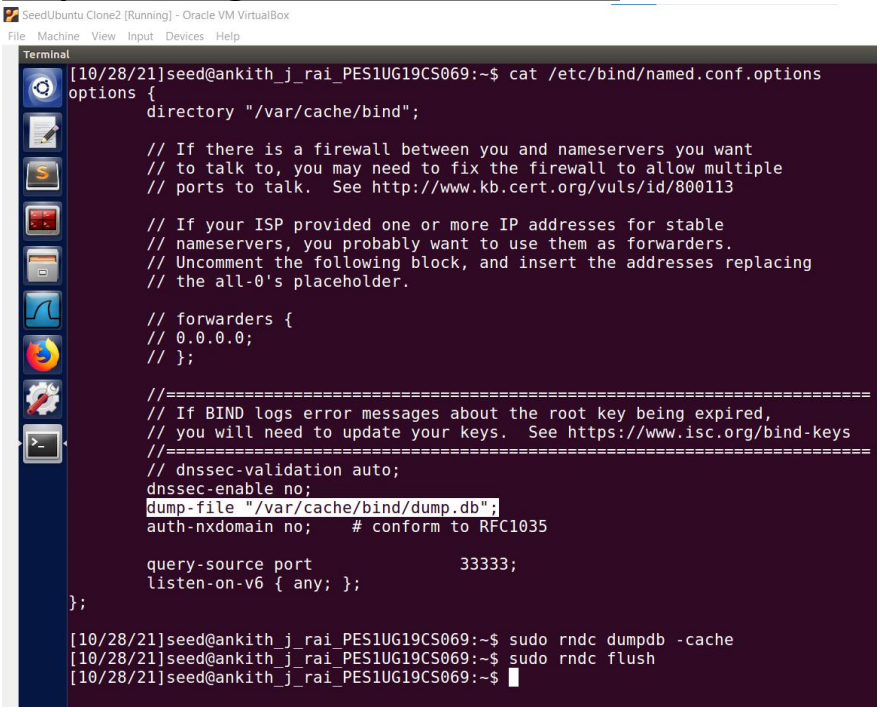
Remote DNS cache Poisoning Attack Lab

Name : Ankith J Rai
SRN : PES1UG19CS069
SEC : B

Machine	IP address
Attacker	10.0.2.5
Victim	10.0.2.10
DNS server	10.0.2.11

Step 1: Configure the BIND9 Server



From the above screenshots we get to know that the cache content is dumped at /var/cache/bind/dump.db if bind is asked to dump it's cache.

Step 2: Turnoff DNSSEC

From the below screenshot we can see that the spoofing attack protection has been removed from the DNS server.

```
SeedUbuntu Clone2 [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help
Terminal
  GNU nano 2.5.3                    File: /etc/bind/named.conf.options

options {
        directory "/var/cache/bind";

        // If there is a firewall between you and nameservers you want
        // to talk to, you may need to fix the firewall to allow multiple
        // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

        // If your ISP provided one or more IP addresses for stable
        // nameservers, you probably want to use them as forwarders.
        // Uncomment the following block, and insert the addresses replacing
        // the all-0's placeholder.

        // forwarders {
        // 0.0.0.0;
        // };

        //========================================================================
        // If BIND logs error messages about the root key being expired,
        // you will need to update your keys.  See https://www.isc.org/bind-keys
        //========================================================================
        # dnssec-validation auto;
        dnssec-enable no;
        dump-file "/var/cache/bind/dump.db";
        auth-nxdomain no;    # conform to RFC1035

        query-source port               33333;
        listen-on-v6 { any; };
};
```

Step 3: Fix the Source Ports

```
SeedUbuntu Clone2 [Running] - Oracle VM VirtualBox
File  Machine  View  Input  Devices  Help
Terminal
[10/28/21]seed@ankith_j_rai_PES1UG19CS069:~$ cat /etc/bind/named.conf.options
options {
        directory "/var/cache/bind";

        // If there is a firewall between you and nameservers you want
        // to talk to, you may need to fix the firewall to allow multiple
        // ports to talk.  See http://www.kb.cert.org/vuls/id/800113

        // If your ISP provided one or more IP addresses for stable
        // nameservers, you probably want to use them as forwarders.
        // Uncomment the following block, and insert the addresses replacing
        // the all-0's placeholder.

        // forwarders {
        // 0.0.0.0;
        // };

        //========================================================================
        // If BIND logs error messages about the root key being expired,
        // you will need to update your keys.  See https://www.isc.org/bind-keys
        //========================================================================
        # dnssec-validation auto;
        dnssec-enable no;
        dump-file "/var/cache/bind/dump.db";
        auth-nxdomain no;    # conform to RFC1035

        query-source port               33333;
        listen-on-v6 { any; };
};

[10/28/21]seed@ankith_j_rai_PES1UG19CS069:~$
```

We can see from the above screenshot that the query source port number has been fixed to 33333.

Step 4: Remove the example.com zone

```
[10/28/21]seed@ankith_j_rai_PES1UG19CS069:~$ cat /etc/bind/named.conf
// This is the primary configuration file for the BIND DNS server named.
//
// Please read /usr/share/doc/bind9/README.Debian.gz for information on the
// structure of BIND configuration files in Debian, *BEFORE* you customize
// this configuration file.
//
// If you are just adding zones, please do that in /etc/bind/named.conf.local

include "/etc/bind/named.conf.options";
include "/etc/bind/named.conf.local";
include "/etc/bind/named.conf.default-zones";
[10/28/21]seed@ankith_j_rai_PES1UG19CS069:~$
```

As we can see we have removed the example.com Zone from /etc/bind/named.conf .

Step 5: Start DNS server

```
[10/28/21]seed@ankith_j_rai_PES1UG19CS069:~$ sudo service bind9 restart
[10/28/21]seed@ankith_j_rai_PES1UG19CS069:~$
```

Now we have restarted the DNS server.

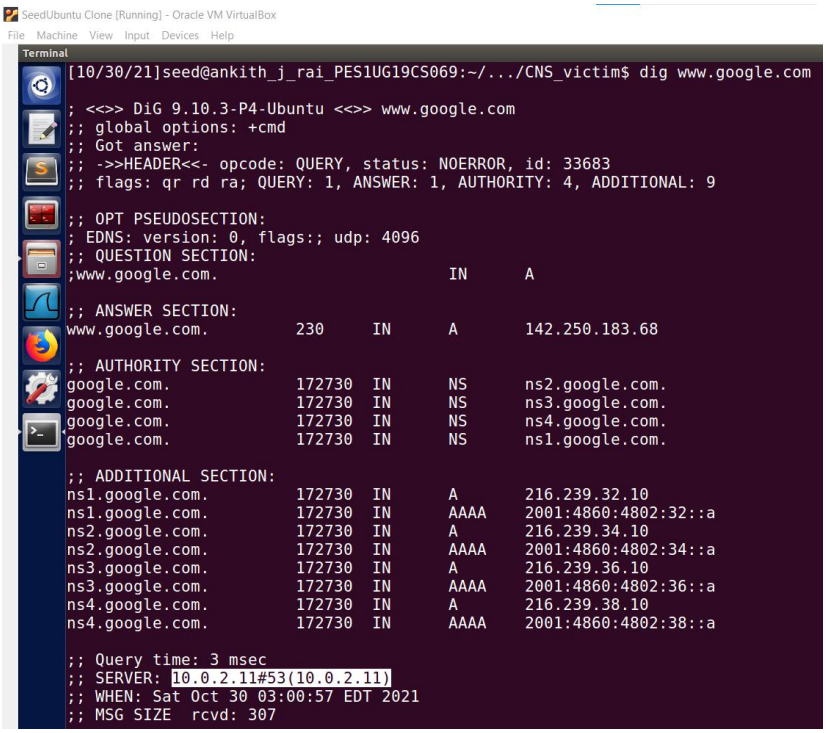
Task 2: Configure the Victim and Attacker Machine

```
[10/30/21]seed@ankith_j_rai_PES1UG19CS069:~/.../CNS_victim$ sudo nano /etc/resolvconf/resolv.conf.d/head
[10/30/21]seed@ankith_j_rai_PES1UG19CS069:~/.../CNS_victim$ cat /etc/resolvconf/resolv.conf.d/head
# Dynamic resolv.conf(5) file for glibc resolver(3) generated by resolvconf(8)
#     DO NOT EDIT THIS FILE BY HAND -- YOUR CHANGES WILL BE OVERWRITTEN
nameserver 10.0.2.11

[10/30/21]seed@ankith_j_rai_PES1UG19CS069:~/.../CNS_victim$ sudo resolvconf -u
[10/30/21]seed@ankith_j_rai_PES1UG19CS069:~/.../CNS_victim$
```

The ip address of the DNS server is 10.0.2.11 . This ip address has been added to the /etc/resolvconf/resolv.conf.d/head of the victim machine as it's local DNS server.

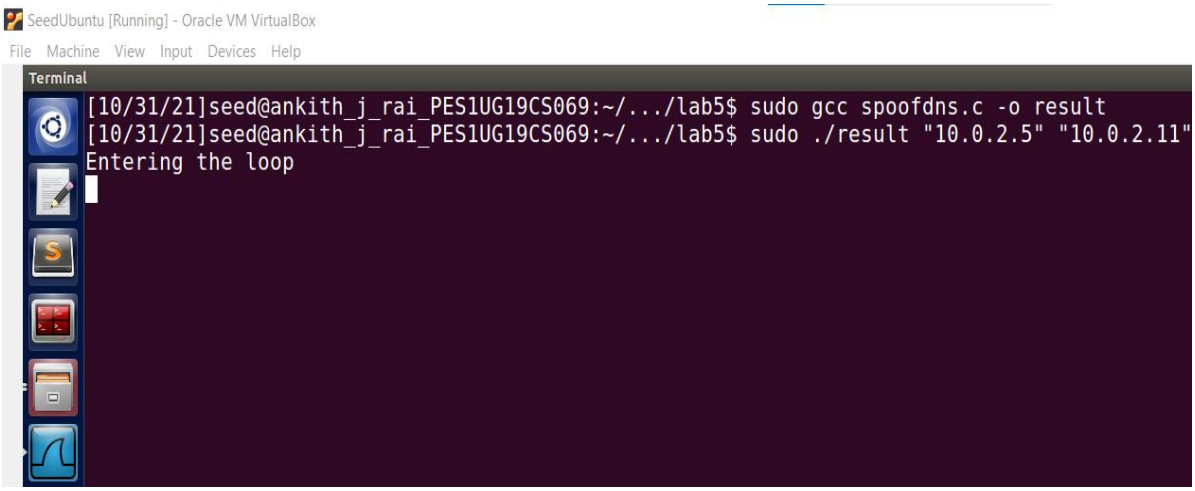
The sudo resolvconf -u has been used so that the change takes effect.



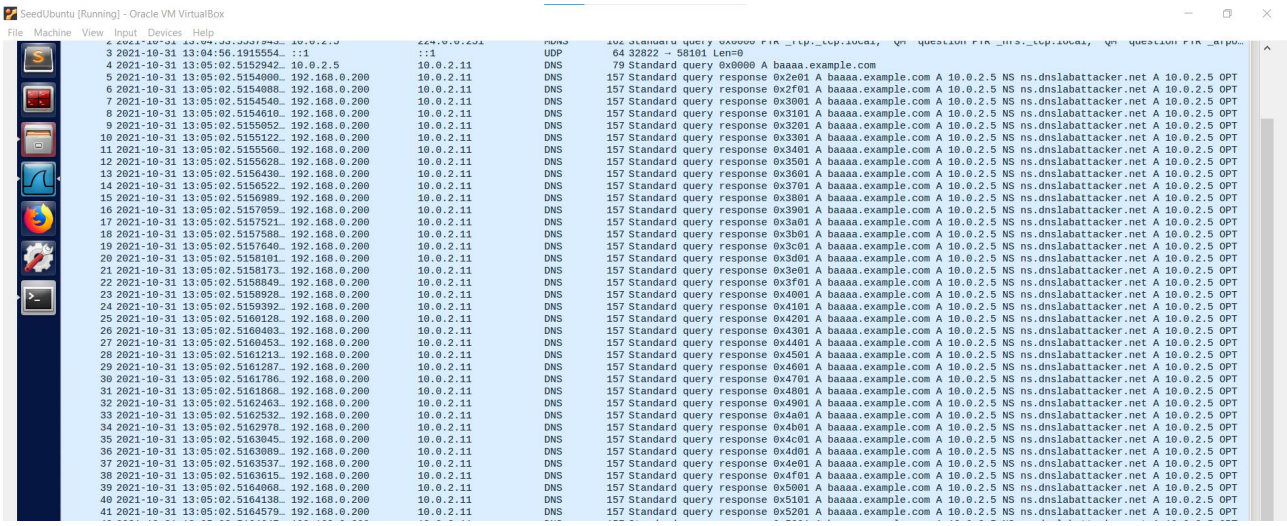
From the above screenshot we can see that on using dig www.google.com we can see that the response is coming from 10.0.2.11 . Hence our setup is successful.

Tasks 3.1 The Kaminsky attack:

Attacker terminal screenshot:

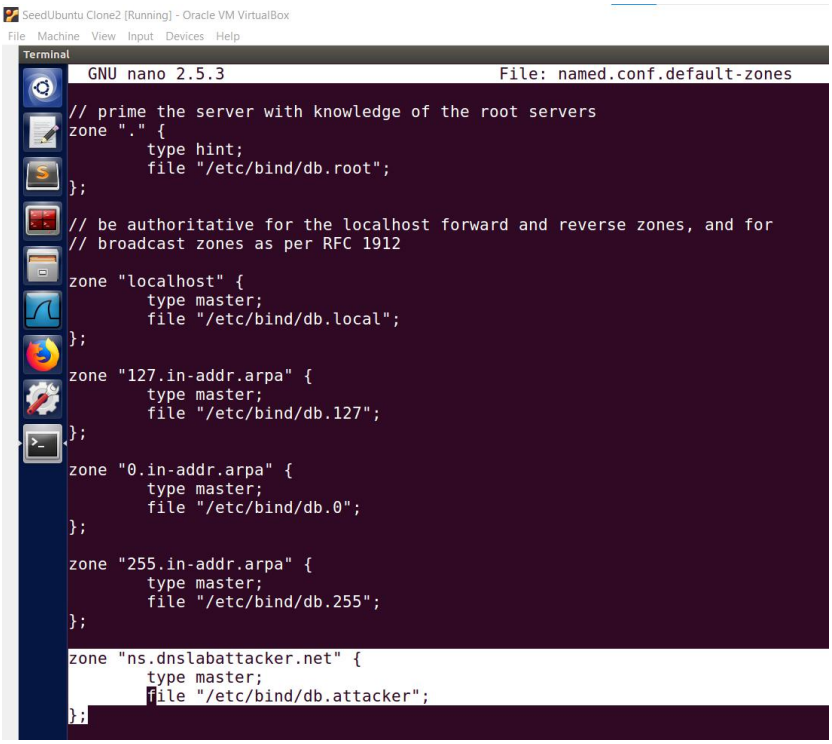


Screenshot of wireshark:



From the above screenshot we can see that DNS response is being sent from 192.168.0.200(remote machine) to 10.0.2.11 . When the response whose transaction id is same as the request query's transaction id,it poisons the DNS server.

RESULT VERIFICATION:



Creating db.attacker on DNS server

```
GNU nano 2.5.3                    File: db.attacker                    Modified

$TTL 604800
@ IN SOA localhost. root.localhost. (
                2; Serial
                604800 ; Refresh
                86400 ; Retry
                2419200 ; Expire
                604800 ) ; Negative Cache TTL;
@ IN NS ns.dnslabattacker.net.
@ IN A 192.168.0.200
@ IN AAAA ::1
```

Creating the example.com zone:

```
[10/31/21]seed@ankith_j_rai_PES1UG19CS069:.../bind$ sudo nano named.conf.local
[10/31/21]seed@ankith_j_rai_PES1UG19CS069:.../bind$ cat named.conf.local
//
// Do any local configuration here
//

// Consider adding the 1918 zones here, if they are not used in your
// organization
//include "/etc/bind/zones.rfc1918";
zone "example.com" {
                type master;
                file "/etc/bind/example.com.db";
};
[10/31/21]seed@ankith_j_rai_PES1UG19CS069:.../bind$
```

Creating the example.com.db file:

```
[10/31/21]seed@ankith_j_rai_PES1UG19CS069:.../bind$ cat example.com.db
$TTL 3D
@                       IN              SOA ns.example.com. admin.example.com. (
                        2008111001
                        8H
                        2H
                        4W
                        1D)
@                       IN              NS              ns.dnslabattacker.net.
@                       IN              MX              10 mail.example.com.
www                     IN              A               1.1.1.1
mail                    IN              A               1.1.1.2
*.example.com.          IN              A               1.1.1.100
[10/31/21]seed@ankith_j_rai_PES1UG19CS069:.../bind$
```

Now diging www.example.com from user machine.

```
[10/31/21]seed@ankith_j_rai_PES1UG19CS069:~/.../CNS_victim$ dig www.example.com

; <<>> DiG 9.10.3-P4-Ubuntu <<>> www.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 4036
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 3

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

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

;; AUTHORITY SECTION:
example.com.            259200  IN      NS      ns.dnslabattacker.net.

;; ADDITIONAL SECTION:
ns.dnslabattacker.net.  604800  IN      A       192.168.0.200
ns.dnslabattacker.net.  604800  IN      AAAA    ::1

;; Query time: 0 msec
;; SERVER: 10.0.2.11#53(10.0.2.11)
;; WHEN: Sun Oct 31 13:27:29 EDT 2021
;; MSG SIZE  rcvd: 139

[10/31/21]seed@ankith_j_rai_PES1UG19CS069:~/.../CNS_victim$
```

From the above screenshot we can see that the ip address of www.example.com now is 1.1.1.1 . This is because the DNS server cache has been poisoned by the remote machine attack.

```
[10/31/21]seed@ankith_j_rai_PES1UG19CS069:~/.../CNS_victim$ dig abcd.example.com

; <<>> DiG 9.10.3-P4-Ubuntu <<>> abcd.example.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 3252
;; flags: qr aa rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 1, ADDITIONAL: 3

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

;; ANSWER SECTION:
abcd.example.com.       259200  IN      A       1.1.1.100

;; AUTHORITY SECTION:
example.com.            259200  IN      NS      ns.dnslabattacker.net.

;; ADDITIONAL SECTION:
ns.dnslabattacker.net.  604800  IN      A       192.168.0.200
ns.dnslabattacker.net.  604800  IN      AAAA    ::1

;; Query time: 1 msec
;; SERVER: 10.0.2.11#53(10.0.2.11)
;; WHEN: Sun Oct 31 13:28:09 EDT 2021
;; MSG SIZE  rcvd: 140

[10/31/21]seed@ankith_j_rai_PES1UG19CS069:~/.../CNS_victim$
```

From the above screenshot we can see that on diging abcd.example.com , we get a response saying that the ip address of abcd.example.com is 1.1.1.100 . This is because the DNS server cache has been poisoned by the remote machine attack.