIP protects changes key values where WEP does not.

WPA requires the use of a *passphrase* rather than a password. The recommended length of a passphrase is 20 - 30 characters, which will immediately have some users running WEP simply because WEP allows a short password to be configured.

There are other potential issues with WPA:

- There's always the legacy issue to consider when it comes to backwards compatibility, but at this point, you should strongly consider replacing WLAN equipment that does not support WPA or a later, stronger solution.
- Choosing the correct EAP flavor can be a challenge - more about that later.
- There's a potential issue with "Michael" (MIC). Access points that run WPA will shut down their Basic Service Set if it receives two packets, one right after the other, that has a bad MIC. A DoS attack specifically designed to counteract Michael can take advantage of this situation.
- Another potential issue lies with the use of pre-shared keys (PSK). If a small passphrase is allowed and then intercepted, a dictionary attack can be run by an attacker, resulting in a compromised passphrase.

Source: http://www.sonicwall.com/downloads/WiFiSec_vs_WPA.pdf

WPA2

Here's what happened next:

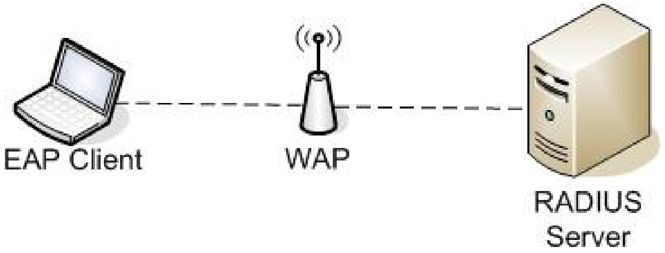
- After WPA was ratified by the Wi-Fi Alliance, the IEEE came out with 802.11i.
- After the IEEE came out with .11i, the Wi-Fi Alliance came out with WPA2.

The good news: .11i and WPA2 are fully compatible and interoperable.

WPA2 is considered fully secure through its use of the *Counter Mode with Cipher Block Chaining Message Authentication Code Protocol,* thankfully referred to as CCMP. This "fully secure" status is earned through the use of the Advanced Encryption Standard (AES) algorithm to perform encryption. As opposed to a stream cipher, AES is a block cipher.

Both WPA and WPA2 can use pre-shared keys (PSK) or 802.1x authentication to authenticate users. The use of 802.1x requires another device to get involved - a RADIUS server. We're going to use one of several versions of the Extensible Authentication Protocol (EAP) to handle communications during the authentication process.

Both the client and AP must support EAP for this to work!



We actually have four different EAP flavors to choose from:

- Cisco LEAP
- EAP-FAST
- Protected EAP (PEAP)
- EAP-TLS

During out discussion of these, be sure to note the different authentication approaches taken by each. Getting them mixed up in the exam room or the real world can lead to an undesirable result.

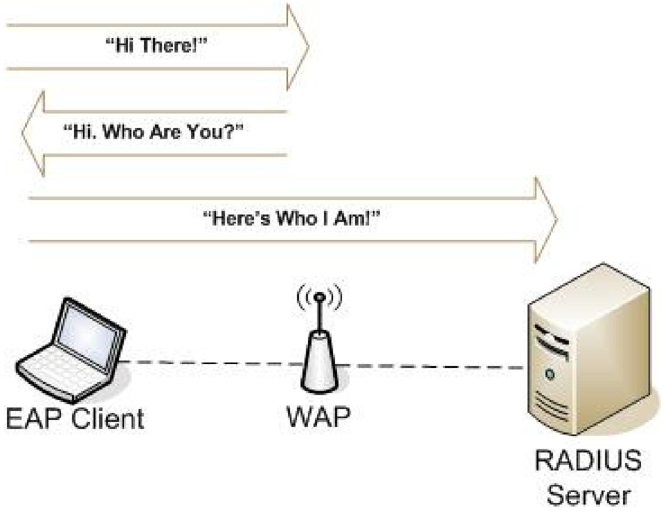
Cisco LEAP

It will not surprise you to learn that Cisco's Lightweight Extensible Authentication Protocol (LEAP) is installed by default on every Cisco wireless product. Since we're preparing for a Cisco exam, let's pay a little extra attention to LEAP.

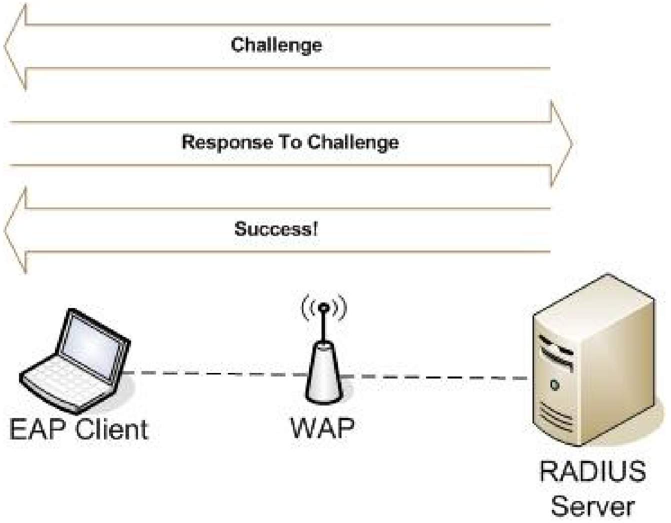
Some LEAP basics:

- LEAP is Cisco-proprietary, but third-party vendors can support it via the Cisco Compatible Extensions Program
- The RADIUS server will authenticate the client, and then the client will authenticate the RADIUS server, resulting in *strong two-way authentication*

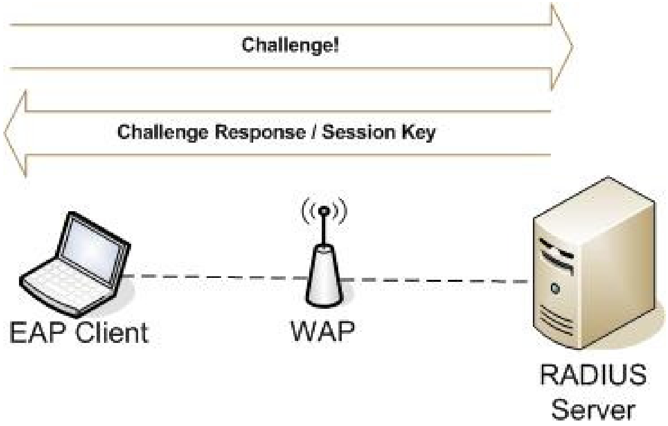
Here's an overview of the LEAP authentication process. The client can either initiate the process by telling the AP it wants to connect or the client can answer an AP "Request/Identity" message, but in either case the client will tell the AP it wants to connect. The AP responds by saying "Who the heck are you?"; when the client responds, that response goes through to the RADIUS server.



The RADIUS server will challenge the client. If all goes well, the client sends a response, and the RADIUS server sends a Success message.



We're not quite done! Remember, LEAP is a *mutual* authentication process - and that means that the client will now challenge the RADIUS server! The RADIUS server answers the challenge and sends the session key to the client, and the mutual authentication process is now complete.



If this process reminds you of CHAP, it should. LEAP is actually based on MS-CHAP v2, and that's not exactly the strongest authentication procedure around. It's pretty easy to find a program online that can crack LEAP, so it's a good thing we've got some stronger options!

Even with that potential issue, LEAP is considered to be stronger than WEP. (Yeah, I know, big deal!) LEAP is not an secure as WPA or WPA2, though.

Extensible Authentication Protocol-Flexible Authentication via Secure Tunneling (EAP-FAST)

The acronym suggests additional speed in the authentication process, but that's not the beauty of EAP-FAST. The key here (no pun intended) is the "secure tunnel" part. Where LEAP is really wide open to attack and compromise, FAST builds a tunnel through which the mutual authentication will take place.

Defined by RFC 4851, EAP-FAST gives us what some would consider the best of both worlds - a secure tunnel for authentication without the "bother" of using secure certificates. You do have the option of configuring certificates with EAP-FAST, though.

EAP-FAST is a three-phase process, but watch the numbering -- the first phase is officially named "Phase Zero". In Phase Zero, we need to get a Protected Access Credential (PAC) on the client. Technically this is an optional phase, since the PAC can be manually configured on the client, but generally the PAC will be dynamically assigned ("provisioned", in EAP-speak).

In Phase One, an encrypted tunnel is created; in Phase Two, credentials are exchanged and mutual authentication is performed.

The eagle-eyed among you might have spotted a potential security issue in that process! If the PAC is dynamically assigned, it could be intercepted en route to the client. If you're concerned about this, you can do one of two things:

- Configure the PAC manually on the client
- Introduce secure certificates to the EAP-FAST process

Protected EAP (PEAP)

Cisco goes to the dark side for PEAP - they joined forces with Microsoft! Well, MS and RSA Security, that is. PEAP is a strong, open-standard security scheme.

Actually, PEAP comes in two different versions:

- PEAPv0/EAP-MSCHAPv2
- PEAPv1/EAP-GTC (Generic Token Card)

With either flavor, there is a secure digital certificate involved. The clients will not have a certificate, but the authentication server will.

Extensible Authentication Protocol-Transport Layer Security (EAP-TLS)

Defined in RFC 2716, EAP-TLS is an open-standard protocol. It's also a bit controversial because it requires the clients to have a secure certificate as well as the authentication server.

"Hot Spots And Gotchas"

We've got more than our fair share of new acronyms in this sections, so let's review them...

TKIP ("tee-kip") is the *Temporal Key Integrity Protocol,* which uses the RCA4 stream cipher, which in turn allows hardware created to use WEP to instead use WPA, a much stronger authentication protocol.

Where WPA uses TKIP, WPA2 uses the even-stronger AES encryption

A comparison of LEAP authentication processes:

> LEAP - Mutual authentication with username and password, similar to CHAP. Stronger than WEP, which isn't much of a compliment, but not as strong as WPA or WPA2.
>
> EAP-FAST - Mutual authentication through a secure tunnel, resulting in better protection for the process than LEAP
>
> EAP-PEAP - Digital certificate configured on the server only

EAP-TLS - Both client and server have a digital certificate.

Potential issues with WEP:

Clear-text keys

Static keys (makes passwords easier to guess)

One-way authentication (client does not authenticate AP, making it easier for rogue access points to infiltrate the WLAN)

Encryption scheme is very easily broken in a matter of seconds

Other WEP notes:

WEP offers both open authentication and shared-key authentication.

Of WEP, WPA, and WPA2, WEP is by far the easiest to compromise.

You might instinctively think it's the other way around, but Cisco's website states that shared key authentication is less secure than open authentication.

Cisco's website also states that if WEP keys do not match during the open authentication process, the client can still authenticate and associate, but cannot forward data.

Potential issues with WPA:

There's always the legacy issue to consider when it comes to backwards compatibility, but at this point, you should strongly consider replacing WLAN equipment that does not support WPA or a later, stronger solution.

While stronger than WEP, both WPA and WEP can use TKIP, which is susceptible to attacks. (WPA2 uses AES.)

There's a potential issue with "Michael" (MIC). Access points that run WPA will shut down their Basic Service Set if it receives two packets, one right after the other, that has a bad MIC. A DoS attack specifically designed to counteract Michael can take advantage of this situation.

Another potential issue lies with the use of pre-shared keys (PSK). If a small passphrase is allowed and then intercepted, a dictionary attack can be run by an attacker, resulting in a compromised passphrase.

Other WPA And WPA2 Notes

WPA2 offers 802.1x ("dot1x") authentication and uses AES, a stronger encryption scheme than either WEP or WPA offers.

This 'n' That

Generally a SSID will be hard-coded on the wireless client, but the client *can* obtain the SSID from the AP.

Know your WMM - 802.1e priority level group "mappings":

While WMM has four priority levels, 802.11e has *eight*. Here's how these levels map to each other:

- Platinum (Voice) - 802.11e Priority Level of 6 or 7
- Gold (Video) - 802.11e Priority Level of 4 or 5
- Silver (Best-Effort) - 802.11e Priority Level of 0 or 3
- Background (Bronze) - 802.11e Priority Level of 1 or 2

MIC helps to prevent man-in-the-middle attacks as well as replay attacks, but does have a vulnerability that can be exploited by a DoS attack. An access point running WPA will shut down their Basic Service Set if it receives two packets, one right after the other, that has a bad MIC.