PRECLUSION OF WPS AND WPA ATTACKS USING ANOMALY DETECTION

S.Rohin[1], S.Lakshmi Narasimman[2], R.Prabhakar[3]
[1,2,3]IV Year B.E(CSE),Thiagarajar College of Engineering, Madurai
Email: rohin1385@gmail.com[1], ssriram78@yahoo.co.in[2] prabha11kar95@gmail.com[3]

Abstract

The Far-reaching usage of the standard called as "IEEE 802.11" has been acting as a solution to support dynamic network coverage with high bandwidth raised several security threats. The wide use of the Wi-Fi(Wireless Fidelity) has enabled us to easily access the internet and it has also paved way for the origin of many hacking attacks. So, in this paper we introduce a new approach for identifying and blacklisting an intruder in the discovery and authentication phase. An anomaly detection and prevention rules are constructed in SNORT(an intrusion prevention system) to prevent stealthy WPS and WPA related attacks. Our analysis focuses on a set of parameter variations caused by the incoming and the outgoing packets. The detection method uses monitoring the router from REAVER tool and the prevention rules relies on the incongruity pattern of the attacks. The results are generated and it is used to precisely pinpoint each intrusion attempts and attack patterns.

Keywords: WPS detection, Reaver snort, blacklisting, intruder identification

I. INTRODUCTION

Encryption such as WPA(Wireless protected access) and WPS (Wi-Fi protected security) play an vital role in securing the access points from malicious users. In fact, they have been prominent in securing privacy of every user. But these encryptions are usually cracked by stealthy dictionary attacks and hybrid dictionary attacks. The hybrid dictionary attack involves the capturing of the packet as a ".pcap" file and run it against an database containing trillion wordlist in it. The cracking process is hastened by specifying keywords to attack against the encryption. These keywords are usually obtained from the victim's social media information and location details.

Now, we know that these encryptions can be compromised by the attackers. So, Inorder to prevent such attacks from happening, a honeypot should be employed to monitor the router all times. This solution aims to blacklist the users who try to intrude with maximum login attempts and delay in the time interval of the logins. However, these solutions fail to identify the wrapper based attacks which involves spoofing of the TCP and IP packets. Route-injection attacks also causes a major threat by diverting the traffic to router stealing confidential data and session id leading to cross site scripting attacks. This methodology also deals with identifying the number of pin attempts in WPS and packet sniffing of IP packet in the network.

In our proposed methodology, we aim to identify and block an intruder at discovery and authentication level. First , we use the Reaver tool monitoring for analyzing the beacons, data ,transfer rate per second , packets lost and frames sent and we create a dataset . And we create another data set by simulating an attack in the router . The attacks which we use are WPS attack and stealthy dictionary attack and wrapper attacks. Then, we created the anomaly detection and prevention rules in snort, based on the above five parameters. These parameters have more efficiency in identifying the intruder. Also our methodology aims to reducing the router intrusion done by rogue hackers who uses long range Wi-Fi antennas to initiate the hack. As they use long range Wi-Fi antennas, the change in the incoming and outgoing of beacon frame pave way for blacklisting the attackers. Thus, this method differs entirely from the usual prevention method. Why this method is introduced ?. Inorder to reduce the major information espionage, a major number of routers which are under attack should be monitored continuously. These routers which underwent an successful attack led to the creation of zombie systems, which are used by hackers to commit major crimes. With the use of our methodology, we can create a solid preclusion for such attacks in the near future.

The paper is organized as follows,

In section 2, we do a literature survey of already existing intrusion prevention method that is used for the WPS and WPA vulnerability problem.

In section 3, we have included brief definitions about the concepts used in the paper.

In section 4, we explained our methodology and its application for preventing the router intrusion.

Section 5 consists of the experimental results and in section 6 we conclude our work.

II. LITERATURE REVIEW

A. Intrusion detection using multiscale traffic analysis

One effective Intrusion detection already introduced in [1] solves this problem by using ***real-time monitoring based on anomaly detection.*** This method proposed continuously monitors the router traffic and uses anomaly detection methodology to identify the pattern of WPS attacks. It also identifies the delay in pin attempts in WPS and recurring delay in authentication. With the help of these patterns, it was able to identify the malicious users. In this paper, we have introduced a new Rule-Based Monitoring with support of anomaly detection. Our aim is to improve the efficiency of the router intrusion.

II. BASIC CONCEPTS

B. Beacon Frame

Beacon Frame acts as an vital part in discovery phase of the access points. Beacon frames consist of an Ethernet header, body and FCS. Beacon frames had been transmitted by the Access Point from the router in an infrastructure Basic service set. In IBSS network beacon generation is distributed among the stations.

The fields of the beacon frame includes timestamp, beacon interval, capability information, SSID, Frequency hopping Parameter set(FH),Direct Sequence Parameter Set(DS),Contention-Free Parameter Set(CF) ,IBSS parameter set, and Traffic Indication Map(TIM).

Beacons have an defined interval of 100ms. An they are usually sent with an CSMA/CA algorithm. However, stations are able to compensate for this difference by inspecting the timestamp in the beacon frame when it has been finally sent.

C. SNORT (IDS/IPS)

Snort is an Intrusion Detection System(IDS) also called as an Burglar alarm of computer networks and is an important part of network perimeter security. Without an IDS, it is impossible to keep track of who entering our entering our network with malicious intention. There are two types of IDS’s Host based and Network based .

Snort is an Network based IDS which can be used to monitor the network traffic for any suspicious activity.. The program can also be used to detect probes or attacks, including, but not limited to, operating system fingerprinting attempts, common gateway interface, buffer overflows, server message block probes, and stealth port scans.

Snort can be configured in three main modes: sniffer, packet logger, and network intrusion detection. In sniffer mode, the program will read network packets and display them on the console. In packet logger mode, the program will log packets to the disk. In intrusion detection mode, the program will monitor network traffic and analyze it against a rule set defined by the user. The program will then perform a specific action based on what has been identified. The rules and attack pattern signatures can be feed into snort to avoid the dangerous possible attacks. Snort works on rules which can be user-defined.

D. REAVER

A Reaver is an robust and practical attack tool that has been included in kali linux(OS). This tool can be used to brute force the WPS pin and it can also be used to monitor the router for its data transmission of the incoming and the outgoing packets. Reaver has been tested against an wide variety of access points and has been used to exploit the WPS vulnerability many times in the past.

E. Anomaly Based Intrusion Detection System

The Anomaly Based Intrusion Detection System is useful for identifying both network and the computer intrusions. It can be used to check the real time network traffic as whether it is normal or anomalous. The classification done by this is usually based on Heuristics or Rules ,rather than patterns or signatures because the patterns and the signatures can be tamper and manipulated. It has been used in the applications of neural networks for greater use.

IV.METHODOLOGY DESCRIPTION

Algorithm And Problem Solving Approach

- Capture the packets between the router and the attacker machine.
- Analyze the logs with graphic model and create rules for detecting attack using SNORT for detection.
- For prevention, create additional rules for Blacklisting the IP based on delay in login attempts and beacon count and change in channels.
- After blacklisting the IP, the attack can be prevented.

This can be implemented in an virtual environment to further understand how NAT (Network Address Translation) and Port Address Translation (PAT) behaves based on the Network attacks in the router.

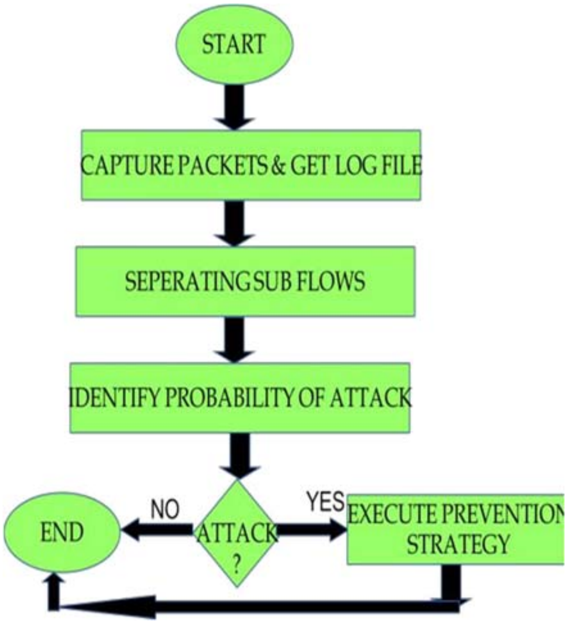

START
CAPTURE PACKETS & GET LOG FILE
SEPERATING SUB FLOWS
IDENTIFY PROBABILITY OF ATTACK
ATTACK ?
NO
YES
END
EXECUTE PREVENTION STRATEGY

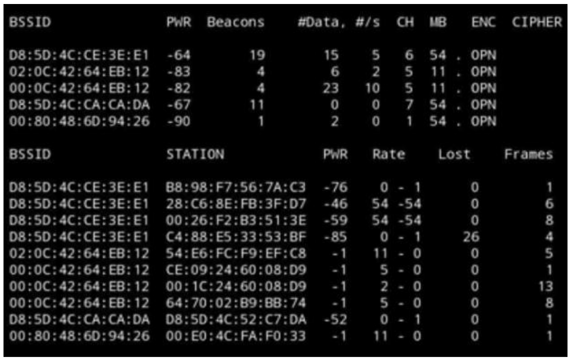
Fig1.This figure represents the Working of our mechanism

Separating Sub-Flows

As mentioned in the above flow chart representation, the separating sub flows involves the creation of rules in the snort, that can be used to identify and prevent the attacks. The step-by-step procedure of the methodology is explained below.

Data Capturing Methodology

The data capturing has been done in an virtual environment but the actual data is took from a real router. In the Virtual machine, the Reaver tool is turned on in the monitoring mode and an external usb wi-fi adapter is connected on the system to monitor the router from the virtual machine. The Reaver mode would look like this.

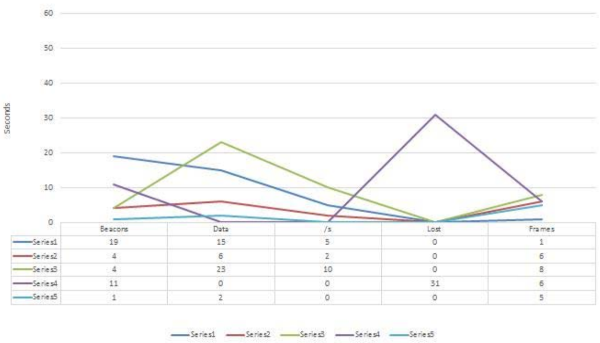


```
BSSID              PWR  Beacons    #Data, #/s  CH  MB   ENC  CIPHER

D8:5D:4C:CE:3E:E1  -64       19       15    5   6  54 . OPN
02:0C:42:64:EB:12  -83        4        6    2   5  11 . OPN
00:0C:42:64:EB:12  -82        4       23   10   5  11 . OPN
D8:5D:4C:CA:CA:DA  -67       11        0    0   7  54 . OPN
00:80:48:6D:94:26  -90        1        2    0   1  54 . OPN

BSSID              STATION            PWR   Rate    Lost    Frames

D8:5D:4C:CE:3E:E1  B8:98:F7:56:7A:C3  -76    0 - 1      0        1
D8:5D:4C:CE:3E:E1  28:C6:8E:FB:3F:D7  -46   54 -54      0        6
D8:5D:4C:CE:3E:E1  00:26:F2:B3:51:3E  -59   54 -54      0        8
D8:5D:4C:CE:3E:E1  C4:88:E5:33:53:BF  -85    0 - 1     26        4
02:0C:42:64:EB:12  54:E6:FC:F9:EF:C8   -1   11 - 0      0        5
00:0C:42:64:EB:12  CE:09:24:60:08:D9   -1    5 - 0      0        1
00:0C:42:64:EB:12  00:1C:24:60:08:D9   -1    2 - 0      0       13
00:0C:42:64:EB:12  64:70:02:B9:BB:74   -1    5 - 0      0        8
D8:5D:4C:CA:CA:DA  D8:5D:4C:52:C7:DA  -52    0 - 1      0        1
00:80:48:6D:94:26  00:E0:4C:FA:F0:33   -1   11 - 0      0        1
```

Comparative Analysis of parameter change Before and After attack With Graphical Model

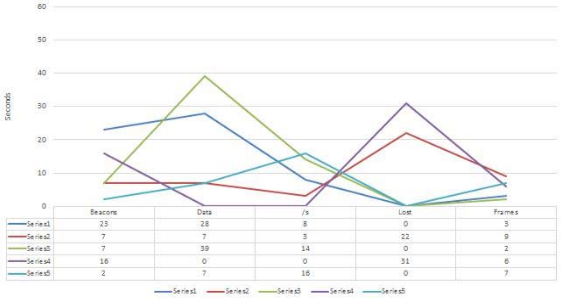
This Comparative Analysis clearly shows the discovery phase changes in the network. The

Below graphical model is created based on the WPA attacks



	Beacons	Data	/s	Lost	Frames
Series1	19	15	5	0	1
Series2	4	6	2	0	6
Series3	4	23	10	0	8
Series4	11	0	0	31	6
Series5	1	2	0	0	5

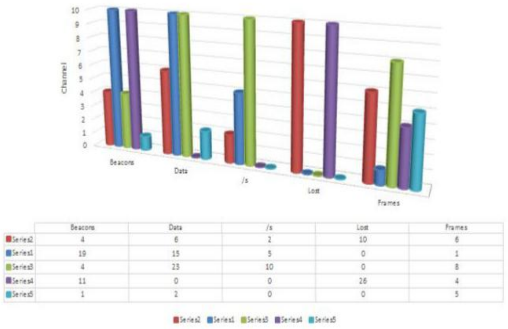
Fig2. This figure represents the idle network with respect to time



	Beacons	Data	/s	Lost	Frames
Series1	23	28	8	0	3
Series2	7	7	3	22	9
Series3	7	39	14	0	2
Series4	16	0	0	31	6
Series5	2	7	16	0	7

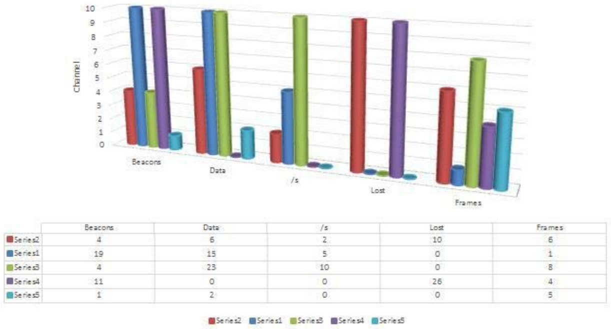
Fig3.This Figure represents the Network under attack with respect to time

There are some additional experimental results we collected which is based on the channel variation, in which the attacker changes channel frequently to initiate the attack on the router.



	Beacons	Data	/s	Lost	Frames
Series2	4	6	2	10	6
Series1	19	15	5	0	1
Series3	4	23	10	0	8
Series4	11	0	0	26	4
Series5	1	2	0	0	5

Fig4. It represents idle Network w.r.t channel.

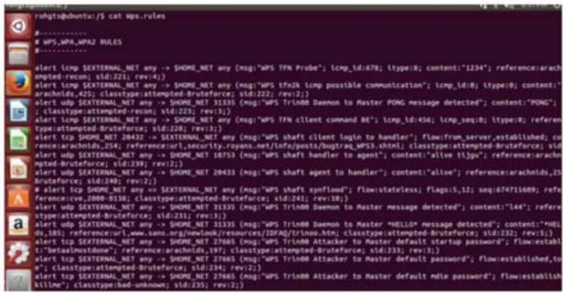


	Beacons	Data	/s	Lost	Frames
Series2	4	6	2	10	6
Series1	19	15	5	0	1
Series3	4	23	10	0	8
Series4	11	0	0	26	4
Series5	1	2	0	0	5

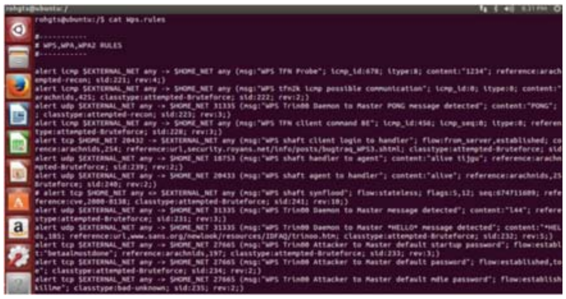
Fig 5.It represents the Network Under attack w.r.to Channel

Rule Creation Based on the Parameter Analysis

Based on the above analysis several rules were created to prevent the WPS attacks. The following rules were created to detect the WPS attacks. This can be seen in the following representation.



```
rohgts@ubuntu:/$ cat wps.rules

#-----------
# WPS,WPA,WPA2 RULES
#-----------

alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"WPS TFN Probe"; icmp_id:678; itype:8; content:"1234"; reference:arach
empted-recon; sid:221; rev:4;)
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"WPS tfn2k icmp possible communication"; icmp_id:0; itype:0; content:"
arachnids,425; classtype:attempted-Bruteforce; sid:222; rev:2;)
alert udp $EXTERNAL_NET any -> $HOME_NET 31335 (msg:"WPS Trin00 Daemon to Master PONG message detected"; content:"PONG";
; classtype:attempted-recon; sid:223; rev:3;)
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"WPS TFN client command BE"; icmp_id:456; icmp_seq:0; itype:0; referen
type:attempted-Bruteforce; sid:228; rev:3;)
alert tcp $HOME_NET 20432 -> $EXTERNAL_NET any (msg:"WPS shaft client login to handler"; flow:from_server,established; co
rence:arachnids,254; reference:url,security.royans.net/info/posts/bugtraq_WPS3.shtml; classtype:attempted-Bruteforce; sid
alert udp $EXTERNAL_NET any -> $HOME_NET 18753 (msg:"WPS shaft handler to agent"; content:"alive tijgu"; reference:arachn
mpted-Bruteforce; sid:239; rev:2;)
alert udp $EXTERNAL_NET any -> $HOME_NET 20433 (msg:"WPS shaft agent to handler"; content:"alive"; reference:arachnids,25
Bruteforce; sid:240; rev:2;)
# alert tcp $HOME_NET any <> $EXTERNAL_NET any (msg:"WPS shaft synflood"; flow:stateless; flags:S,12; seq:674711609; refe
ference:cve,2000-0138; classtype:attempted-Bruteforce; sid:241; rev:10;)
alert udp $EXTERNAL_NET any -> $HOME_NET 31335 (msg:"WPS Trin00 Daemon to Master message detected"; content:"l44"; refere
stype:attempted-Bruteforce; sid:231; rev:3;)
alert udp $EXTERNAL_NET any -> $HOME_NET 31335 (msg:"WPS Trin00 Daemon to Master *HELLO* message detected"; content:"*HEL
ds,185; reference:url,www.sans.org/newlook/resources/IDFAQ/trinoo.htm; classtype:attempted-Bruteforce; sid:232; rev:5;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 27665 (msg:"WPS Trin00 Attacker to Master default startup password"; flow:establ
t:"betaalmostdone"; reference:arachnids,197; classtype:attempted-Bruteforce; sid:233; rev:3;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 27665 (msg:"WPS Trin00 Attacker to Master default password"; flow:established,to
e"; classtype:attempted-Bruteforce; sid:234; rev:2;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 27665 (msg:"WPS Trin00 Attacker to Master default mdie password"; flow:establish
killme"; classtype:bad-unknown; sid:235; rev:2;)
```

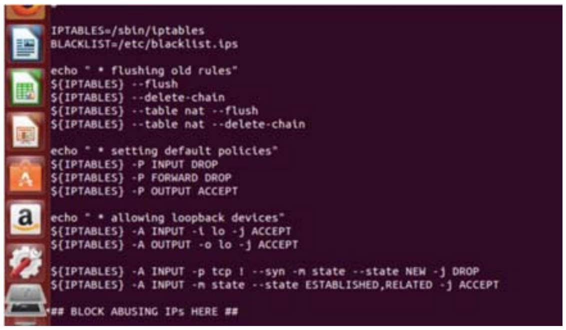


```
rohgts@ubuntu:/$ cat wps.rules

#-----------
# WPS,WPA,WPA2 RULES
#-----------

alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"WPS TFN Probe"; icmp_id:678; itype:8; content:"1234"; reference:arach
empted-recon; sid:221; rev:4;)
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"WPS tfn2k icmp possible communication"; icmp_id:0; itype:0; content:"
arachnids,425; classtype:attempted-Bruteforce; sid:222; rev:2;)
alert udp $EXTERNAL_NET any -> $HOME_NET 31335 (msg:"WPS Trin00 Daemon to Master PONG message detected"; content:"PONG";
; classtype:attempted-recon; sid:223; rev:3;)
alert icmp $EXTERNAL_NET any -> $HOME_NET any (msg:"WPS TFN client command BE"; icmp_id:456; icmp_seq:0; itype:0; referen
type:attempted-Bruteforce; sid:228; rev:3;)
alert tcp $HOME_NET 20432 -> $EXTERNAL_NET any (msg:"WPS shaft client login to handler"; flow:from_server,established; co
rence:arachnids,254; reference:url,security.royans.net/info/posts/bugtraq_WPS3.shtml; classtype:attempted-Bruteforce; sid
alert udp $EXTERNAL_NET any -> $HOME_NET 18753 (msg:"WPS shaft handler to agent"; content:"alive tijgu"; reference:arachn
mpted-Bruteforce; sid:239; rev:2;)
alert udp $EXTERNAL_NET any -> $HOME_NET 20433 (msg:"WPS shaft agent to handler"; content:"alive"; reference:arachnids,25
Bruteforce; sid:240; rev:2;)
# alert tcp $HOME_NET any <> $EXTERNAL_NET any (msg:"WPS shaft synflood"; flow:stateless; flags:S,12; seq:674711609; refe
ference:cve,2000-0138; classtype:attempted-Bruteforce; sid:241; rev:10;)
alert udp $EXTERNAL_NET any -> $HOME_NET 31335 (msg:"WPS Trin00 Daemon to Master message detected"; content:"l44"; refere
stype:attempted-Bruteforce; sid:231; rev:3;)
alert udp $EXTERNAL_NET any -> $HOME_NET 31335 (msg:"WPS Trin00 Daemon to Master *HELLO* message detected"; content:"*HEL
ds,185; reference:url,www.sans.org/newlook/resources/IDFAQ/trinoo.htm; classtype:attempted-Bruteforce; sid:232; rev:5;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 27665 (msg:"WPS Trin00 Attacker to Master default startup password"; flow:establ
t:"betaalmostdone"; reference:arachnids,197; classtype:attempted-Bruteforce; sid:233; rev:3;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 27665 (msg:"WPS Trin00 Attacker to Master default password"; flow:established,to
e"; classtype:attempted-Bruteforce; sid:234; rev:2;)
alert tcp $EXTERNAL_NET any -> $HOME_NET 27665 (msg:"WPS Trin00 Attacker to Master default mdie password"; flow:establish
killme"; classtype:bad-unknown; sid:235; rev:2;)
```

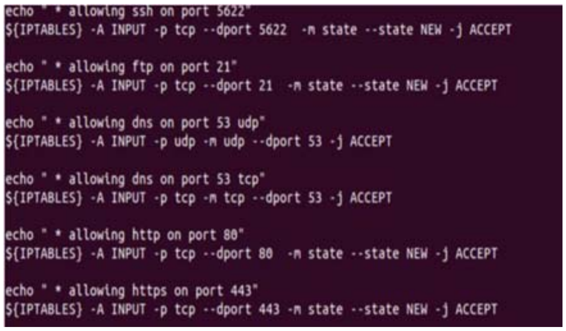
The above two images clearly explains the rules created based on the comparative analysis. The comparative analysis is based on both time and channel, but the time series plays a vital role in pinpointing the users attempt to attack the router or the network.

Blacklisting the Identified Users

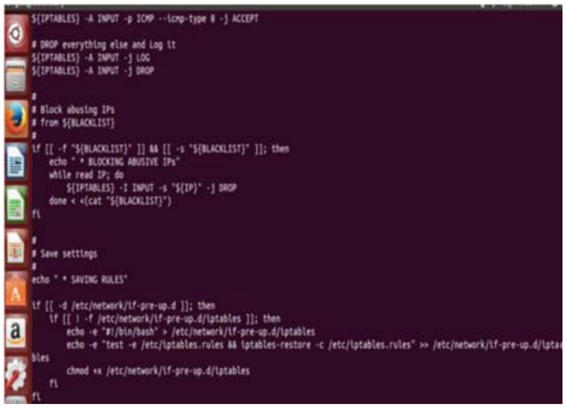
We have created some additional rules to support the snort to blacklist an user from the network. These rules just check for the malicious users and fetch their IP from the IP tables and flush their IP’s to the firewall rules through which they can be blacklisted from the network. The rules can be shown as follows:



```
IPTABLES=/sbin/iptables
BLACKLIST=/etc/blacklist.ips

echo " * flushing old rules"
${IPTABLES} --flush
${IPTABLES} --delete-chain
${IPTABLES} --table nat --flush
${IPTABLES} --table nat --delete-chain

echo " * setting default policies"
${IPTABLES} -P INPUT DROP
${IPTABLES} -P FORWARD DROP
${IPTABLES} -P OUTPUT ACCEPT

echo " * allowing loopback devices"
${IPTABLES} -A INPUT -i lo -j ACCEPT
${IPTABLES} -A OUTPUT -o lo -j ACCEPT

${IPTABLES} -A INPUT -p tcp ! --syn -m state --state NEW -j DROP
${IPTABLES} -A INPUT -m state --state ESTABLISHED,RELATED -j ACCEPT

## BLOCK ABUSING IPs HERE ##
```



```
echo " * allowing ssh on port 5622"
${IPTABLES} -A INPUT -p tcp --dport 5622  -m state --state NEW -j ACCEPT

echo " * allowing ftp on port 21"
${IPTABLES} -A INPUT -p tcp --dport 21  -m state --state NEW -j ACCEPT

echo " * allowing dns on port 53 udp"
${IPTABLES} -A INPUT -p udp -m udp --dport 53 -j ACCEPT

echo " * allowing dns on port 53 tcp"
${IPTABLES} -A INPUT -p tcp -m tcp --dport 53 -j ACCEPT

echo " * allowing http on port 80"
${IPTABLES} -A INPUT -p tcp --dport 80  -m state --state NEW -j ACCEPT

echo " * allowing https on port 443"
${IPTABLES} -A INPUT -p tcp --dport 443 -m state --state NEW -j ACCEPT
```



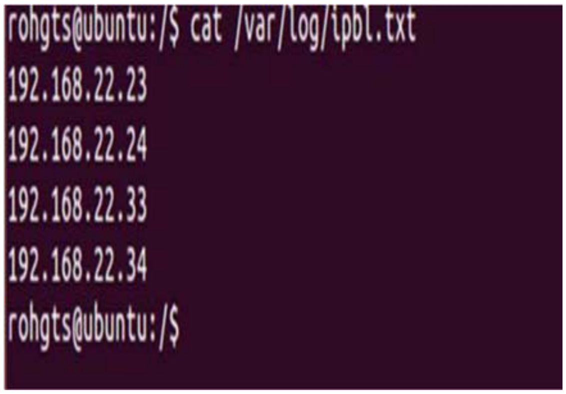
```
${IPTABLES} -A INPUT -p ICMP --icmp-type 8 -j ACCEPT

# DROP everything else and Log it
${IPTABLES} -A INPUT -j LOG
${IPTABLES} -A INPUT -j DROP

#
# Block abusing IPs
# from ${BLACKLIST}
#
if [[ -f "${BLACKLIST}" ]] && [[ -s "${BLACKLIST}" ]]; then
    echo " * BLOCKING ABUSIVE IPs"
    while read IP; do
        ${IPTABLES} -I INPUT -s "${IP}" -j DROP
    done < <(cat "${BLACKLIST}")
fi

#
# Save settings
#
echo " * SAVING RULES"

if [[ -d /etc/network/if-pre-up.d ]]; then
    if [[ ! -f /etc/network/if-pre-up.d/iptables ]]; then
        echo -e "#!/bin/bash" > /etc/network/if-pre-up.d/iptables
        echo -e "test -e /etc/iptables.rules && iptables-restore -c /etc/iptables.rules" >> /etc/network/if-pre-up.d/ipta
bles
        chmod +x /etc/network/if-pre-up.d/iptables
    fi
fi
```

These Rules run in background to enhance the efficiency of snort.

V. EXPERIMENTS & RESULTS

The attack simulation was performed on ***a ROUTER with WPS-Enabled and Encryption set to WPA***. This experiment was done in an non-IP spoofed environment. We have conducted a series of different hacking attacks such as stealthy WPS attack with delay of pin tries through Reaver tool , hybrid dictionary attack with wordlist updated into the database, and we also did normal hydra brute force attack.

For our experiment, we created specific rules to blacklist the IP’s from the analysis done on the changes in parameters. The results can be found in the log file, as we see the blocked IP’s :



```
rohgts@ubuntu:/$ cat /var/log/ipbl.txt
192.168.22.23
192.168.22.24
192.168.22.33
192.168.22.34
rohgts@ubuntu:/$
```

From the above result, our mechanism successfully blocked the malicious users from the network.

VI. CONCLUSION

Intrusion Preclusion is one of the many greatest challenges in the field of network Security. While problems like brute forcing of WPS pins and malicious activity in the network are being solved and the other type of attacks like stealthy dictionary attacks and hybrid attacks are doing their damages as we speak. Intrusion preclusion using signatures and anomaly patterns are the emerging defense mechanisms to prevent these types of attacks. There is never a permanent solution to any problem but with further research in intrusion preclusion , it will be possible to build a secure network environment.

Future works on this paper include using this same technique in IoT(internet of things) to prevent the WPS attacks between the Wi-Fi and its sensors. Further research in this area will bring great prospects for the blooming of IoT security. Another usage of this mechanism can be used to prevent the hacking attacks through ToR network .

ACKNOWLEDGMENT

The idea for this paper was guided and persuaded by Dr. K.Sundara Kantham. The work was supported by Dept. of Computer Science and Engineering, Thiagarajar College of Engineering, Madurai.

REFERENCES

[1] “Using Multiscale Traffic Analysis to Detect WPS”, presented by Ivo Petiz, Eduardo Rocha, Paulo Salvador, Ant´onio Nogueira at “**IEEE International Conference on Communications-2013**”

[2] (2012, September) Reaver WPS - Brute force attack against Wi-Fi protected setup. [Online].
Available:http://code.google.com/p/ReaverWPS/

[3] A. Bittau, M. Handley, and J. Lackey, "The final nail in WEP's coffin," in IEEE Symposium on Security and Privacy, may 2006, pp. 15–400.

[4] (2011, March) Snort home page. [Online]. Available: http://www.snort.org/

[5] "Wi-fi protected setup white paper," Tech. Rep., January 2007. [Online]. Available: https://www.wi-fi.org/knowledge-center/white-papers

[6] T. Shon and J. Moon, "A hybrid machine learning approach to network anomaly detection," Information Sciences, vol. 177, no. 18, pp. 3799–3821, September 2007.