2010 2nd International Conference on Education Technology and Computer (ICETC)

Wireless Hacking - A WiFi Hack By Cracking WEP

S Vinjosh Reddy*
Computer Science Engineering
Jagan's College of Engg & Tech
Nellore, India
vinjoshreddy@yahoo.com

K Rijutha
Computer Science Engineering
Jagan's College of Engg & Tech
Nellore, India
rijuthakona@yahoo.com

K Sai Ramani
Computer Science Engineering,
Jagan's College of Engg & Tech
Nellore, India
ramanikompella@gmail.com

Sk Mohammad Ali
Deccan College of Engg & Tech
Hyderabad, India
mohammadmunna.ali@gmail.com

CH. Pradeep Reddy
Assistant Professor, SITE
VIT University, India
pradeep1417@gmail.com

Abstract **- Wireless Local Area Networks frequently referred to as WLANs or Wi-Fi networks are all the vehemence in recent times. People are installing these in houses, institutions, offices and hotels etc, without any vain. In search of fulfilling the wireless demands, Wi-Fi product vendors and service contributors are exploding up as quickly as possible. Wireless networks offer handiness, mobility, and can even be less expensive to put into practice than wired networks in many cases. With the consumer demand, vendor solutions and industry standards, wireless network technology is factual and is here to stay. But how far this technology is going provide a protected environment in terms of privacy is again an anonymous issue. Realizing the miscellaneous threats and vulnerabilities associated with 802.11-based wireless networks and ethically hacking them to make them more secure is what this paper is all about. On this segment, we'll seize a look at common threats, vulnerabilities related with wireless networks. And also we have discussed the entire process of cracking WEP (Wired Equivalent Privacy) encryption of WiFi, focusing the necessity to become familiar with scanning tools like Cain, NetStumbler, Kismet and MiniStumbler to help survey the area and tests we should run so as to strengthen our air signals.**

Keywords- WiFi Hacking; WEP; Kismet; Cain; NetStumbler

I. INTRODUCTION

The Institute of Electrical and Electronics Engineers (IEEE) provides 802.11 set of standards for WLANs. The wing ".11" refers to a subset of the 802 group which is the wireless LAN working group [1, 14]. Many industry groups are involved in work with wireless systems, however the IEEE 802.11 working group and the Wi-Fi Alliance [15] came out as key troupes. At present, Wi-Fi schemes shaped a demand in the market and they are in reality everywhere. But by this augmented exposure comes the amplified risk, the extensive use of wireless systems has facilitated make them a huge target than the IEEE ever negotiated for. (Not many flaws such as the Wired Equivalent Privacy (WEP) in the 802.11 wireless network protocol help things, either.). Through the expediency, cost reserves, and efficiency gains of wireless networks raise security risks. The regular security issues, like weak passwords, spyware, and missing patches are not the things that are going to matter. Networking with no wires brings in an intact new set of vulnerabilities [2, 3] from an entirely different point of view. Here comes the concept of ethical hacking. *Ethical hacking* [20, 21], occasionally called as white-hat hacking is the use of hacking to check and advance the defenses against *unethical* hackers. It may be compared to access testing and susceptibility testing, but it goes even deeper. Ethical hacking entails the usage of same tools and practices the bad guys make use of, however it also involves wide range forefront planning, a set of precise tools, multifaceted testing methodologies [21], and adequate report to fix any problems before the bad guys exploit our privacy.

II. NEED TO TEST OUR WIRELESS ARRANGEMENTS

Insecurity of Wireless networks is on track ever since the premature days of the 802.11b standard of 1990s. The standard's initiation, major 802.11 limitations [2], such as physical security, encryption flaws has been discovered. Because of these, two wireless security standards have come out to help struggle back at the enemy:

978-1-4244-6370-1/$26.00 © 2010 IEEE

Wi-Fi Protected Access (WPA) [4, 10]**:** Developed by the Wi-Fi Alliance, served as an intervening standard to fix the well-known WEP vulnerabilities.

IEEE 802.11i (identified as WPA2) [3, 19]**:** An official IEEE standard, that integrate the WPA fixes for WEP with additional encryption and authentication mechanisms.

Like many security standards, the problem with these wireless security solutions is not that they don't work, it's because of the network administrators who are resistant to change and don't fully implement them. They don't like to reconfigure their wireless systems and don't want to implement new security mechanisms feeling that the management becomes difficult. These look like ignorable things, but they depart many wireless networks defenseless and waiting to be compromised. Though WPA, WPA2 and the various other wireless protection techniques described in this paper have been implemented, network might still be at risk. As up to our practice, we have seen many providing some security mechanisms either the above ones or the other. But even with many wireless security standards and vendor solutions [5] available, the greater parts of systems are still wide open to assail.

III. THE DANGERS OUR SYSTEMS MUG

Just going deep into the ethical-hacking process, we should know a couple of terms we'll be using throughout this paper. They are,

Threat: A threat is a sign of target to cause disturbance within an information system. A few paradigms of threat agents are hackers, annoyed employees, and malware such as viruses or spyware that can inflict disorder on a wireless network.

Vulnerability [11]**:** It is a flaw inside an information system that can be browbeaten by a threat. We'll be seeking out Wireless network vulnerabilities all through this paper. Going further than these nuts and bolts, precise things can happen when vulnerabilities have been exploited by a threat. This state is called *risk.* Risks allied with vulnerable wireless networks [5, 21] comprise,

- Full access to files
- Stolen passwords
- Wired network back door entry points
- DoS attacks [11] causing productivity losses
- Violation of laws and regulations relating to privacy, corporate financial reporting, and more
- Zombies: A hacker attacks other networks using our system making us look like bad guys

IV. CONSIDERING CHALLENGER

The problem really is not with these wireless networks, in and of themselves. It's with the malicious hackers waiting there for an opportunity over vulnerabilities to make our work thornier. So as to better defend your systems, we have to think like a hacker. Even though it's impossible to reach the identical wicked mindset as the punks, we will be able to see where they're approaching from technically and what could be their upshot on us. For beginners, hackers are probably to target systems that need minimum effort to break in. Mostly the primary object is an organization having one or two wireless APs. We've found that smaller wireless networks undoubtedly work in hacker's favor, for many reasons. Those are accurately the class of things that elegant hackers make use of. Yet undersized networks aren't the merely vulnerable ones. There are many other flaws that hackers can use in networks of all extents.

Right through this paper, we seek to point out the ways that cyber crooks work while performing specific hacks. The more aware you are to the hacker state of mind, the close our security testing will be leading to secured wireless network. Hackers usually don't want to lift our information or crash our systems. They habitually just want to prove to themselves and their allies that they can crack in. Sometimes these guys want to use a system for attacking other people's networks under mask. Otherwise they're jaded, and just yearn for seeing the information flying through airwaves and finally for the taking. A person like Uber hackers go anywhere the money is literal.

V. FOREFRONT VIEW OF NECESSITIES

Ahead of going down the ethical-hacking toll road, it's vital that we plan everything in advance. This take account of:

1. Acquiring permission from our boss or project sponsor or client to carry out our tests
2. Over viewing testing objectives
3. Reconciling what tests to run
4. Grasping the ethical hacking techniques [21] before carrying out our tests.

VI. COLLECTING RIGHT TOOLS

Picking up the appropriate security testing tools is again an important part of the ethical-hacking process. Just for the reason that a wireless hacking tool is premeditated to perform a certain test, but that doesn't signify it will. We may have to nip our settings or locate another tool altogether. Also we should be aware of potential for *false positives* [21] (screening that there's vulnerability when there's not) and also *false negatives* (screening that there's no vulnerability when there is). The subsequent tools [6] are some of our likings for performing tests on wireless networks and are crucial for executing wireless hacks:

- Google - yep, a Web site which is a huge tool
- Laptop computer
- GPS satellite receiver
- Network Stumbler [16]- network stumbling software
- AiroPeek - network analysis software
- QualysGuard - vulnerability assessment software
- WEP encryption cracking software [17]

We'll get on to work with these tools in further detail shortly on in this paper, when we present some specific wireless hacks.

VII. INSPECTING FOR PROTECTION

Once if we get everything geared up, it's time to make our sleeves and dig up our hands dirty by carrying out different ethical hacks against our wireless set-up. There are cluster of security tests we can perform to know how feeble our wireless systems are to bother. The outcomes of these tests will definitely show us what security punctures may or may not be fixed to build a more secure wireless network. We will sketch out various counteract measures we can employ to fix the weaknesses we identify. In the next few segments, we will outline a variety of security attacks to establish the root for vulnerability tests we'll be operating against our wireless network.

A. *Unethical Attacks*

These kinds of attacks take advantage of human weaknesses like lack of consciousness, negligence and ignoring strangers. We also enclose physical vulnerabilities which can make an invader have a chance on firsthand access to our wireless devices. These attacks [4, 21] incorporate,

- Flouting into wireless devices that clients mounted on their own and left unsecured
- Some sort of attacks where a hacker fake as somebody else and persuade users to give out excessively much information about our network
- Unauthorized assessment of APs, antennae and some other wireless infrastructure to reconfigure and confine data off it.

B. *Attacks Concerning Network*

There are a group of techniques the dire guys can use to shatter within our wireless realm or at any rate leave it wilted in a nonworking state. Network based attacks [21] comprise of,

- Mounting mischief wireless APs and dodging wireless clients into linking to them.
- Holding data off the network from a distance by under our own steam etc.
- Attacking the transactions of the client in network by sending up MAC addresses, setting up a medium (Slotting in a wireless system in between an AP and wireless user) and more
- Abusing network protocols
- Carrying Denial-of-service (DoS) attacks
- Jamming RF signals [12]

C. *Attacks Concerning Software*

Since the security harms with the 802.11 protocol [12] weren't adequate, we have to be anxious about operating systems and utilities on wireless-client machines readily vulnerable to exploit. Now we'll some of the software attacks:

- Hacking the operating system and further applications on wireless-client gear
- Contravening through default settings like passwords and SSIDs [12] that are effortlessly known
- Cracking WEP keys [4] and pattering into the network's encryption scheme
- Getting way in by the use of feeble network authentication methods

VIII. SIMPLE HACK WITH A DRIVE

Now, we just try to go out and look around scanning for open Wi-Fi hotspots [5]. In many circles, this is deemed as a sport and is increasing in fame crossways the globe. Any Windows machine enabled with Wi-Fi has the ability to scan hotspots by installing either NetStumbler or Cain [16]. When we go with Linux, it's a different chronicle. Because the Linux lets direct access to the hardware and there are also several issues to regard. These embrace compatibility, right drivers plus familiarity of iwconfig or else related configuration utility for use of card in loose mode. Almost all 'Live Linux' divisions look after majority of work for us when the Wi-Fi card includes compatible chipsets. PRISM 2 [17] is the regular and sounded Wi-Fi chipset meant for Linux use. Another unit, which became popular by its ease of use and working ability with most Linux environments, is the Orinoco Gold card. With an NDIS driver, we can use Windows based cards in a Linux environment. However, when it comes to scanning these are not going to work as a result of inability to access the hardware openly.

Nearly all Windows support scanning utilities make use of a technique of scanning called 'Active scanning' as of the limited access to the hardware. While we scan for Wi-Fi with active scanning, a request has been sent through our device on every channel and records all replies. An immense traffic is produced and it is noisy even.

A. *Going For Tackle*

An active Windows based scanner we normally use for producing the information needed to map Wi-Fi hotspots along with SSID, Encryption and GPS coordinates is the NetStumbler. Because the program continuously yelps out 'ANY ACCESS POINTS ARE AVAILABLE NEAR', the retorts are more copious. NetStumbler make use of Windows drivers for accessing Wi-Fi card, ensuing the Wireless Zero Configuration [8] to shut down once run. In WinXP, this Wireless Zero Configuration permits the OS to

locate available Wi-Fi networks. When we go on road, this will become a big problem for connecting to an access point. The simplest mode to resolve this problem is just saving the NetStumbler data, closing program and refreshing the available networks. Cain & Able is the finest one we find as open outstanding auditing program for windows platform. It exercises ARP [12] poisoning, a VoIP logger, password crackers, and also has a built in Wi-Fi scanner. It overcomes the negative aspect that we have with NetStumbler since a third party driver 'WinPcap' [8] (for most low level network programs) is used by this application. The volume of Access points with Cain & Able is not up to the mark as NetStumbler does, so the selection lies on a preference.

Kismet [17] is trendy for the reason that it uses 'Passive scanning' schemes and it won't interfere with Wi-Fi signals or network traffic. With a passive scanner, the information is logged only if an access point is transmitted. Mostly it's impossible to spot as giving us flush information than the stated counter parts. Once sufficient traffic is produced or dynamic traffic go by, the IP address range of the AP can be grabbed with no log in. If the network doesn't use DHCP [12], the access point's IP address can be known very handy.

B. *Area Scan*

The Fig 1. shown below is a sample of data that may be similar to the data we will also hit upon. We have to be very vigilant that the proportion of Encrypted Vs. Non-encrypted networks will show a discrepancy from place to place. One more significant thing to stare at is the SSID names lists. Most of them are with the default name. The routers of the Broadband using default name will possibly have the default passwords on them too and these are further attention grabbing targets than a concealed SSID.

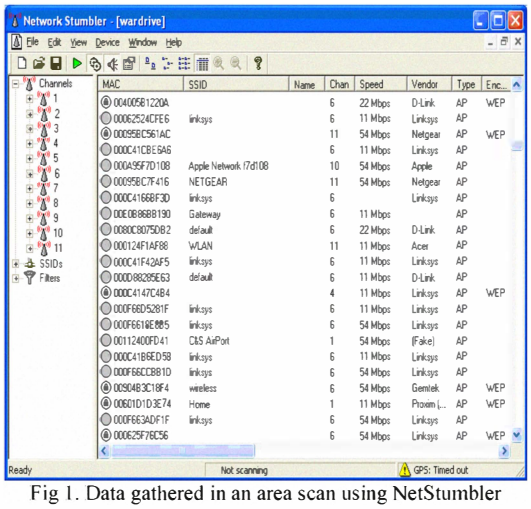

Fig 1. Data gathered in an area scan using NetStumbler

The NetStumbler file (NS1), may be uploaded to majority of the Wi-Fi public depositories online for the people to spot like wifimaps.com, wigle.net etc. At the moment we've used NetStumbler, Cain or Kismet to gather the data and now we are able to start our hunt for cracking the WEP. The primary section of the data we should look after to initiate with is the SSID, MAC address and also the Channel.

C. *Congregating Information*

With the desired information, an illicit now starts to hit the WEP or he installs a Warcracker (tiny computer premeditated for automated information gathering and cracking procedure) which is accessed remotely or could be picked up at a short time. Usually to stay legal when involved, it's a fine initiative to build a lab environment among various Wi-Fi AP's as targets and many systems having WiFi cards to intensify the quantity of attractive traffic. This attractive traffic comprises of key negotiation packets and makes us to collect sufficient information by snuffling to crack the WEP key in a lesser span of time. The traffic we are talking about can be produced with programs like Aireplay and Void11 [6]. These create the necessary WEP initialization vectors to crack. Along with the revealed ones, Airodump [17] is simple to use and assists this process.

For instance, let the target Wi-Fi AP has WLAN SSID, MAC address of XX:XX:XX:XX:XX:XX and also the channel of 9. Now, we will utilize Airodump to arrest the feeble IV packets and begin the passive packet capture in to a file named keygen. The command supposed to be like the following in the root level command line shell,

root@home[\]# airodump wlan0 keygen XX:XX:XX:XX:XX:XX [8]

All of the interesting packets are saved in the file called keygen.txt which will be used in a very short period of time. But, except we have much of time on our hands, we may perhaps yearn for speeding up the process slightly. The frequent tool we use to deauthenticate [12] the wireless clients is Void11. It works efficiently in a lab environment, but it sets off prompts in a company background and is an indication of a probable attack of our system. Through a Kismet scan, there's a client system with YY:YY:YY:YY:YY:YY as the MAC address. It is essential because we will target the Wireless Access Point along with this MAC address for generating the information we require. With Void11, the command looks as follows,

root@home[\]# void11_penetration -D -s YY:YY:YY:YY:YY:YY -B XX:XX:XX:XX:XX:XX wlan0 [8]

For even more less time, several people employ Void11 in combination with Aireplay. Aireplay captures convincing traffic and repeats the traffic to send it to the AP to produce more righteous traffic.

root@home[\]# aireplay -i wlan0 -b XX:XX:XX:XX:XX:XX -m 68 -n 68 -d YY:YY:YY:YY:YY:YY [8]

When the programs Void11 and Aireplay are running, Airodump will capture the packets used in the cryptanalysis. With numerous systems producing the traffic, a sniffer records data faster and raise the nominal time to reveal the key. To save the traffic to a file, we use Airodump and then Aircrack[17] receive that file to assail the key. The entire trick is forcing the Wi-Fi appliance to cause the righteous traffic.

D. *WEP Getting Cracked*

After all, we now have the file prepared to be thrown to the massacre. This is the time to consider Aircrack which will do the rest of our work. It uses Airodump data of the file and starts crack process for generating the right key. For the 128 bit WEP [7] to break, the file should hold 200,000 to 700,000 distinctive IV packets. Thinking that we cover a sufficient amount of file, we hit the file to obtain the key. With Aircrack, the command in the root level command line looks as follows,

root@home[\]# aircrack -f 2 -m XX:XX:XX:XX:XX:XX -n 128 -q 3 keygen.cap* [8]

If the key has been revealed, we will see the message 'KEY FOUND!'. We've made Wireless Access Point to compromise and now it can be accessed. And finally we've broken Wi-Fi encryption! An equivalent method was employed in Los Angeles ISSA meeting where some local FBI special agents cracked a WEP key of 128 bits in just 3 minutes with the usage of normally found tools available over the Internet. This issue is pointed out only to show that still WEP 128 is an exposed encryption and shouldn't be the one used to secure Wi-Fi hotspots. We should remember that the more computers produce interesting packets, the quicker we can crack the WEP.

IX. CONCLUSION

Wireless networks like Wi-Fi being the most spread technology over the world is vulnerable to the threats of Hacking. It is very important to protect a network from the hackers in order to prevent exploitation of confidential data. The better way to do this is, *just think like a hacker*. At a glance, we've talked about the whole process of cracking WEP encryption of Wi-Fi in this paper. All this is made only to figure out the necessity in getting touch with some of the scanning tools like NetStumbler, Cain, Kismet, MiniStumbler etc to survey the Wireless locality. The tools that have been stated will give us the ability to break our own WEP key and this may be the time to go to the next rank of security, the WPA. Let us try to hack all the standards of Wireless networks ethically in order to make a system very protected.

REFERENCES

[1] L. Zhou and Z.J. Haas, "Securing Ad Hoc Networks," *IEEE Network*, vol. 13, no. 6, 1999, pp. 24-30.

[2] Jinyang Li, Charles Blake, Douglas S. J. De Couto, Hu Imm Lee, Robert Morris, "Capacity of Ad Hoc Wireless Networks," In Proc. of Mobicom (mobicom01) conference, 2001.

[3] M. Junaid , Dr Muid Mufti, M.Umar Ilyas, "Vulnerabilities of IEEE 802.11i Wireless LAN," Transactions On Engineering, Computing And Technology V11 February 2006 Issn 1305-5313.

[4] Martin Beck, Erik Tews, "Practical attacks against WEP and WPA," November 8, 2008.

[5] US-CERT, "Using Wireless Technology Securely," produced by US-CERT, a government organization, 2008.

[6] Michael Roche ,"Wireless Hacking Tools," available at http://www.cse.wustl.edu/~jain/cse571-07/ftp/wireless_hacking/

[7] R. Teramura, Y. Asakura, T. Ohigashi, H. Kuwakado, and M. Morii, "Breaking WEP with Any 104-bit Keys - All WEP Keys Can Be Recovered Using IP Packets Only," Proc. of SCIS2009, CDROM, 1A2-6, Jan. 2009.

[8] Jeremy Martin, "The art of casual WiFi Hacking," CISSP-ISSAP, 2009.

[9] D. Waterman (Eds.), *"Interconnection and the internet"*: Selected papers from the 1996 TC conference.

[10] V. Moen, H. Raddum, and K.J. Hole, "Weaknesses in the temporal key hash of WPA," ACM SIGMOBILE Mobile Computing and Communications Review, vol.8, pp.76-83, 2004.

[11] Stanley, Richard A."Wireless LAN risks and vulnerabilities," Information systems control Journal, volume2, 2002.

[12] IEEE-SA Standards Board, "Wireless lan medium access control (MAC) and physical layer (PHY) specifications," Communications Magazine, IEEE, 2007.

Website Searches

[13] IEEE 802 standards, http://standards.ieee.org/getieee802

[14] IEEE 802.11 working group, http://grouper.ieee.org/groups/802/11/index.html

[15] Wireless Ethernet Compatibility Alliance, http://www.wirelessethernet.org/index.html

[16] WiFi -Windows, http://www.oxid.it (Cain & Able) http://www.NetStumbler.com

[17] WiFi - Linux, http://www.kismetwireless.net/ (Kismet) http://freshmeat.net/projects/aircrack(Aircrack) http://sourceforge.net/projects/airsnort (Airsnort) http://sourceforge.net/projects/airpwn (Airpwn) http://wepcrack.sourceforge.net (WEPCrack)

[18] IEEE 802.15 working group, http://grouper.ieee.org/groups/802/15/index.html

[19] IEEE 802 LAN/MAN Standards Committee, http://grouper.ieee.org/groups/802/index.html

Text Books

[20] Regina D Hartley, "Ethical Hacking: Teaching Students To Hack"

[21] Wiley Publications, "Introduction To Ethical Hacking," available at www.media.wiley.com