

Wi-Fi Security and the Implications Related to It (December 2009)

Matthew Seigel, *Student, Missouri State University*

Abstract— In this paper I will first cover who develops and tests Wi-Fi standards. Then I will be discussing the Wireless Security. I will cover WEP-RC4, WPA-TKIP, and WPA2-AES. I will show weaknesses and strengths of each encryption method. I will also be covering authentication methods for 802.11 protocols, including PSK and RSNA. I will relate the information I discovered while scanning my local area for Wi-Fi Access Points as well. Last I will discuss my possible solution to any problems stemming from insecure wireless.

Index Terms—Wireless LAN, Security, Authentication, Communication system security

I. INTRODUCTION

Wireless communication has spread throughout the industrialized world. Everything from cellular telephones to Wireless Fidelity has become common place in industrialized nations. Wireless Fidelity, or Wi-Fi, in particular was a revolutionary technology for the computer sector when it was first introduced in 1999. The originating term, Wireless Fidelity, has now become taboo to use when referring to Wi-Fi even though the Wi-Fi Alliance called it that in several white papers when it was first getting started. There have been other wireless communication schemes for computers, but the awesome part about Wi-Fi is that the IEEE, the people who control the Wi-Fi protocol development, have managed to get a universal standard setup throughout the world. Wi-Fi constitutes a new reign of freedom for the chronic and casual computer users. No longer are we tethered to our traditional cables, desk, and chair. We have the ability to surf and enjoy the internet from new locations. Wireless Local Area Networks, or WLANs, allow us to separate ourselves from the spaces typically used for computers and let us enjoy access to a network and the internet from previously unforeseen locations. However, as with any type of communication technology, there is always concern about the security of a new generation of communication technology. The purpose of this paper is to explore the security methodologies that are deployed with these technologies and their effectiveness. Unfortunately, Wi-Fi security has even affected large companies like TJ Maxx that have been penetrated because of their lack of security. I will be performing a proof-of-risk experiment in my own lab to

determine the difficulty of breaking into differing Wi-Fi security types and will conduct a survey of the Springfield, Missouri area to determine which technologies are currently deployed in the wild.

II. WI-FI DEVELOPMENT AND STANDARDS

The Wi-Fi development, maintenance, and standard creation are actually done by the IEEE. The IEEE is the world's leading professional association for the advancement of technology, according to their website (<http://www.ieee.org/portal/site>). The IEEE actually refers to Wi-Fi by its technical name: IEEE 802.11. IEEE 802.11 covers all versions of Wi-Fi technology. There are different flavors of 802.11 including B, G, and N. The previously listed ones are the most common versions in use today. The IEEE also specifies the types of security available for Wi-Fi communication.

There are currently only three main divisions of encryption technologies available to Wi-Fi communication: WEP, WPA, and WPA2. In a home setting when encryption is used most people will use a PSK, or Pre-Shared Key, for authentication. The PSK is a predefined password that grants user's access to your WLAN. The PSK authenticates you to the network and then the encryption is what protects your communication. In the corporate world we have more complex authentication systems that allow for different passwords for different users. This type of authentication requires a server to do the authentication and then after the authentication has been confirmed, the access point will setup the encryption. I will go into more depth later.

III. WI-FI ALLIANCE

The Wi-Fi Alliance is also a critical part in the deployment this technology. Although the Wi-Fi Alliance doesn't develop the standards they do verify the equipment using them. The Wi-Fi Alliance is responsible for certifying all the routers, access points, and other devices that use the IEEE 802.11 standard. The Wi-Fi alliance uses the standards developed by the IEEE to decide if the device certifiable. They have symbols that adorn the packaging of devices to specify their compatibility. Below are some images of them with labels depicting what they are a certification for. These help consumer tell if they are getting the correct kind of equipment for their network.

IEEE 802.11 N Certified



IEEE 802.11 Draft N Certified



Draft N is not the official release of the Wireless N standard. It was used before the standard was completed.

IEEE 802.11 B and G Certified



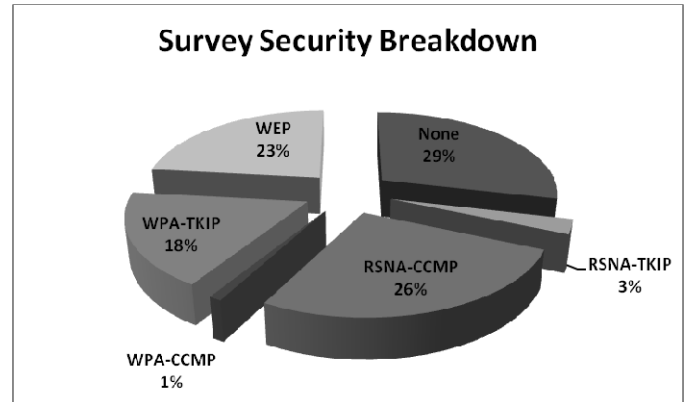
IV. SURVEY INTRODUCTION

I will be intermingling my results obtained while doing my survey with information later on in this presentation. Since I am doing that it is vital that I first explain how I did my survey and state what results I acquired. To perform the survey I brought my laptop with me while going about daily tasks and used a program called inSSIDer to log the Wi-Fi Access Points I encountered. I then wrote a program to parse the logs and count the number of each type of wireless security I encountered. To ensure the uniqueness of each Access Point I kept track of the MAC address of each one that I found. I collected a total of 1,738 unique Access Points. When you look at this data you have to put it in context so let me set out a few things for you to consider. First of all, I did not cover all of Springfield. Second, most of these Access Points are probably business related since I didn't drive through a lot of residential areas. Third, some of the open access networks are possibly hot spots and are not really open access (I assume that the number of hot spots is <10%). Fourth, RSNA access points are made for businesses so even if there are a large amount, a lot of them probably connect to the same network.

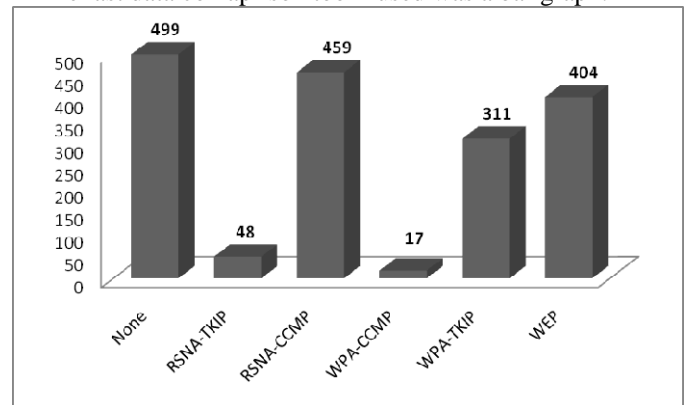
The specific numbers relating to each type of security can be found in the table below:

Security Type	Number Found
None	499
RSNA-TKIP	48
RSNA-CCMP	459
WPA-CCMP	17
WPA-TKIP	311
WEP	404

Another way to see the data is through a pie graph. I created a percentage based pie graph to more easily understand my results.



The last data comparison tool I used was a bar graph.



From these figures you can immediately tell a few things. WEP, WPA-TKIP, RSNA-CCMP, and no protection are all pretty equal compared with the other two technologies. The terms used in my charts are slightly different than what I have talked about in the past so let me relate them for you. RSNA is a server based authentication method that can have two types of encryption, TKIP and AES through CCMP. Server based authentication allows a server to take care of authentication allowing for different users to have different passwords and still be able to connect to the WAN. This type of authentication is used on larger scale operations such as business and education. Missouri State University uses RSNA based authentication on most of its Access Points. TKIP is the older version of protection and CCMP is the newer 'better' protocol based off the AES algorithm. WPA-TKIP is regular WPA, while WPA-CCMP is analogous with WPA2. WPA uses the

older TKIP algorithm with a PSK. WPA2 uses the AES algorithm, CCMP, and a PSK for security. WPA2 will only work with CCMP and AES. Since RSNA is used in a business setting and in a vast majority of cases you can see that businesses using it seem to keep their infrastructure up to date. Most of the RSNA Access Points are using WPA2 based encryption. However, if you look at the PSK based Access Points that are usually found in smaller companies and homes, you can see that a vast majority are WPA.

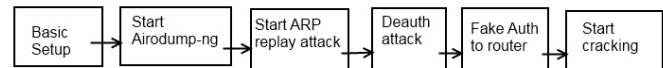
V. WEP

WEP, Wired Equivalent Privacy, came out with the first iteration of Wi-Fi. This was the first type encryption that was available to users of Wi-Fi and because of this, it is the most widely recognized by the average user. WEP is supported by almost every Wi-Fi device, however it has been deprecated. As with many dead technologies, its deprecated status doesn't prevent users from implementing it. It is still one of the most common types of encryption deployed. WEP uses the RC4 encryption algorithm for confidentiality. It also has integrity verification using a CRC-32 checksum. WEP's authentication is based off of a single password that is used to connect to the router, a PSK.

Unfortunately WEP is easily breakable. Its breakability comes from the implementation of RC4. RC4 is a stream cipher. Stream cipher encryption works by using a short key and then stretching it to an infinite pseudo-random key stream. Then you take the pseudo-random key stream and XOR it with the regular text, this produces the encrypted message. Since both the client and Access Point know the key, it is then XORed again upon arrival at the destination to recover the original text. This creates a problem however, if the same key stream is used on more than one packet and those packets are captured, the person who captured the packets can find the unencrypted text and then eventually the key. To avoid this the designers of WEP change the key stream using Initialization Vectors. Initialization Vectors manipulate the shared key and make it so that you will never have the exact same stream key for a packet. The Initialization Vector is then appended to the packet for the receiving side. The key to breaking WEP is to capture these Initialization Vectors (IVs). Their construction directly correlates with the password that was used to authenticate to the WLAN.

To capture the IVs and penetrate a WEP encrypted network you will need a few free products. If you don't want to go through the trouble of gathering all the tools, you can download a Linux distribution like BackTrack that has them all already installed. BackTrack can be used as a bootable ISO, bootable USB devices, or by just installing it. BackTrack is also free. I used this approach because it was by far the easiest way to get going quickly. I used the Aircrack-ng suite for the actual capturing and cracking of my test router's WEP. Below are the steps I used to penetrate my wireless. I used the ARP request replay attack to generate IVs faster. This attack works as long as someone is associated with the router. They don't have to be surfing the internet or anything, they just need to be associated. All the instructions are ran in terminal. Right

before the instructions I made a diagram that is a brief overview of the process.



1) Change the wireless mode to monitor. wlan0 is the name of my wireless/interface card. You can find yours using iwconfig in BackTrack.

```
iwconfig wlan0 mode monitor
```

2) Set your wireless card to up. This basically turns it on.

For me it is:

```
ifconfig wlan0 up
```

3) Now you need to create a folder to store you're captured packet dump file in.

```
mkdir ~/dump
```

4) Start up airodump-ng for capturing packets. You may need to start it up first without capturing to a file to find the channel and BSSID of your target AP.

```
airodump-ng -c 6 --bssid 00:23:AE:17:BC:C7 -w ~/dump/arr wlan0
```

-c is the channel of the target access point. --bssid is the BSSID, also the MAC, of the target access point. -w is where to write your log to.

5) In a new terminal window start your ARP replay attack.

When using this attack aireplay-ng will listen for an ARP request then replay it back to the router. The ARP request will contain IV vectors. This is a great way to generate a lot of IVs. It took me about 5 minutes to generate 40,000 IVs.

```
aireplay-ng --arpreply -x 150 -b 00:23:AE:17:BC:C7 wlan0
```

--arpreply is the attack to use while -x 150 says to use a transmission rate of 150 packets per second. This is a good number because your wireless card won't pick up more than that most likely. -b is specifying the BSSID of the target and wlan0 is the interface to transmit over.

6) Open a new terminal window. Inside this window we will be performing a deauthentication attack. This sends deauthentication packets to the clients of the router and disconnects them from it. Since by default you will reconnect this will generate an ARP request which is just what we need for an ARP replay attack.

```
aireplay-ng --deauth 5 -a 00:23:AE:17:BC:C7 wlan0
```

--deauth tells aireplay-ng to do a deauthentication attack. The 5 after it is how many deauthentication packets to send. -a 00:23:AE:17:BC:C7 specifies the target of the attack and wlan0 is my wireless interface.

7) Now you need to fake authenticate with the target AP for the ARP replay attack to work properly.

```
aireplay-ng --fakeauth 0 -a 00:23:AE:17:BC:C7 wlan0
```

--fakeauth specifies the attack type. The 0 behind fakeauth is the re-association delay. -a is the target and wlan0 is the interface.

8) Last you will be starting up the password cracker. It can run while you are still gathering packets and it will automatically retry cracking at certain IV collection intervals. However, you should wait to start it until you have about 25,000 under the #Data column in your airodump-ng window. #Data is the number of IVs collected. If you don't wait until you have 25,000 IVs collected the attempted cracking process

could take longer than it would to gather enough IVs for the next interval of cracking. This would slow the cracking process down.

```
aircrack-ng -z ~/dump/arr-01.cap
```

-z tells aircrack-ng to use the PTW cracking method and ~/dump/arr-01.cap is the packet capture being created by airodump-ng inside the folder we made earlier.

9) Wait about 10 minutes maximum and the password should be broken. I was able to do it in under 6 minutes with all the commands typed in and just pressing enter in each of the boxes. You will now want to stop all the running process. I would like to point out that there are other attacks that you can use to accelerate the speed of entry as well. You can use a combination of these attacks to gain access very quickly, however waiting a couple extra minutes wasn't a big deal to me. I have seen reports of people breaking into WEP in under a minute. Another factor that wasn't present when I did my testing was traffic. I had my iPhone connected to my router to deauthenticate. It wasn't browsing at all though. If it and other computers were actively using their Wi-Fi connections the attack would have gone a lot faster because the browsing activity would create even more IVs.

This is the reason WEP shouldn't be used. I could show someone with horrible computer knowledge how to do this rather easily and it would be just as effective as someone with years in penetration testing doing it. Unfortunately around 25% of the routers I encountered in my survey were WEP. The next iteration of security for typical home use is much more secure in implementation.

VI. WPA-PSK

Wi-Fi Protected Access, or WPA, was the second security method developed and implemented in Wi-Fi communication. There are two versions of it. The PSK version of WPA uses a Pre-Shared Key, just like WEP. This is generally used in home or small business environments. I will be discussing the other version of WPA later. WPA was designed to replace WEP because of the security problems related to WEP.

WPA uses the Temporal Key Integrity Protocol (TKIP) to encrypt its key. TKIP was defined in the 802.11i specification in a section regarding encryption. WPA-TKIP is not completely new though. It actually builds on WEP and tries to improve it. WEP and WPA-TKIP both use the RC4 algorithm, but unlike WEP for TKIP the key is always 128 bits long. One vast improvement in TKIP is that the key for each packet is different. The key is generated using the ESSID, the name, of the Access Point, a sequence number assigned to each packet, and a base key. The sequence number is used as an IV in WPA. This plugs a major security hole that was present in WEP. It changes the IV every time preventing any type of packet collision based attacks like the ARP replay attack that I used. The base key remains the same when using WPA-PSK and this is the chink in its armor.

Since the base key remains the same for WPA-PSK you can use brute force methods to gain access to the AP. The only thing you actually need to be able to brute force is the 4 way association handshake of a client. Once you are able to capture this, you can leave the premises and crack the

password at your leisure. The easiest way to get the four way handshake is to perform a deauthentication attack on a currently connected client. Then when that person reconnects automatically (this is the default setting), you will get the handshake packets you need. This setup is still considerably more secure than WEP. It also would require a more determined person to break into a WPA connection. Raising the bar to WPA may just make an attacker find an easier target. coWPAtty and aircrack-ng are both programs that allow you to try to brute force the PSK for WPA Access Points. Direct brute forcing can take a long time with WPA. When you directly brute force the WPA key you go through all possible keys until you find a matching one. Since there are a ridiculous amount of possibilities this method will take a long time unless you get lucky. A flavor of this type of attack that could possibly be more successful is using a password instead of every possibility. This still takes awhile though and the time grows with the more passwords you add to your list. There is a way to get around the aforementioned slow methods though.

Pre-computed Look Up Tables are a great way to perform fast WPA-PSK breaking. There are many advantages and a couple disadvantages involved with this method though. First let me explain what a Pre-computed Look Up Table is and how it operates. A Pre-computed Look Up Table is a table of values that offers a time-memory tradeoff. Basically everything you want to know is computed before hand and stored inside the table, then when you need to use it you just look it up in the table. The tables used for WPA-PSK cracking are generated by taking the list of things you want to be in the Pre-computed Look Up Table (PCLUT) and running them through same process as an Access Point would if it were generating a key stream based off a PSK. You then store the results in the PCLUT. The PCLUT I use for WPA-PSK cracking is a combination of the 1,000 most common ESSIDs and the 1,000,000 most common pass phrases. It is about 40 gigabytes though. To me, the storage trade off is very small compared to the benefits especially with the low cost of hard drive space. Normal brute force and password list based cracking can compare about 3 order of magnitude slower than when you are using a PCLUT. The other downsides of this attack is the immense amount of time spent to create the PCLUT and the fact that it only works for those specific ESSIDs and passwords. This method will grant you full access to the network and allow you to decrypt all packets. There are a couple of other attacks based on weaknesses in the CRC32 checksum, but they don't allow you to decrypt all packets and get on the network.

A new attack on WPA has been discovered by some people in Japan at Kobe University and Hiroshima University. As of now it is called the Practical Message Falsification attack or the Ohigashi-Morii attack because the last names of the developers of the attack are Ohigashi and Morii. It is based off the Beck-Tews attack and doesn't let you decrypt all packets. With this attack you decrypt one of the smaller packets, like an ARP request. This takes about a minute using the Ohigashi-Morii attack method where previously it took 12 to 15 minutes with the Beck-Tews approach. Also, the older attack relied on Quality of Service (IEEE 802.11e) to be

active on the target Access Point, the newer version doesn't. Once the small packet has been decrypted you can modify the packet and send it back into the network. This enables you to perform ARP poisoning attacks on the network. As I stated before though, you cannot decrypt all the traffic on the network making this attack only slightly useful. You cannot figure out the encryption key, yet, with this method and therefore you will never be able to gain complete access to the network. You do figure out the current key stream being used by the packets, but it changes rather frequently so it is of limited use. This could be basis step for future WPA attack development that might eventually lead to be WPA being as useless as WEP, only time will be able to tell us the rest of this story.

Eventually WPA will probably end up like WEP because it is built on a technology with security flaws. This is very unfortunate considering almost 20% of the people in my survey are using this type of protection. Luckily though the newest version of wireless encryption, WPA2, doesn't share those flaws which I will talk about more later. There is also another more version of WPA that is available, RSNA-TKIP.

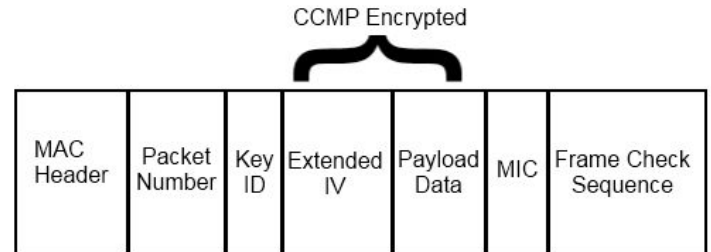
VII. RSNA-TKIP

RSNA-TKIP is vastly the same as WPA-PSK. They both use TKIP as you can see by the name and have the same issues. The main difference between the two technologies is that RSNA-TKIP doesn't use a Pre-Shared Key; it uses an authentication server. RSNA stands for Robust Security Network Association. When a client connects to a RSNA-TKIP network they don't authenticate directly with the Access Point, they get passed onto the authentication server where it checks to see if you should be added to the network. The authentication server tells the Access Point if it should add the client. The big security difference in RSNA-TKIP and WPA-PSK is that everyone can use different keys to connect and therefore have different key streams. So the main security vulnerability that the PCLUT attacks focus on has been taken reduced. Even though this technology stands up to attack better than the basic WPA-TKIP, most home user's can't use it. The cost of creating and maintaining an authentication server just for personal wireless is impractical and most business user's would rather upgraded to newer technology. Also, it still isn't completely secure. These reason show why only 3% of my survey use this technology. For the previous two reasons and many others the most recent iteration of wireless communication security was developed.

VIII. WPA-CCMP / WPA2

WPA2 is also referred to as WPA-CCMP. Even though it is called WPA2 it is very different from WPA. One of the few things they do have in common is that they can use a Pre-Shared Key. Everyone who connects uses the same password. After seeing the previous failures in WEP and WPA-PSK, you would probably assume that WPA2 has exposed flaws in its armor as well. Luckily, they seem to have gotten it mostly right this time. WPA2 began being certified by the Wi-Fi

Alliance in 2004. WPA2 uses AES and CCMP. This is what a WPA2 packet looks like:



AES, or Advanced Encryption Standard, is the encryption standard recommended for electronic data encryption by the National Institute of Standards and Technology (NIST). The AES algorithm was released by NIST in May of 2002. AES is an iterative, symmetric-key block cipher that can be implemented with different key sizes. In WPA2 a 256 bit key is used for encryption. The same key is used to both encrypt and decrypt the data with the AES. WPA2 also uses CCMP or Counter Mode with Cipher Block Chaining Message Authentication Code Protocol. CCMP is an encryption method that, in this case uses, AES for the message confidentiality. The counter mode makes it much more difficult to spot a pattern and therefore reduces the possibility of the encryption being broken. These two cryptographic techniques are used to create WPA2 and make it a very good contender against cracking.

Although cracking the encryption brute force style is highly unlikely with current technologies, there are ways to get into a WPA2 access point that uses a PSK. What we can do is use the same approaches as WPA. With Pre-Computed Lookup Tables of the most common passwords and SSIDs we can crack the password and be able to connect as long as that pair is in the tables. Once again all you need is the initial connection handshake captured to try to break the encryption and be able to figure the password out. You can also use a password list just like WPA. The likely hood you will get in is based off the whether the person who setup the access point used an uncommon password and ESSID. If the ESSID and password are common then the likely hood that you can get into the network in a reasonable amount of time is much greater.

IX. ROGUE ACCESS POINTS

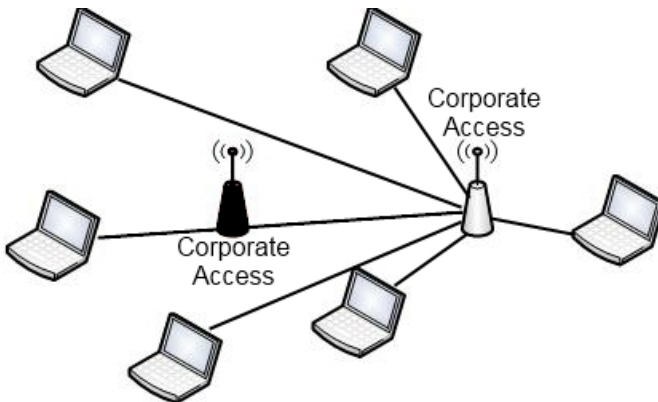
Most people don't know what a Rogue Access Point is. A Rogue Access Point is a device that is in the network or interfering with your network traffic and isn't suppose to be there. There are a couple ways you can look at Rogues Access Points.

One of the less malicious Rogue Access Points is when an employee (or anyone with network access) takes it upon themselves to install a router without administration approval or without letting the people in charge of security know. This type of thing happens most often in a business setting and usually employees don't think they're doing anything wrong. Also it is usually a less technical user that does it because the more technical users realize it is probably a breach of policy. This type of Rogue Access Point creates a large hole in any

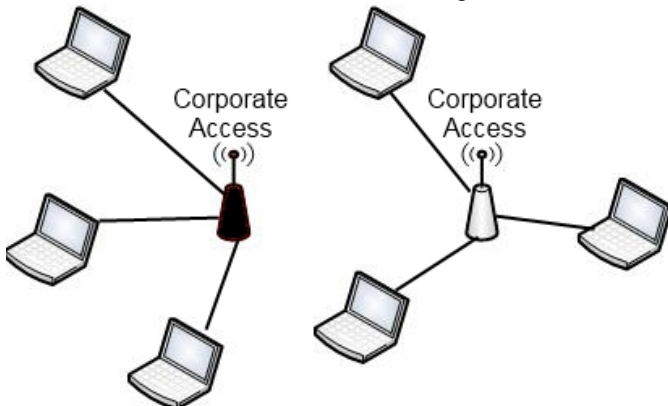
type of environment. It could lead to data compromise from the access point if outsiders connect and launch an attack from it. User education and network mapping are the best ways, in my opinion, to prevent this type of thing from happening. This is a less threatening version of a Rogue Access Point.

The more dangerous type of Rogue Access Point is introduced by an outsider targeting you. This type of attack is usually only found in a corporate or enterprise level domain because it requires a little more planning than just breaking a WEP key. When deploying this type of Access Point the attacker will impersonate a valid router that is on the target network. They will get an Access Point and name it the same as the router they're masking over. Then they will perform deauthorization attack on all the clients associated to the target router. Then the impersonating router will offer itself up for connection. Since almost all wireless users are setup to automatically reconnect after being disconnected they will reconnect. Some however will reconnect to the impersonating router because it has a better signal and others will connect to the actual router.

Before Deauthorization attack, Black is Rogue Access Point



After Deauthorization attack, Black is Rogue Access Point



There are several methods you can use to take the target router down to ensure that people will only re-associate with the attacker's impersonating router, but generally the attacker doesn't need to do this since someone will connect to the Rogue Access Point. To take down the target router the attacker does need to have a computer present, not just the impersonating router. Since no type of router identification is currently in use in the current Wi-Fi security technologies available, you are unable to tell if you have connected to the correct router. Now you may be wondering why it matters if a

client has connected to another impersonating router, this will be explained shortly.

The impersonating router is loaded with alternate firmware. The firmware allows you to take on the properties of a target AP and impersonate it. It uses the same ESSID and BSSID. One such firmware is called Jasager. With this firmware the router accepts any client that wants to connect and then when they authenticate it just accepts the authentication. It then logs the authentication information they used for that AP. The attacker can then log into the Rogue Access Point and download the authentication information logs. If the attacker wants to go further into the impersonation they can provide internet access to their Rogue Access Point. Then when the user tries to connect to a SSL or TLS encrypted website the Rogue Access Point will masquerade as that website intercepting their login information then passing it on to the actual site to not raise any suspicion. If the user is savvy they could notice this though because the domain certificate would not match the actual domain. However since most people don't check these so it is not a problem. It is also possible to use something like SSLStrip to intercept their login information for websites.

As you can tell this type of Rogue Access Point is much more of a danger to a business than the other. User education will not help in this case and neither will network scanning because the router isn't connected to the target's network. There do exist some defenses against Rogue Access Points though. There are three ways that I know of that allow you to detect Rogue Access Points.

The first method involves using a Wi-Fi sniffer. A sniffer is a program that allows you to view the current Wi-Fi traffic in the area. An example is the program that was used earlier to break into WEP, airodump-ng. With a sniffer you would have to physically walk around the building scanning the Wi-Fi activity. This can be very time consuming depending on the size of your facility. Another downside is that then you have to find some way of identifying the Access Points that are yours and ones that are Rogue Access Points. This can be difficult because a Rogue Access Point can change not only its SSID but MAC address to match the target AP. This is the least effective method for detecting Rogue APs for the reasons I mentioned.

The second method of detection involves the use of Wi-Fi probes in the network. Probes allow you to constantly monitor the Wi-Fi around your building simultaneously. You can then use these to detect any new APs that are trying to blend into your network or any new ones in range. This is a very effective method but is expensive. You have to deploy many probes to cover your building and in addition you have to have someone monitor the reports that are generated by the probe to see if it really is a problem or not. The probes also require more infrastructure. You have to run Ethernet to them so they can communicate and then power as well. There exists one last method of monitoring.

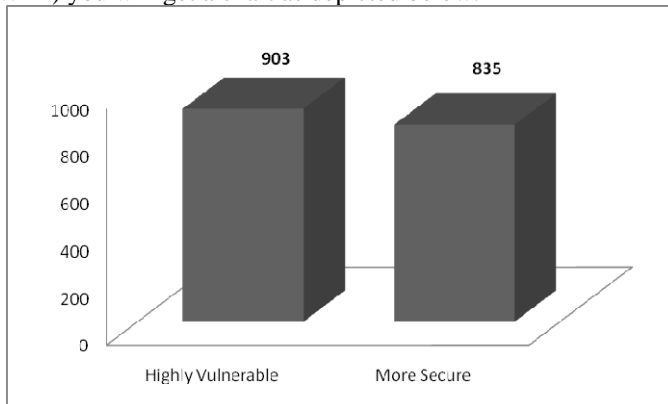
The last method is much like using a probe network to detect Rogue APs. A few router manufacturers have been equipping their APs with the ability to monitor Wi-Fi as well as act as an AP and service clients. This allows you to only have one box that does both the Rogue AP detection and client servicing.

The infrastructure costs are lowered because of this, but otherwise the same ideas apply to this method of Rogue AP detection as probe based detection.

Rogue Access Points can be a serious security hole that goes largely unaccounted for even in big business. Most companies don't want to spend the resource to be able to detect Rogue APs; for this reason a Rogue AP is a great attack vector that often isn't noticed. The amount of information garnered by this type of attack can be tremendous and you can use the same basic approach anywhere you want to implement the attack.

X. IMPLICATIONS

There are many ways to look at this data. Open access is the most common type of Access Point that I encountered. I also encountered a large amount of WEP Access Points and as I showed you, breaking into WEP is easy. All it does is offer a small amount of protection from those who aren't technical. Honestly, I could probably teach someone how to break into WEP in about 30 minutes. I could explain the process and show them how. To me, WEP is basically having no security. When you look at the amount of people with actual security versus the amount of people with none or very weak security (WEP) you will get a chart as depicted below.



In this graph I combined the numbers of people with no security and WEP. Then I combined the rest of the categories of security to produce the other number. As you can see from the graph they are about even. That means half the people with Wi-Fi that I surveyed are practically unprotected. Leaving your Access Point unprotected horrible practice.

Most people probably don't even realize that they're vulnerable. Most routers don't come with any type of access protection enabled by default. Out of the box they're vulnerable. This is to provide easy setup for non-technical users. If the person doesn't secure their network, they're leaving it wide open to even the non-technical user. It may be easier to setup, but you are leaving a wide open hole for anyone who wants access to your network.

One reason that you don't want to leave your network open like this is because people could gain access to your network then use your connection as a base to launch internet based attacks. It could also be used for other nefarious activities like downloading child porn or pirating music or movies. Anything that has been done over your network connection is going to be traced back to you. You will be assumed to be the

person participating in these activities. That means you will have the burden of proof to show that you are not the one committing these illegal acts. Solid evidence to prove you are not the one who was commit these acts is hard to come by, especially since no type of logs or tracking is done with almost all Small Office, Home, and Home Office users. There are other things that bad security effects as well.

If someone is able to connect to your network from your Wi-Fi Access Point they will have behind-the-firewall access to your network. Most companies that don't have a security professional working for them heavily rely (and even probably some that do have a security professional working for them) on their firewall to filter out any hacking or malicious attacks. The firewall is not part of the equation because of the way they're attaching themselves to the network. Since there is no firewall separating them from your network they have more direct access to machines that are located on the network. They can also use scanning tools like NMap to blueprint the network and Metasploit to exploit any vulnerabilities found. For a home setting they probably won't even need to go that far. Many computers share files to their local network which can then be accessed by the person on your Wi-Fi. Financial documents, personal information, and much more is probably stored on your computer and connecting behind the firewall gives an intruder direct access.

Until many of these problems are solved people are still going to be caught up in this. Unfortunately, there doesn't seem to be a trend of trying to fix this. Most manufactures still ship their routers with no encryption at all enabled by default, and the few that ship with encryption enabled use WEP. The ease of use is better with no encryption but the consequences of not protecting yourself and your network can be quite drastic.

XI. PROPOSED SOLUTION

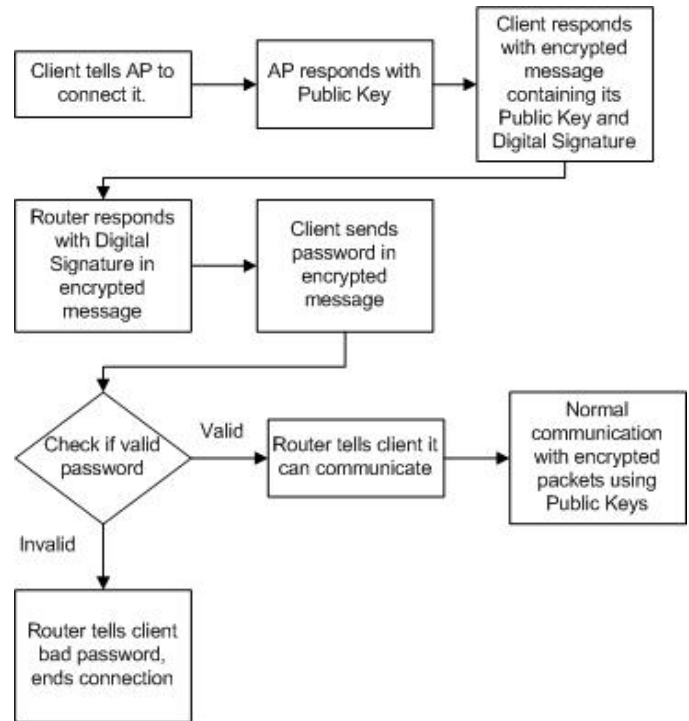
My proposed solution for making Wi-Fi communication secure uses Public Key Encryption. Public Key Encryption uses two keys, also called a key pair, and is an asymmetric version of encryption. One key is sent over to the person that wants to send you information to encrypt the data. The key that is sent doesn't need to be encrypted because it cannot decrypt data. Then the sender uses the key to encrypt the data and sends it to the receiver. Once the information has arrived at the receiver the secret key, or private key, is used to decrypt the information. It is a one way process, the sender cannot decrypt the information. If the receiver encrypted something with their public key and sent it somewhere, the recipient would not be able to decrypt it. This type of technology is used in SSL, TLS, PGP, and many other high end security applications today.

Another advantage of using Public Key Encryption is that you can implement Digital Signatures. Digital Signatures are a way to uniquely identify yourself. You can sign a message with your unique identification and then the receiver can check it to verify that you really sent the message. This is very important to ensure the integrity of a message for the receiver and one of the holes in other protocols.

I have a couple of solutions that would fix problems with wireless communication depending on how radically they protocol would be modified. WPA2 is a good way to encrypt information but the handshake is where it breaks down.

The first method would be to just wrap the handshake with Public Key Encryption. First the client would ask the AP for its Public Key. The router would respond in clear (unencrypted) text with its Public Key. Now the client can encrypt its Public Key and Digital Signature then send these back to the AP. Public Keys are designed to be sent without encryption, but the reason I suggest encrypting the clients Public Key is so that another person can't encrypt information and send it to the AP and impersonate the client. Since the AP has the client's Public key, the client can ignore any packets sent to it that aren't encrypted and then no one will be able to inject packets or tamper with the initiating connection. Now that the router has the clients Digital Signature it can verify that the incoming information really is coming from a specific client. It also has the Public Key for the client so it can encrypt its Digital Signature with clients Public Key and send it to the them. Now the client is able to verify that any information that is sent to it is from the router. Now both the AP and the client have each other's Public Keys and can encrypt data sent to one another. Now you can securely do the WPA2 handshake without fear. After the WPA2 handshake has completed you can just use the WPA2 method of encryption for the rest of the communication. This method would break all current attacks against WPA2 and be a slight modification to the existing authentication method. This is kind of like what was done with WPA where they just modified WEP to try and make it secure. Personally, I rather have a protocol that was designed for security completely at the outset, not a modified version of one so I have another solution.

This solution would involve making a new type of Wi-Fi encryption that is solely based on Public Key Encryption. It would be similar to the scenario I just described where the authentication handshake is encrypted using Public Key Encryption. The same starting steps would be used. The client would request the APs Public Key. The AP would send it to the client. The client would then encrypt its Public Key and Digital signature than send those to the AP. After that the AP would send its Digital Signature (encrypted) to the client. This is where the difference comes it. Instead of switching off to WPA2 based communication you just keep using Public Key Encryption. Here is a diagram of the process.



You would still need to implement some type of authorization after the initial exchanges. So once the secure connection is established you would verify a users password with PSK or RSNA methods. The same method of authentication could be used because you have a secure connection to communicate over. This would cover all your bases and make a very secure connection immune to the current Wi-Fi attacks. Overall this technology would be better than the current methods, but all good things have downsides.

There are a couple foreseeable downside to Public Key Encryption though. It is more resource intensive then other methods of encryption. I consider this a minor disadvantage though because of the speed at which our technologies advances. With multi-core CPUs all over I think this process would be rather easily threadable to allow fast processing in more modern processors. It may be a few years before multi-core processors find their way into lower end devices, but it would probably take longer than that long to get the specification written and approved by the IEEE anyway. The other downside is that Rogue Access Points can still be a problem. I did come up with a way to work around this and prevent them though. Unfortunately, it wouldn't be very easy for the user to implement. If you could get the Digital Signature of the APs you would like to communicate with before you actually connect, you could store them. Then your wireless could be setup to only connect to the APs that you have stored. This would probably only happen in a corporate environment because the user would have to physically add the Digital Signatures of the APs to their computer before connecting. I think most home users wouldn't want to go through this process, but it is still a rather easy way to prevent Rogue Access Points compared to the current methods and I think it would be really useful for enterprise level users.

XII. CONCLUSION

Wi-Fi security isn't something we can ignore. It is something we encounter in everyday life. Since it is such a prevalent thing we should really focus on making it secure from attack. Hopefully this paper has contained a lot of useful information that you can apply in your career. I have presented you with the issues related to the current methods of wireless encryption and authentication. I also came up with a suggested solution for the problems that currently exist. I believe my solution is very viable and would create a truly secure architecture for Wi-Fi communication. If anyone has any input or noticed any holes in my solution I would be grateful if they brought them to my attention.

XIII. RESOURCES

Although none of my paper is directly quoted from articles I found during the writing of my paper I would like to include all the resources I used in my research. Below is a list of the articles I used.

I will also release all the stuff that I used in my paper, including the logs and program I used to merge them on my website. If you are interested my website is located at <http://brokenbytes.info>. The information will be posted on my blog as well as in the Tutorials section.

Bradbury, Danny. "Wireless Security: the Unseen Threat." Computer Weekly 26 Feb. 2008: 28-30. Business Source Premier. EBSCO. MSU, Springfield, MO. 5 Apr. 2008

Dysart, Joe. "The Case for Wi-Fi Security." American School Board Journal 195 (2008): 58-59. Academic Search Premier. EBSCO, Springfield, MO. 5 Apr. 2008. Keyword: Wireless Communication.

Issac, Biju, and Lawan Mohammed. "War Driving and WLAN Security Issues." Information Systems Management Fall 2007: 289-298. Business Source Premier. EBSCO. MSU, Springfield, MO. 5 Apr. 2008.

Loo, Alfred. "The Myths and Truths of Wireless Security." Communications of the ACM Feb. 2008: 66-71. Business Source Premier. EBSCO. MSU, Springfield, MO. 5 Apr. 2008.

Wright, Joshua. "Scary Wireless Security Threats." Network World 3 Mar. 2008: 12. Business Source Premier. EBSCO. MSU, Springfield, MO. 5 Apr. 2008.

Wired News,
<http://www.wired.com/threatlevel/2009/10/vulnerable-devices>

MetaGeek inSSIDer Wi-Fi Scanner,
<http://www.metageek.net/products/inssider>

Wi-Fi Alliance, <http://www.wi-fi.org/index.php>

Nabble Forum Thread On WEP,
<http://n2.nabble.com/Wired-Equivalent-Privacy-WEP-t1378555.html>

Berkley Computer Science Program article on WEP,
<http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html>

Cracking WEP Video Tutorial, <http://forums.remote-exploit.org/tutorials-guides/7872-xploit-z-video-volume-2-e-z-no-client-korek-chopchop-attack-tutorial.html>

Pre-computed WPA and WPA2 Look Up Tables Information, <http://www.renderlab.net/projects/WPA-tables/>

Pre-computed WPA and WPA2 Look Up Tables Download Links, <http://rainbowtables.shmoo.com/>

MSDN RNSA Overview, <http://msdn.microsoft.com/en-us/library/aa503310.aspx>

About.com WPA Encryption Cracked,
<http://netsecurity.about.com/b/2008/11/07/wpa-encryption-cracked.htm>

Video tutorials, <http://infinityexists.com/>

Information on TKIP,
<http://www.networkworld.com/reviews/2004/1004wirelesskip.html>

New WPA attack techniques,
<http://www.networkworld.com/news/2009/082709-new-attack-cracks-common-wi-fi.html>

Using Pre-Computed Lookup Tables from Renderman and H1kari with airolib-ng, <http://forum.aircrack-ng.org/index.php?PHPSESSID=954aaf5dd99f687475ad3578105fcd8&topic=4383.new>

Explanation of what Rainbow Tables are,
<http://www.codinghorror.com/blog/archives/000949.html>

Microsoft's WPA Wireless Security for Home Networks,
http://www.microsoft.com/windowsxp/using/networking/expert/bowman_03july28.msp

NIST AES,
<http://csrc.nist.gov/publications/fips/fips197/fips-197.pdf>

MSDN AES Information, <http://msdn.microsoft.com/en-us/magazine/cc164055.aspx>

Aircrack-ng article on cracking WPA/WPA2,
http://aircrack-ng.org/doku.php?id=cracking_wpa

Simple Guide to Aircrack-ng,
<http://ubuntuforums.org/showthread.php?t=528276>

Jasager Rogue Access Point Firmware,
<http://www.digininja.org/jasager/>

Jasager help and installation instructions,
<http://hak5.org/forums/index.php?s=c9287f08754d883b912f20a276165a83&showforum=49>

Introduction to RSA,
<http://mathcircle.berkeley.edu/BMC3/rsa/node4.html>

WPA attacks article,
<http://arstechnica.com/security/news/2008/11/wpa-cracked.ars/2>

WPA attack article,
<http://www.mobileparadigm.com/2009/09/09/new-wpa-attack/>

Cracking WPA with airolib-ng database,
http://www.metacafe.com/watch/yt-aR7Vpfk_HDU/wpa_cracking_with_airolib_ng_database/

BackTrack 4 Download Site, http://www.remote-exploit.org/backtrack_download.html