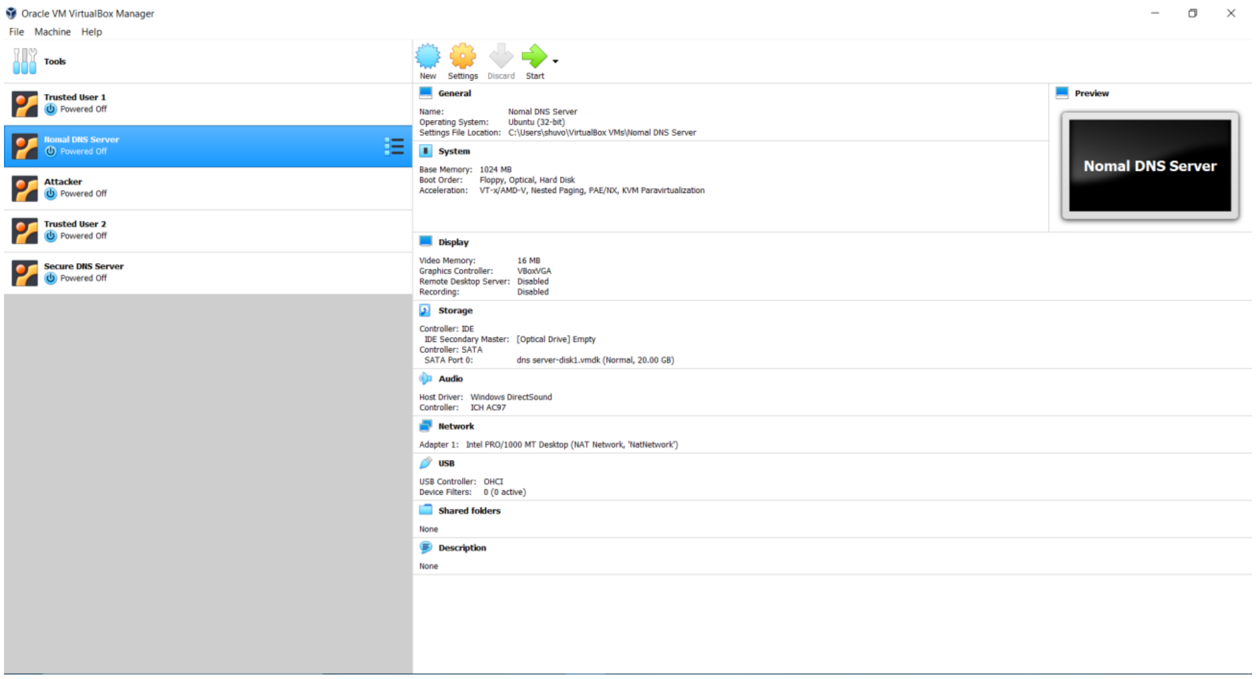
DoS Attack to the DNS Server (Using Spoofed IP Address)

Final Report & Implementation Demo

Source Code: https://github.com/shiroe41/DoS-Attack-to-the-DNS-Server.git

Attack Strategy: We will execute the DoS attack with **DNS NXDOMAIN Flood attack** approach. With this approach, the DNS server is flooded with non-existent domain-name requests which makes the DNS cache overflow, and the server uses up all its resources to query these domain-names and thus becomes unable to answer DNS query to valid clients. We will also spoof the IP address of the attacker so that the attacker cannot be traced.

We will be using Oracle VM VirtualBox to simulate this attack and some prevention measures.



Trusted User 1 — Powered Off
Nomal DNS Server — Powered Off
Attacker — Powered Off
Trusted User 2 — Powered Off
Secure DNS Server — Powered Off

General
```
Name:                    Nomal DNS Server
Operating System:        Ubuntu (32-bit)
Settings File Location:  C:\Users\shuvo\VirtualBox VMs\Nomal DNS Server
```
System
```
Base Memory:   1024 MB
Boot Order:    Floppy, Optical, Hard Disk
Acceleration:  VT-x/AMD-V, Nested Paging, PAE/NX, KVM Paravirtualization
```
Display
```
Video Memory:            16 MB
Graphics Controller:     VBoxVGA
Remote Desktop Server:   Disabled
Recording:               Disabled
```
Storage
```
Controller: IDE
 IDE Secondary Master:   [Optical Drive] Empty
Controller: SATA
 SATA Port 0:            dns server-disk1.vmdk (Normal, 20.00 GB)
```
Audio
```
Host Driver:   Windows DirectSound
Controller:    ICH AC97
```
Network
```
Adapter 1:  Intel PRO/1000 MT Desktop (NAT Network, 'NatNetwork')
```
USB
```
USB Controller:   OHCI
Device Filters:   0 (0 active)
```
Shared folders

None

Description

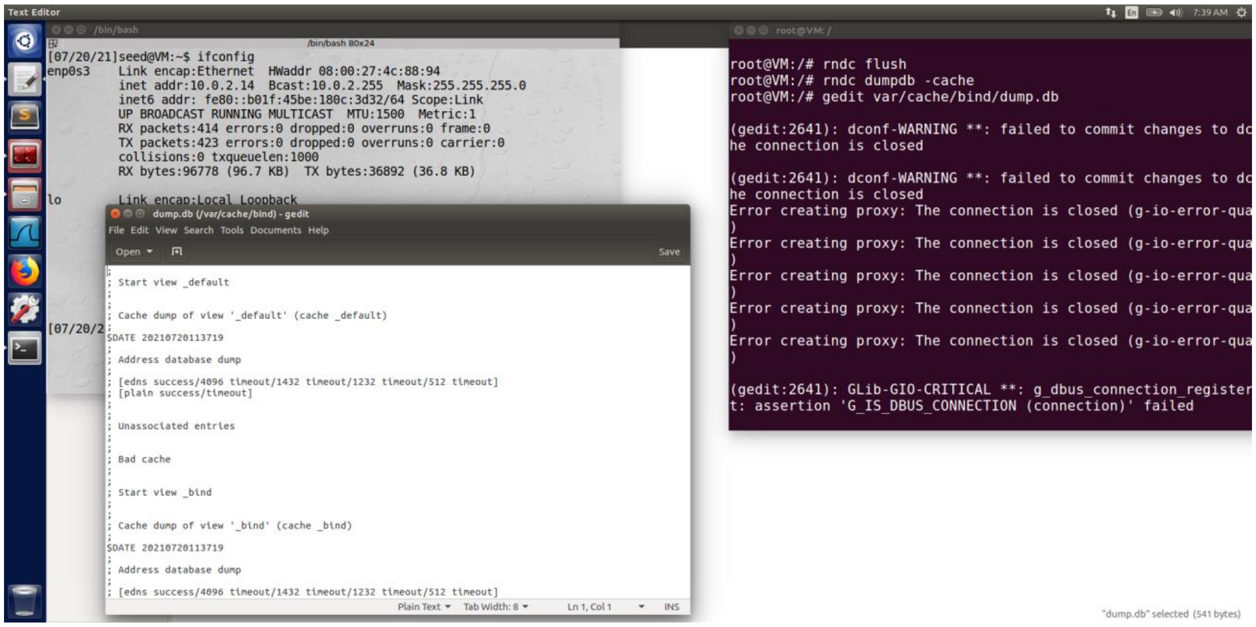
None

So, here we created 5 virtual machines. We will use the **Attacker**, **Trusted User 1** and **Normal DNS Server** to simulate the DoS attack. After that we will use the **Attacker**, **Trusted User 2** and **Secure DNS Server** to simulate some prevention measures.

Attack Steps:

- The DNS query packet is built with appropriate header values and question segment. A randomly generated garbage domain name is used to build the question segment. The format used is xxxxx.xxxxx.xxxxx
- IP header with spoofed IP Address and UDP header is created. The DNS Packet is the payload here. All of them are concatenated into a single packet.
- Then the packet is sent to the target DNS server.
- This process is run in an infinite loop.
- This attack is executed from the terminal as superuser, and it can be terminated with CTRL+c keyboard command.

Snapshots of the Attack:

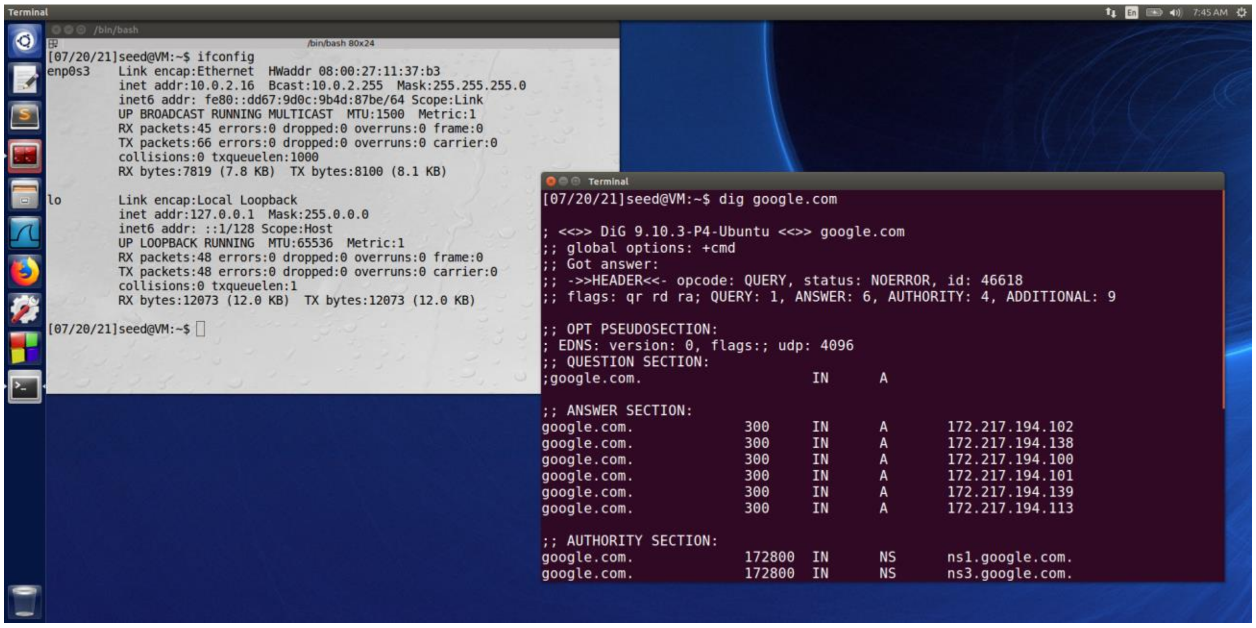

Text Editor
/bin/bash
/bin/bash 80x24
dump.db (/var/cache/bind) - gedit
File Edit View Search Tools Documents Help
Open
Save
Plain Text
Tab Width: 8
Ln 1, Col 1
INS
root@VM: /
7:39 AM
"dump.db" selected (541 bytes)

```
[07/20/21]seed@VM:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:4c:88:94
          inet addr:10.0.2.14  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::b01f:45be:180c:3d32/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:414 errors:0 dropped:0 overruns:0 frame:0
          TX packets:423 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:96778 (96.7 KB)  TX bytes:36892 (36.8 KB)

lo        Link encap:Local Loopback

[07/20/2
```

```
;
; Start view _default
;
;
; Cache dump of view '_default' (cache _default)
;
$DATE 20210720113719
;
; Address database dump
;
; [edns success/4096 timeout/1432 timeout/1232 timeout/512 timeout]
; [plain success/timeout]
;
;
; Unassociated entries
;
;
; Bad cache
;
;
; Start view _bind
;
;
; Cache dump of view '_bind' (cache _bind)
;
$DATE 20210720113719
;
; Address database dump
;
; [edns success/4096 timeout/1432 timeout/1232 timeout/512 timeout]
```

```
root@VM:/# rndc flush
root@VM:/# rndc dumpdb -cache
root@VM:/# gedit var/cache/bind/dump.db

(gedit:2641): dconf-WARNING **: failed to commit changes to dc
he connection is closed

(gedit:2641): dconf-WARNING **: failed to commit changes to dc
he connection is closed
Error creating proxy: The connection is closed (g-io-error-qua
)
Error creating proxy: The connection is closed (g-io-error-qua
)
Error creating proxy: The connection is closed (g-io-error-qua
)
Error creating proxy: The connection is closed (g-io-error-qua
)
Error creating proxy: The connection is closed (g-io-error-qua
)

(gedit:2641): GLib-GIO-CRITICAL **: g_dbus_connection_register
t: assertion 'G_IS_DBUS_CONNECTION (connection)' failed
```

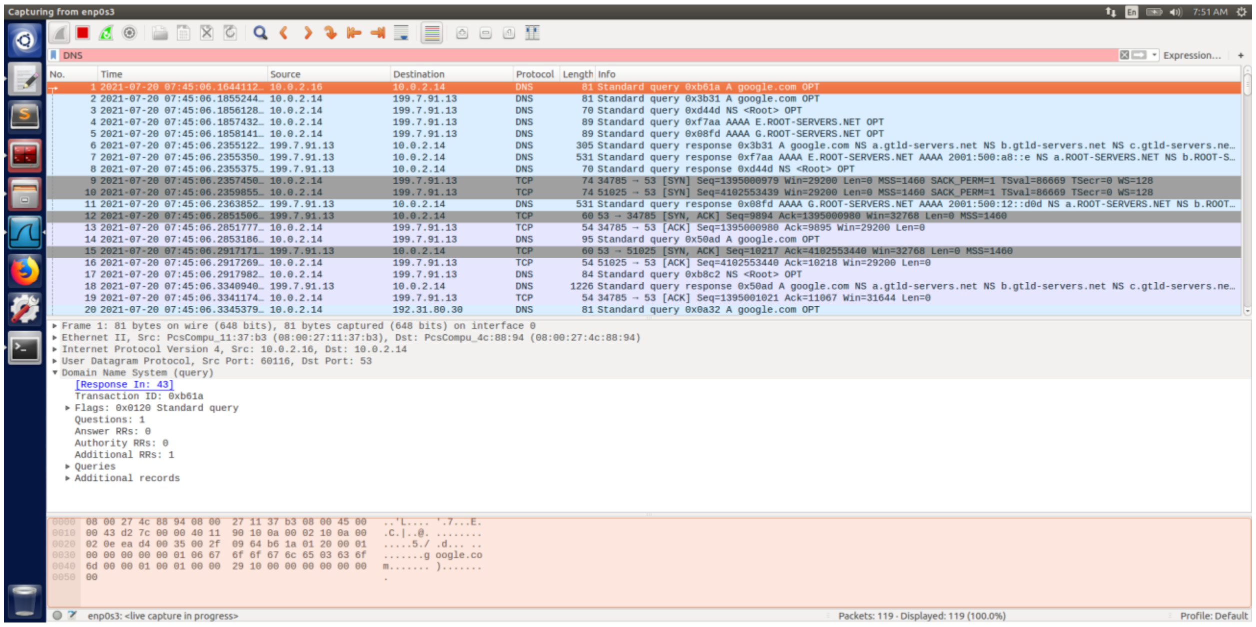
This is the initial state of the **Normal DNS Server** machine. We can see its IP address is 10.0.2.14 Its DNS cache size is 541 bytes, and it contains no record for the time being. Now we use 'dig' command from **Trusted User 1** to send a valid DNS query and see the changes.



```
[07/20/21]seed@VM:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:11:37:b3
          inet addr:10.0.2.16  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::dd67:9d0c:9b4d:87be/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:45 errors:0 dropped:0 overruns:0 frame:0
          TX packets:66 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:7819 (7.8 KB)  TX bytes:8100 (8.1 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:48 errors:0 dropped:0 overruns:0 frame:0
          TX packets:48 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:12073 (12.0 KB)  TX bytes:12073 (12.0 KB)

[07/20/21]seed@VM:~$
```

```
[07/20/21]seed@VM:~$ dig google.com

; <<>> DiG 9.10.3-P4-Ubuntu <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 46618
;; flags: qr rd ra; QUERY: 1, ANSWER: 6, AUTHORITY: 4, ADDITIONAL: 9

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

;; ANSWER SECTION:
google.com.             300     IN      A       172.217.194.102
google.com.             300     IN      A       172.217.194.138
google.com.             300     IN      A       172.217.194.100
google.com.             300     IN      A       172.217.194.101
google.com.             300     IN      A       172.217.194.139
google.com.             300     IN      A       172.217.194.113

;; AUTHORITY SECTION:
google.com.             172800  IN      NS      ns1.google.com.
google.com.             172800  IN      NS      ns3.google.com.
```

This is **Trusted User 1** machine with IP address 10.0.2.16; We can see the 'dig' command executed successfully and we got a valid DNS response. Now let's check the server's status.



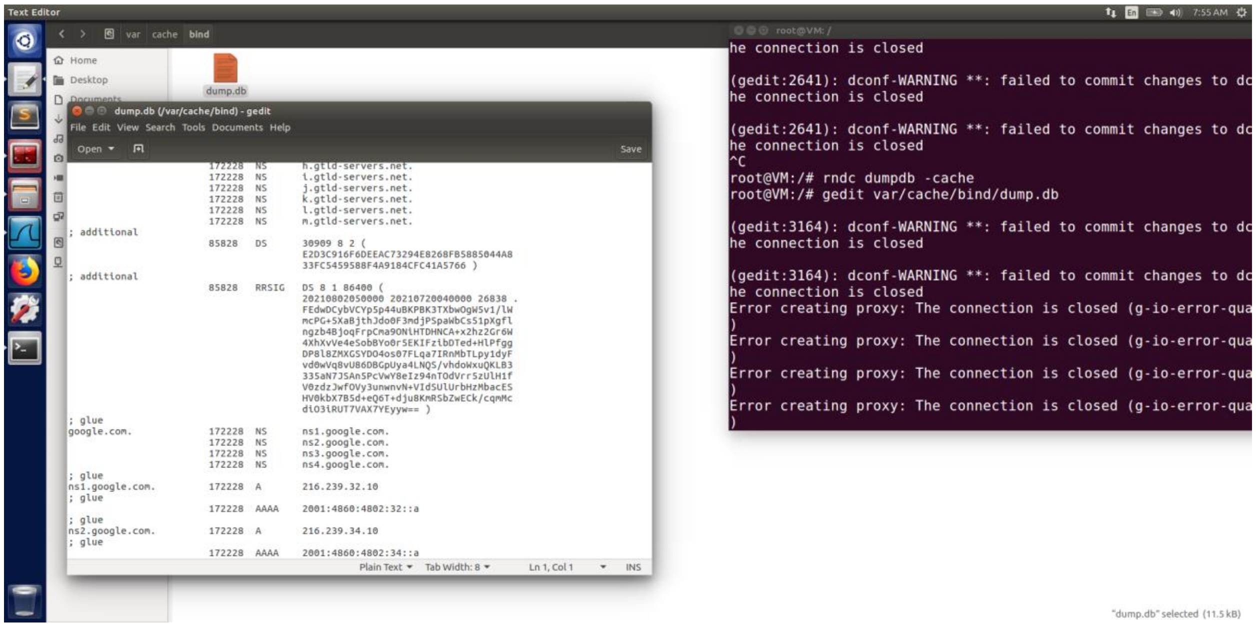
```
No. Time                           Source        Destination   Protocol Length Info
  1 2021-07-20 07:45:06.1644112…   10.0.2.16     10.0.2.14     DNS        81 Standard query 0xb61a A google.com OPT
  2 2021-07-20 07:45:06.1855244…   10.0.2.14     199.7.91.13   DNS        81 Standard query 0x3b31 A google.com OPT
  3 2021-07-20 07:45:06.1856128…   10.0.2.14     199.7.91.13   DNS        70 Standard query 0xd44d NS <Root> OPT
  4 2021-07-20 07:45:06.1857432…   10.0.2.14     199.7.91.13   DNS        89 Standard query 0xf7aa AAAA E.ROOT-SERVERS.NET OPT
  5 2021-07-20 07:45:06.1858141…   10.0.2.14     199.7.91.13   DNS        89 Standard query 0x08fd AAAA G.ROOT-SERVERS.NET OPT
  6 2021-07-20 07:45:06.2355122…   199.7.91.13   10.0.2.14     DNS       305 Standard query response 0x3b31 A google.com NS a.gtld-servers.net NS b.gtld-servers.net NS c.gtld-servers.ne…
  7 2021-07-20 07:45:06.2355350…   199.7.91.13   10.0.2.14     DNS       531 Standard query response 0xf7aa AAAA E.ROOT-SERVERS.NET AAAA 2001:500:a8::e NS a.ROOT-SERVERS.NET NS b.ROOT-S…
  8 2021-07-20 07:45:06.2355375…   199.7.91.13   10.0.2.14     DNS        70 Standard query response 0xd44d NS <Root> OPT
  9 2021-07-20 07:45:06.2357450…   10.0.2.14     199.7.91.13   TCP        74 34785 → 53 [SYN] Seq=1395000979 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=86669 TSecr=0 WS=128
 10 2021-07-20 07:45:06.2359855…   10.0.2.14     199.7.91.13   TCP        74 51025 → 53 [SYN] Seq=4102553439 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=86669 TSecr=0 WS=128
 11 2021-07-20 07:45:06.2363852…   199.7.91.13   10.0.2.14     DNS       531 Standard query response 0x08fd AAAA G.ROOT-SERVERS.NET AAAA 2001:500:12::d0d NS a.ROOT-SERVERS.NET NS b.ROOT…
 12 2021-07-20 07:45:06.2851506…   199.7.91.13   10.0.2.14     TCP        60 53 → 34785 [SYN, ACK] Seq=9894 Ack=1395000980 Win=32768 Len=0 MSS=1460
 13 2021-07-20 07:45:06.2851777…   10.0.2.14     199.7.91.13   TCP        54 34785 → 53 [ACK] Seq=1395000980 Ack=9895 Win=29200 Len=0
 14 2021-07-20 07:45:06.2853186…   10.0.2.14     199.7.91.13   DNS        95 Standard query 0x50ad A google.com OPT
 15 2021-07-20 07:45:06.2917171…   199.7.91.13   10.0.2.14     TCP        60 53 → 51025 [SYN, ACK] Seq=10217 Ack=4102553440 Win=32768 Len=0 MSS=1460
 16 2021-07-20 07:45:06.2917269…   10.0.2.14     199.7.91.13   TCP        54 51025 → 53 [ACK] Seq=4102553440 Ack=10218 Win=29200 Len=0
 17 2021-07-20 07:45:06.2917982…   10.0.2.14     199.7.91.13   DNS        84 Standard query 0xb8c2 NS <Root> OPT
 18 2021-07-20 07:45:06.3340940…   199.7.91.13   10.0.2.14     DNS      1226 Standard query response 0x50ad A google.com NS a.gtld-servers.net NS b.gtld-servers.net NS c.gtld-servers.ne…
 19 2021-07-20 07:45:06.3341174…   10.0.2.14     199.7.91.13   TCP        54 34785 → 53 [ACK] Seq=1395001021 Ack=11067 Win=31644 Len=0
 20 2021-07-20 07:45:06.3345379…   10.0.2.14     192.31.80.30  DNS        81 Standard query 0x0a32 A google.com OPT
```

```
▸ Frame 1: 81 bytes on wire (648 bits), 81 bytes captured (648 bits) on interface 0
▸ Ethernet II, Src: PcsCompu_11:37:b3 (08:00:27:11:37:b3), Dst: PcsCompu_4c:88:94 (08:00:27:4c:88:94)
▸ Internet Protocol Version 4, Src: 10.0.2.16, Dst: 10.0.2.14
▸ User Datagram Protocol, Src Port: 60116, Dst Port: 53
▾ Domain Name System (query)
    [Response In: 43]
    Transaction ID: 0xb61a
  ▸ Flags: 0x0120 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 1
  ▸ Queries
  ▸ Additional records
```

```
0000  08 00 27 4c 88 94 08 00  27 11 37 b3 08 00 45 00   ..'L.... '.7...E.
0010  00 43 d2 7c 00 00 40 11  90 10 0a 00 02 10 0a 00   .C.|..@. ........
0020  02 0e ea d4 00 35 00 2f  09 64 b6 1a 01 20 00 01   .....5./ .d... ..
0030  00 00 00 00 00 01 06 67  6f 6f 67 6c 65 03 63 6f   .......g oogle.co
0040  6d 00 00 01 00 01 00 00  29 10 00 00 00 00 00 00   m....... ).......
0050  00                                                 .
```

Capturing from enp0s3
DNS
Expression…
enp0s3: <live capture in progress>
Packets: 119 · Displayed: 119 (100.0%)
Profile: Default

We can check the packets in Wireshark. Here we can see a DNS query about google.com from Source 10.0.2.16 to Destination 10.0.2.14 which is the machine itself. Now if we check the DNS cache status,

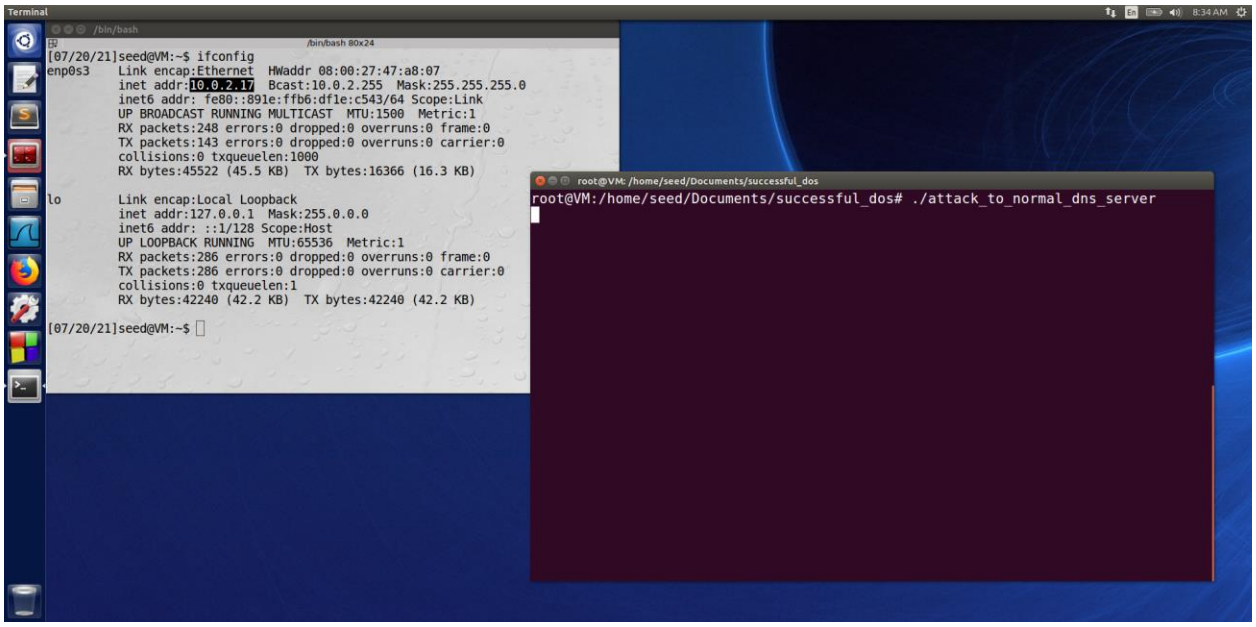


```
         172228  NS      h.gtld-servers.net.
         172228  NS      i.gtld-servers.net.
         172228  NS      j.gtld-servers.net.
         172228  NS      k.gtld-servers.net.
         172228  NS      l.gtld-servers.net.
         172228  NS      m.gtld-servers.net.
; additional
         85828   DS      30909 8 2 (
                         E2D3C916F6DEEAC73294E8268FB5885044A8
                         33FC5459588F4A9184CFC41A5766 )
; additional
         85828   RRSIG   DS 8 1 86400 (
                         20210802050000 20210720040000 26838 .
                         FEdwDCybVCYp5p44uBKPBK3TXbwOgW5v1/lW
                         mcPG+5XaBjthJdo0F3mdjPSpaWbCs51pXgfl
                         ngzb4BjoqFrpCma9ONiHTDHNCA+x2hz2Gr6W
                         4XhXvVe4eSobBYo0r5EKIFzibDTed+HlPfgg
                         DP8l8ZMXGSYDO4os07FLqa7IRnMbTLpy1dyF
                         vd0wVq8vU86DBGpUya4LNQS/vhdoWxuQKLB3
                         335aN7JSAn5PcVwY8eIz94nTOdVrr5zUlH1f
                         V0zdzJwfOVy3unwnvN+VIdSUlUrbHzMbacES
                         HV0kbX7B5d+eQ6T+dju8KmRSbZwECk/cqmMc
                         diO3iRUT7VAX7YEyyw== )
; glue
google.com.      172228  NS      ns1.google.com.
                 172228  NS      ns2.google.com.
                 172228  NS      ns3.google.com.
                 172228  NS      ns4.google.com.
; glue
ns1.google.com.  172228  A       216.239.32.10
; glue
                 172228  AAAA    2001:4860:4802:32::a
; glue
ns2.google.com.  172228  A       216.239.34.10
; glue
                 172228  AAAA    2001:4860:4802:34::a
```

```
he connection is closed

(gedit:2641): dconf-WARNING **: failed to commit changes to dc
he connection is closed

(gedit:2641): dconf-WARNING **: failed to commit changes to dc
he connection is closed
^C
root@VM:/# rndc dumpdb -cache
root@VM:/# gedit var/cache/bind/dump.db

(gedit:3164): dconf-WARNING **: failed to commit changes to dc
he connection is closed

(gedit:3164): dconf-WARNING **: failed to commit changes to dc
he connection is closed
Error creating proxy: The connection is closed (g-io-error-qua
)
Error creating proxy: The connection is closed (g-io-error-qua
)
Error creating proxy: The connection is closed (g-io-error-qua
)
Error creating proxy: The connection is closed (g-io-error-qua
)
```

"dump.db" selected (11.5 kB)

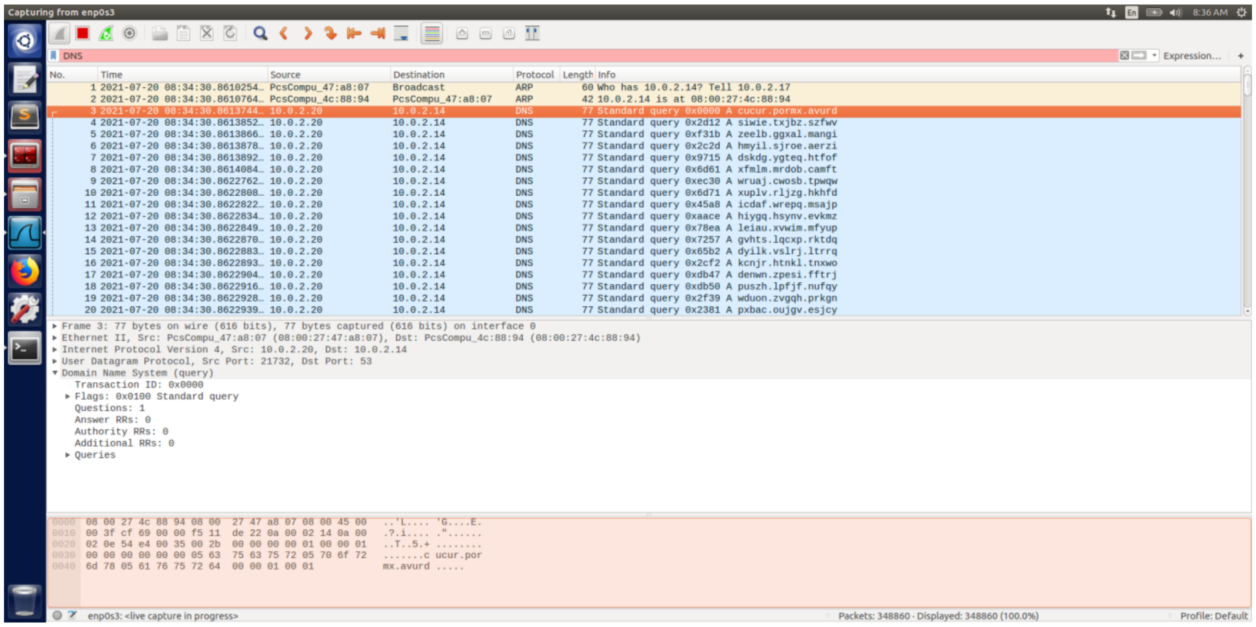
We can see the entry google.com in the DNS cache and the cache size increased to 11.5 KB. Now we will start the DoS attack and see what happens.



```
[07/20/21]seed@VM:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:47:a8:07
          inet addr:10.0.2.17  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::891e:ffb6:df1e:c543/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:248 errors:0 dropped:0 overruns:0 frame:0
          TX packets:143 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:45522 (45.5 KB)  TX bytes:16366 (16.3 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:286 errors:0 dropped:0 overruns:0 frame:0
          TX packets:286 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:42240 (42.2 KB)  TX bytes:42240 (42.2 KB)

[07/20/21]seed@VM:~$
```

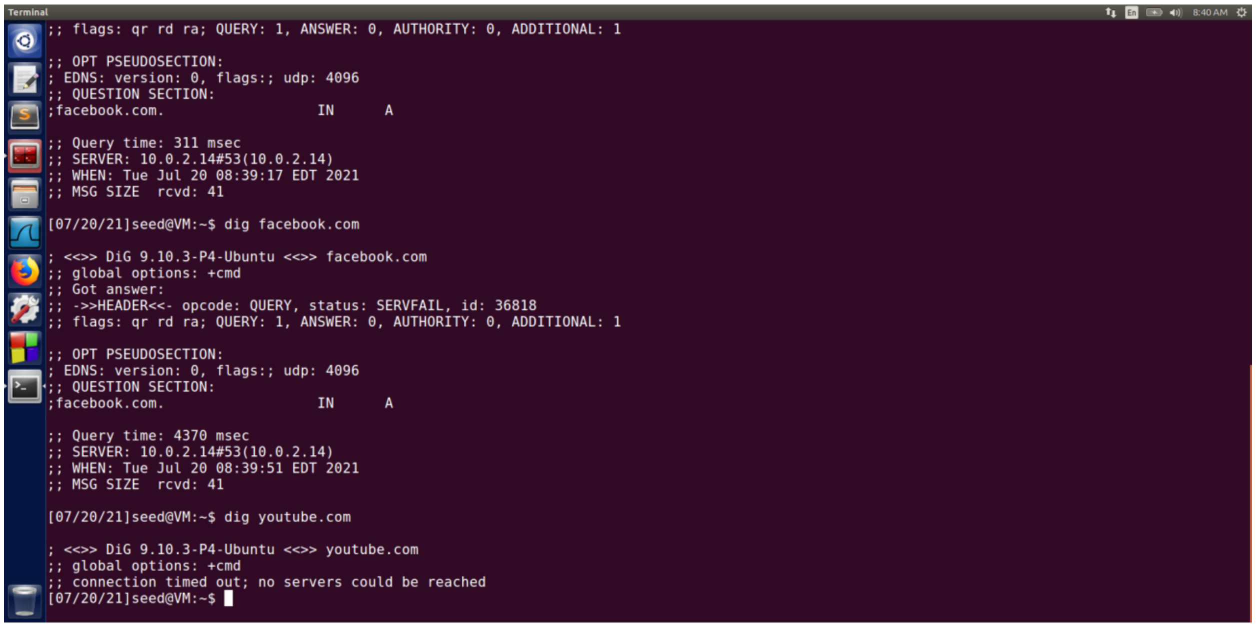
```
root@VM:/home/seed/Documents/successful_dos# ./attack_to_normal_dns_server
```

This is the **Attacker** machine with IP address 10.0.2.17; We start the DoS attack from here. Now let’s check the server using Wireshark



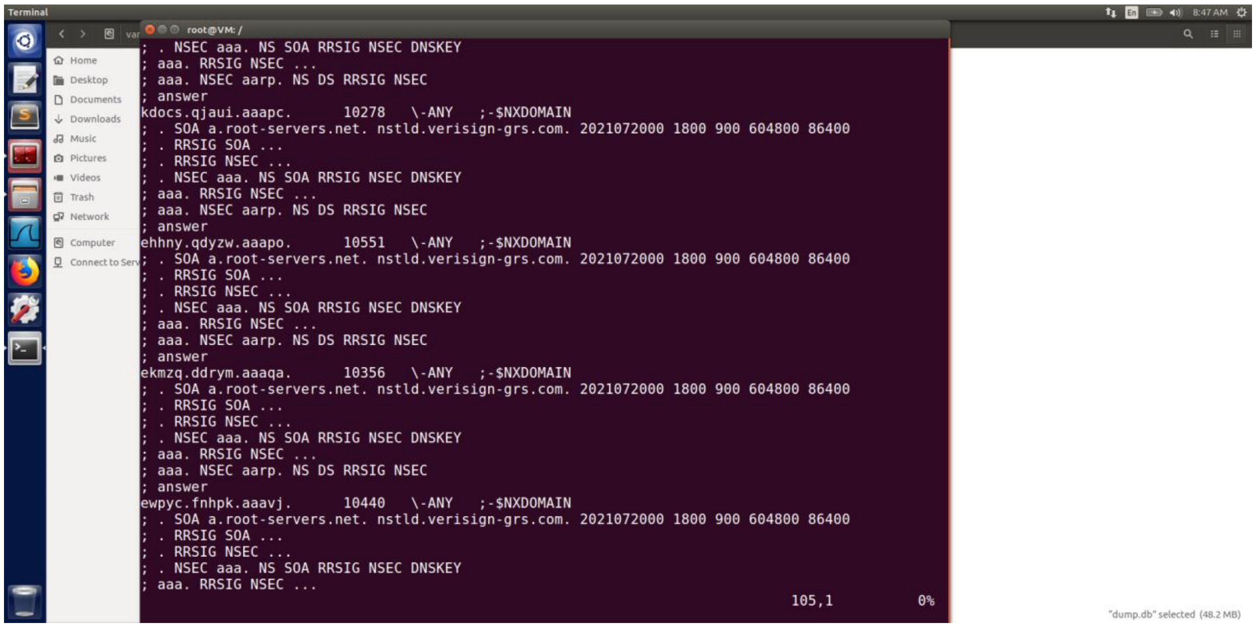
```
No.  Time                            Source              Destination         Protocol Length Info
   1 2021-07-20 08:34:30.8610254…  PcsCompu_47:a8:07  Broadcast           ARP       60 Who has 10.0.2.14? Tell 10.0.2.17
   2 2021-07-20 08:34:30.8610764…  PcsCompu_4c:88:94  PcsCompu_47:a8:07   ARP       42 10.0.2.14 is at 08:00:27:4c:88:94
   3 2021-07-20 08:34:30.8613744…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0x0000 A cucur.pormx.avurd
   4 2021-07-20 08:34:30.8613852…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0x2d12 A siwie.txjbz.szfwv
   5 2021-07-20 08:34:30.8613866…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0xf31b A zeelb.ggxal.mangi
   6 2021-07-20 08:34:30.8613878…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0x2c2d A hmyil.sjroe.aerzi
   7 2021-07-20 08:34:30.8613892…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0x9715 A dskdg.ygteq.htfof
   8 2021-07-20 08:34:30.8614084…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0x6d61 A xfmlm.mrdob.camft
   9 2021-07-20 08:34:30.8622762…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0xec30 A wruaj.cwosb.tpwqw
  10 2021-07-20 08:34:30.8622808…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0x6d71 A xuplv.rljzg.hkhfd
  11 2021-07-20 08:34:30.8622822…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0x45a8 A icdaf.wrepq.msajp
  12 2021-07-20 08:34:30.8622834…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0xaace A hiygq.hsynv.evkmz
  13 2021-07-20 08:34:30.8622849…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0x78ea A leiau.xvwim.mfyup
  14 2021-07-20 08:34:30.8622870…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0x7257 A gvhts.lqcxp.rktdq
  15 2021-07-20 08:34:30.8622883…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0x65b2 A dyilk.vslrj.ltrrq
  16 2021-07-20 08:34:30.8622893…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0x2cf2 A kcnjr.htnkl.tnxwo
  17 2021-07-20 08:34:30.8622904…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0xdb47 A denwn.zpesi.fftrj
  18 2021-07-20 08:34:30.8622916…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0xdb50 A puszh.lpfjf.nufqy
  19 2021-07-20 08:34:30.8622928…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0x2f39 A wduon.zvgqh.prkgn
  20 2021-07-20 08:34:30.8622939…  10.0.2.20          10.0.2.14           DNS       77 Standard query 0x2381 A pxbac.oujgv.esjcy

▸ Frame 3: 77 bytes on wire (616 bits), 77 bytes captured (616 bits) on interface 0
▸ Ethernet II, Src: PcsCompu_47:a8:07 (08:00:27:47:a8:07), Dst: PcsCompu_4c:88:94 (08:00:27:4c:88:94)
▸ Internet Protocol Version 4, Src: 10.0.2.20, Dst: 10.0.2.14
▸ User Datagram Protocol, Src Port: 21732, Dst Port: 53
▾ Domain Name System (query)
    Transaction ID: 0x0000
  ▸ Flags: 0x0100 Standard query
    Questions: 1
    Answer RRs: 0
    Authority RRs: 0
    Additional RRs: 0
  ▸ Queries

0000  08 00 27 4c 88 94 08 00  27 47 a8 07 08 00 45 00   ..'L.... 'G....E.
0010  00 3f cf 69 00 00 f5 11  de 22 0a 00 02 14 0a 00   .?.i.... ."......
0020  02 0e 54 e4 00 35 00 2b  00 00 00 00 01 00 00 01   ..T..5.+ ........
0030  00 00 00 00 00 00 05 63  75 63 75 72 05 70 6f 72   .......c ucur.por
0040  6d 78 05 61 76 75 72 64  00 00 01 00 01            mx.avurd .....

enp0s3: <live capture in progress>        Packets: 348860 · Displayed: 348860 (100.0%)        Profile: Default
```

We can see, the server is flooded with garbage DNS queries from IP address 10.0.2.20 which is not the attacker’s own IP address. So, IP spoofing was successful. Now let’s check the Client machine and try to use 'dig' command like before.



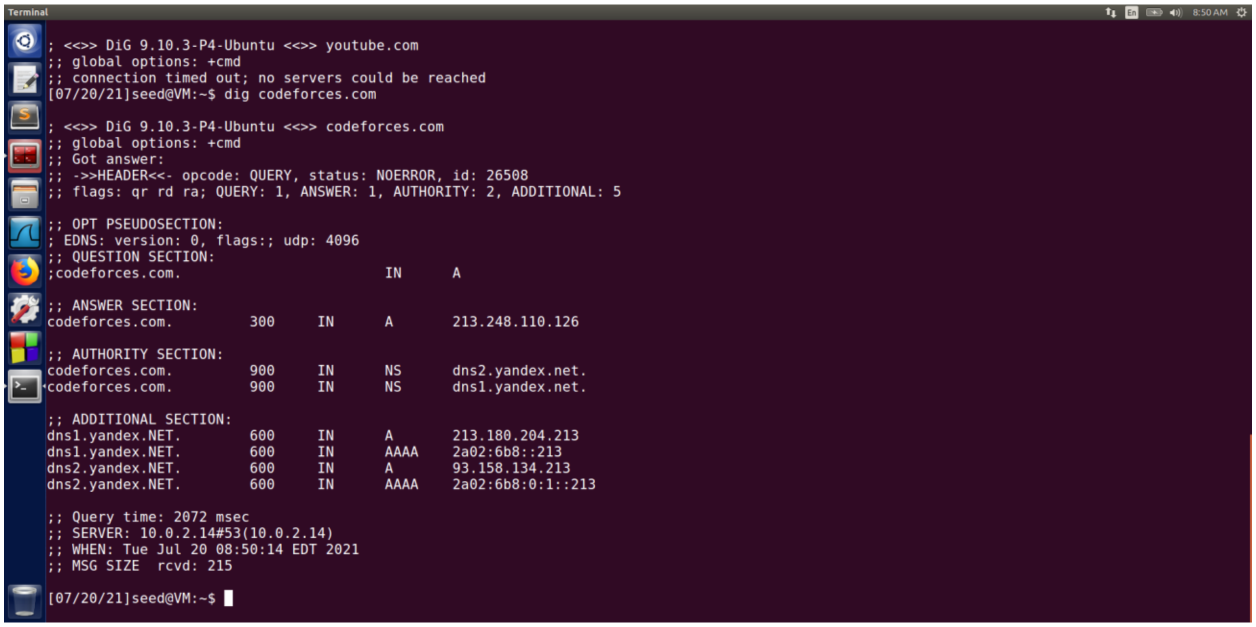
```
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

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

;; Query time: 311 msec
;; SERVER: 10.0.2.14#53(10.0.2.14)
;; WHEN: Tue Jul 20 08:39:17 EDT 2021
;; MSG SIZE  rcvd: 41

[07/20/21]seed@VM:~$ dig facebook.com

; <<>> DiG 9.10.3-P4-Ubuntu <<>> facebook.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: SERVFAIL, id: 36818
;; flags: qr rd ra; QUERY: 1, ANSWER: 0, AUTHORITY: 0, ADDITIONAL: 1

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

;; Query time: 4370 msec
;; SERVER: 10.0.2.14#53(10.0.2.14)
;; WHEN: Tue Jul 20 08:39:51 EDT 2021
;; MSG SIZE  rcvd: 41

[07/20/21]seed@VM:~$ dig youtube.com

; <<>> DiG 9.10.3-P4-Ubuntu <<>> youtube.com
;; global options: +cmd
;; connection timed out; no servers could be reached
[07/20/21]seed@VM:~$
```

We can see all the requests failed with either **SERVFAIL** status or **Connection Timed Out** Status. So, the client is successfully being denied of service. Let’s stop the attack and check the server cache.



```
; . NSEC aaa. NS SOA RRSIG NSEC DNSKEY
; aaa. RRSIG NSEC ...
; aaa. NSEC aarp. NS DS RRSIG NSEC
; answer
kdocs.qjaui.aaapc.      10278   \-ANY   ;-$NXDOMAIN
; . SOA a.root-servers.net. nstld.verisign-grs.com. 2021072000 1800 900 604800 86400
; . RRSIG SOA ...
; . RRSIG NSEC ...
; . NSEC aaa. NS SOA RRSIG NSEC DNSKEY
; aaa. RRSIG NSEC ...
; aaa. NSEC aarp. NS DS RRSIG NSEC
; answer
ehhny.qdyzw.aaapo.      10551   \-ANY   ;-$NXDOMAIN
; . SOA a.root-servers.net. nstld.verisign-grs.com. 2021072000 1800 900 604800 86400
; . RRSIG SOA ...
; . RRSIG NSEC ...
; . NSEC aaa. NS SOA RRSIG NSEC DNSKEY
; aaa. RRSIG NSEC ...
; aaa. NSEC aarp. NS DS RRSIG NSEC
; answer
ekmzq.ddrym.aaaqa.      10356   \-ANY   ;-$NXDOMAIN
; . SOA a.root-servers.net. nstld.verisign-grs.com. 2021072000 1800 900 604800 86400
; . RRSIG SOA ...
; . RRSIG NSEC ...
; . NSEC aaa. NS SOA RRSIG NSEC DNSKEY
; aaa. RRSIG NSEC ...
; aaa. NSEC aarp. NS DS RRSIG NSEC
; answer
ewpyc.fnhpk.aaavj.      10440   \-ANY   ;-$NXDOMAIN
; . SOA a.root-servers.net. nstld.verisign-grs.com. 2021072000 1800 900 604800 86400
; . RRSIG SOA ...
; . RRSIG NSEC ...
; . NSEC aaa. NS SOA RRSIG NSEC DNSKEY
; aaa. RRSIG NSEC ...
                                                                    105,1         0%
```

We can see the cache is filled with NXDOMAIN entries from the attack. Its size has increased to 48.2 MB. Now that we stopped the attack, let’s try to use 'dig' command from the **Trusted User 1** machine again.



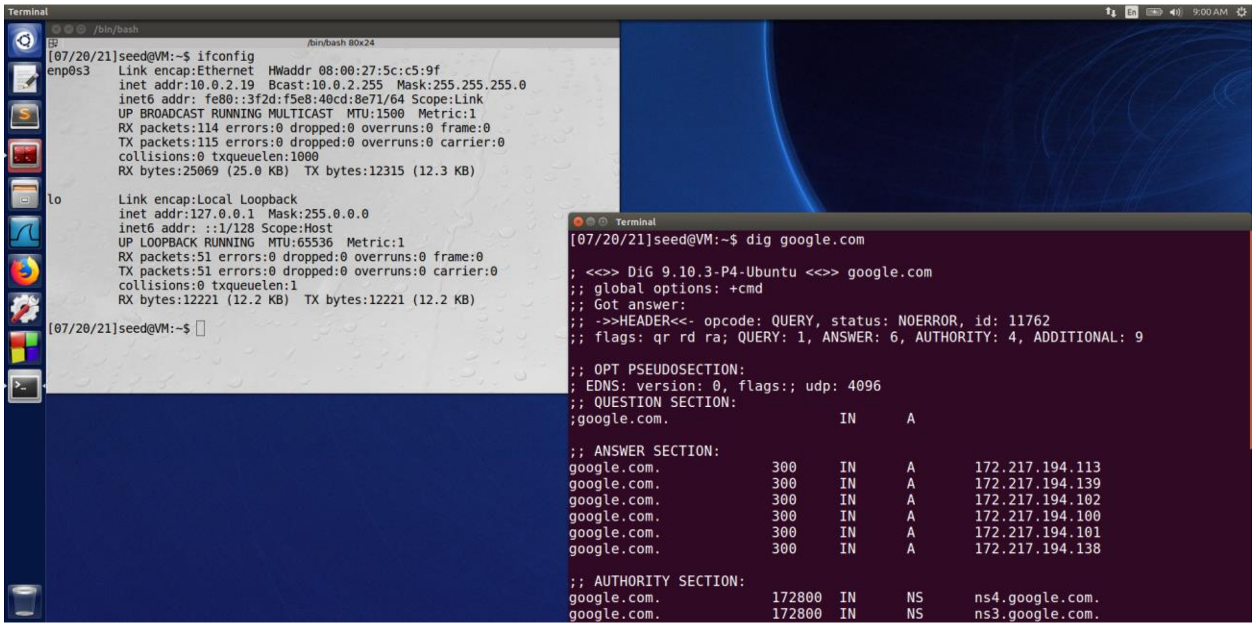
```
; <<>> DiG 9.10.3-P4-Ubuntu <<>> youtube.com
;; global options: +cmd
;; connection timed out; no servers could be reached
[07/20/21]seed@VM:~$ dig codeforces.com

; <<>> DiG 9.10.3-P4-Ubuntu <<>> codeforces.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 26508
;; flags: qr rd ra; QUERY: 1, ANSWER: 1, AUTHORITY: 2, ADDITIONAL: 5

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

;; ANSWER SECTION:
codeforces.com.         300     IN      A       213.248.110.126

;; AUTHORITY SECTION:
codeforces.com.         900     IN      NS      dns2.yandex.net.
codeforces.com.         900     IN      NS      dns1.yandex.net.

;; ADDITIONAL SECTION:
dns1.yandex.NET.        600     IN      A       213.180.204.213
dns1.yandex.NET.        600     IN      AAAA    2a02:6b8::213
dns2.yandex.NET.        600     IN      A       93.158.134.213
dns2.yandex.NET.        600     IN      AAAA    2a02:6b8:0:1::213

;; Query time: 2072 msec
;; SERVER: 10.0.2.14#53(10.0.2.14)
;; WHEN: Tue Jul 20 08:50:14 EDT 2021
;; MSG SIZE  rcvd: 215

[07/20/21]seed@VM:~$
```

We can see the command is working again just as it was before.

So, from the observation above, we can say that the DoS attack is working perfectly.

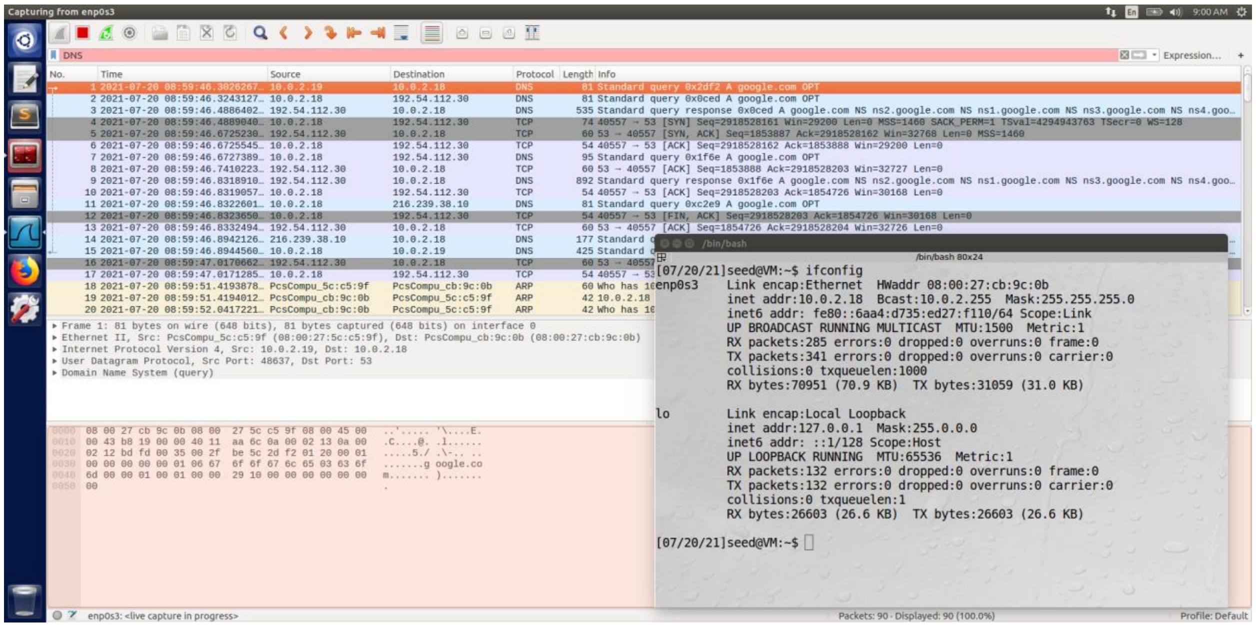
Snapshots of Attack on a Secured DNS Server:

Here we will target the **Secure DNS Server** machine for our attack. The client will be **Trusted User 2,** and the attacker will be same as before.



```
[07/20/21]seed@VM:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:5c:c5:9f
          inet addr:10.0.2.19  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::3f2d:f5e8:40cd:8e71/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:114 errors:0 dropped:0 overruns:0 frame:0
          TX packets:115 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:25069 (25.0 KB)  TX bytes:12315 (12.3 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:51 errors:0 dropped:0 overruns:0 frame:0
          TX packets:51 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:12221 (12.2 KB)  TX bytes:12221 (12.2 KB)

[07/20/21]seed@VM:~$
```

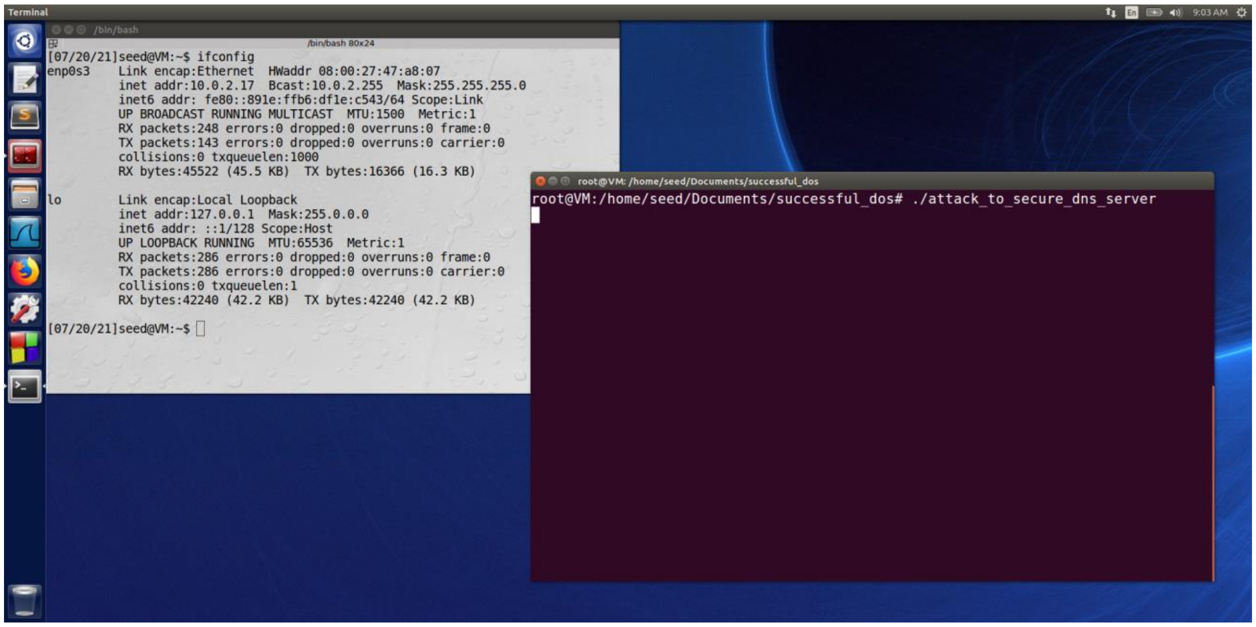
```
[07/20/21]seed@VM:~$ dig google.com

; <<>> DiG 9.10.3-P4-Ubuntu <<>> google.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 11762
;; flags: qr rd ra; QUERY: 1, ANSWER: 6, AUTHORITY: 4, ADDITIONAL: 9

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

;; ANSWER SECTION:
google.com.             300     IN      A       172.217.194.113
google.com.             300     IN      A       172.217.194.139
google.com.             300     IN      A       172.217.194.102
google.com.             300     IN      A       172.217.194.100
google.com.             300     IN      A       172.217.194.101
google.com.             300     IN      A       172.217.194.138

;; AUTHORITY SECTION:
google.com.             172800  IN      NS      ns4.google.com.
google.com.             172800  IN      NS      ns3.google.com.
```



```
No. Time                           Source             Destination        Protocol Length Info
  1 2021-07-20 08:59:46.3026267…   10.0.2.19          10.0.2.18          DNS       81 Standard query 0x2df2 A google.com OPT
  2 2021-07-20 08:59:46.3243127…   10.0.2.18          192.54.112.30      DNS       81 Standard query 0x0ced A google.com OPT
  3 2021-07-20 08:59:46.4886402…   192.54.112.30      10.0.2.18          DNS      535 Standard query response 0x0ced A google.com NS ns2.google.com NS ns1.google.com NS ns3.google.com NS ns4.goo…
  4 2021-07-20 08:59:46.4889040…   10.0.2.18          192.54.112.30      TCP       74 40557 → 53 [SYN] Seq=2918528161 Win=29200 Len=0 MSS=1460 SACK_PERM=1 TSval=4294943763 TSecr=0 WS=128
  5 2021-07-20 08:59:46.6725230…   192.54.112.30      10.0.2.18          TCP       60 53 → 40557 [SYN, ACK] Seq=1853887 Ack=2918528162 Win=32768 Len=0 MSS=1460
  6 2021-07-20 08:59:46.6725545…   10.0.2.18          192.54.112.30      TCP       54 40557 → 53 [ACK] Seq=2918528162 Ack=1853888 Win=29200 Len=0
  7 2021-07-20 08:59:46.6727389…   10.0.2.18          192.54.112.30      DNS       95 Standard query 0x1f6e A google.com OPT
  8 2021-07-20 08:59:46.7410223…   192.54.112.30      10.0.2.18          TCP       60 53 → 40557 [ACK] Seq=1853888 Ack=2918528203 Win=32727 Len=0
  9 2021-07-20 08:59:46.8318910…   192.54.112.30      10.0.2.18          DNS      892 Standard query response 0x1f6e A google.com NS ns2.google.com NS ns1.google.com NS ns3.google.com NS ns4.goo…
 10 2021-07-20 08:59:46.8319057…   10.0.2.18          192.54.112.30      TCP       54 40557 → 53 [ACK] Seq=2918528203 Ack=1854726 Win=30168 Len=0
 11 2021-07-20 08:59:46.8322601…   10.0.2.18          216.239.38.10      DNS       81 Standard query 0xc2e9 A google.com OPT
 12 2021-07-20 08:59:46.8323650…   10.0.2.18          192.54.112.30      TCP       54 40557 → 53 [FIN, ACK] Seq=2918528203 Ack=1854726 Win=30168 Len=0
 13 2021-07-20 08:59:46.8332494…   192.54.112.30      10.0.2.18          TCP       60 53 → 40557 [ACK] Seq=1854726 Ack=2918528204 Win=32726 Len=0
 14 2021-07-20 08:59:46.8942126…   216.239.38.10      10.0.2.18          DNS      177 Standard q
 15 2021-07-20 08:59:46.8944560…   10.0.2.18          10.0.2.19          DNS      425 Standard q
 16 2021-07-20 08:59:47.0170662…   192.54.112.30      10.0.2.18          TCP       60 53 → 40557
 17 2021-07-20 08:59:47.0171285…   10.0.2.18          192.54.112.30      TCP       54 40557 → 53
 18 2021-07-20 08:59:51.4193878…   PcsCompu_5c:c5:9f  PcsCompu_cb:9c:0b  ARP       60 Who has 10
 19 2021-07-20 08:59:51.4194012…   PcsCompu_cb:9c:0b  PcsCompu_5c:c5:9f  ARP       42 10.0.2.18
 20 2021-07-20 08:59:52.0417221…   PcsCompu_cb:9c:0b  PcsCompu_5c:c5:9f  ARP       42 Who has 10

▸ Frame 1: 81 bytes on wire (648 bits), 81 bytes captured (648 bits) on interface 0
▸ Ethernet II, Src: PcsCompu_5c:c5:9f (08:00:27:5c:c5:9f), Dst: PcsCompu_cb:9c:0b (08:00:27:cb:9c:0b)
▸ Internet Protocol Version 4, Src: 10.0.2.19, Dst: 10.0.2.18
▸ User Datagram Protocol, Src Port: 48637, Dst Port: 53
▸ Domain Name System (query)

0000  08 00 27 cb 9c 0b 08 00  27 5c c5 9f 08 00 45 00   ..'.....'\....E.
0010  00 43 b8 19 00 00 40 11  aa 6c 0a 00 02 13 0a 00   .C....@. .l......
0020  02 12 bd fd 00 35 00 2f  be 5c 2d f2 01 20 00 01   .....5./ .\-.. ..
0030  00 00 00 00 00 01 06 67  6f 6f 67 6c 65 03 63 6f   .......g oogle.co
0040  6d 00 00 01 00 01 00 00  29 10 00 00 00 00 00 00   m....... ).......
0050  00                                                 .

enp0s3: <live capture in progress>     Packets: 90 · Displayed: 90 (100.0%)     Profile: Default
```

```
[07/20/21]seed@VM:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:cb:9c:0b
          inet addr:10.0.2.18  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::6aa4:d735:ed27:f110/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:285 errors:0 dropped:0 overruns:0 frame:0
          TX packets:341 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:70951 (70.9 KB)  TX bytes:31059 (31.0 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:132 errors:0 dropped:0 overruns:0 frame:0
          TX packets:132 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:26603 (26.6 KB)  TX bytes:26603 (26.6 KB)

[07/20/21]seed@VM:~$
```

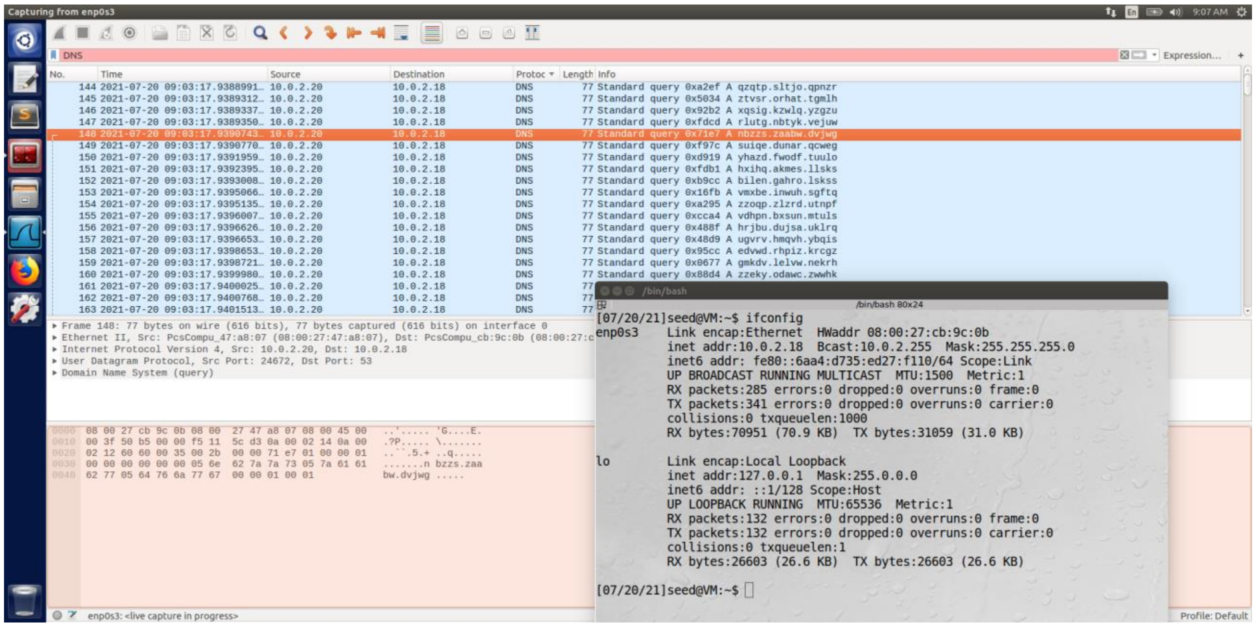
So, we can see the **Trusted User 2** machine has IP address 10.0.2.19 and sends DNS query to **Secured DNS Server** machine with IP address 10.0.2.18; The query is resolved, and the client got the DNS response. Now, we will start the attack.



```
[07/20/21]seed@VM:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:47:a8:07
          inet addr:10.0.2.17  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::891e:ffb6:df1e:c543/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:248 errors:0 dropped:0 overruns:0 frame:0
          TX packets:143 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:45522 (45.5 KB)  TX bytes:16366 (16.3 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:286 errors:0 dropped:0 overruns:0 frame:0
          TX packets:286 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:42240 (42.2 KB)  TX bytes:42240 (42.2 KB)

[07/20/21]seed@VM:~$
```

```
root@VM:/home/seed/Documents/successful_dos# ./attack_to_secure_dns_server
```

Here, we started the DoS attack. Let's check the DNS server status with Wireshark.

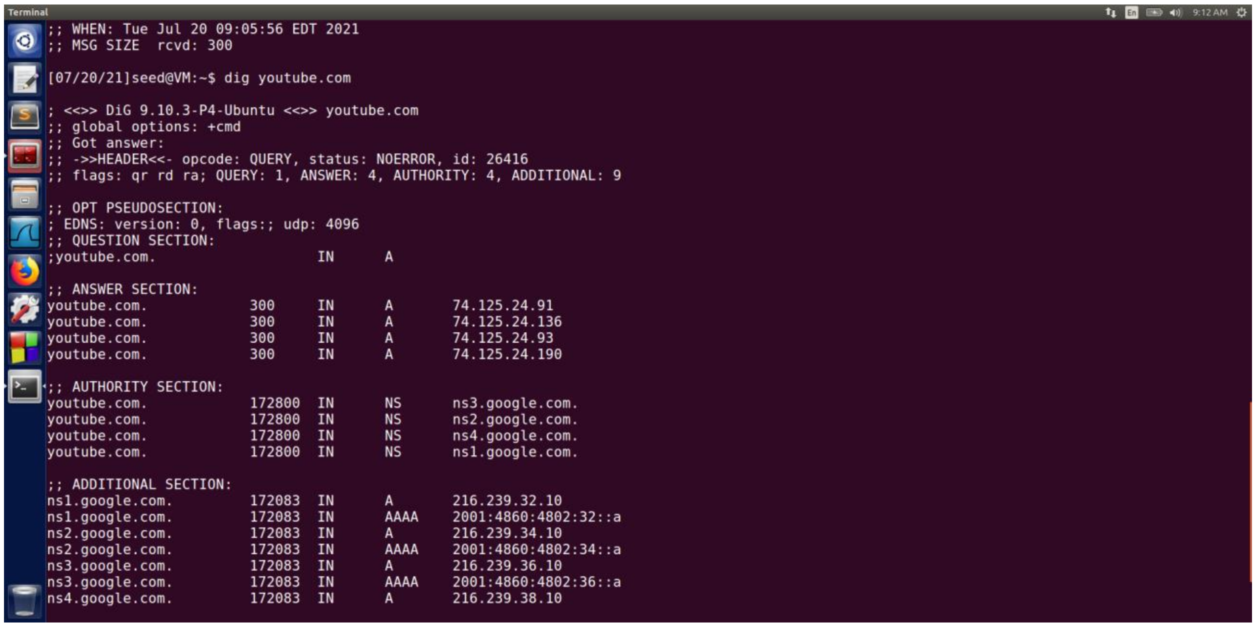


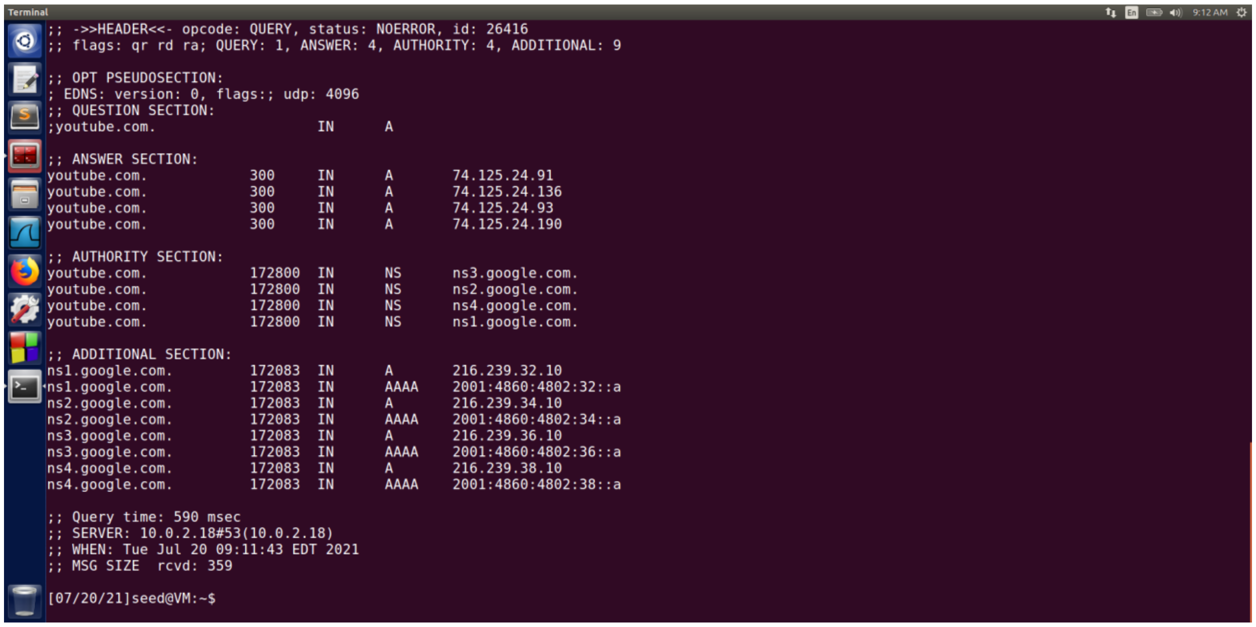
```
No.  Time                          Source     Destination Protoc Length Info
144 2021-07-20 09:03:17.9388991… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0xa2ef A qzqtp.sltjo.qpnzr
145 2021-07-20 09:03:17.9389312… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0x5034 A ztvsr.orhat.tgmlh
146 2021-07-20 09:03:17.9389337… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0x92b2 A xqsig.kzwlq.yzgzu
147 2021-07-20 09:03:17.9389350… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0xfdcd A rlutg.nbtyk.vejuw
148 2021-07-20 09:03:17.9390743… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0x71e7 A nbzzs.zaabw.dvjwg
149 2021-07-20 09:03:17.9390770… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0xf97c A suiqe.dunar.qcweg
150 2021-07-20 09:03:17.9391959… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0xd919 A yhazd.fwodf.tuulo
151 2021-07-20 09:03:17.9392395… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0xfdb1 A hxihq.akmes.llsks
152 2021-07-20 09:03:17.9393008… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0xb9cc A bilen.gahro.lskss
153 2021-07-20 09:03:17.9395066… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0x16fb A vmxbe.inwuh.sgftq
154 2021-07-20 09:03:17.9395135… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0xa295 A zzoqp.zlzrd.utnpf
155 2021-07-20 09:03:17.9396007… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0xcca4 A vdhpn.bxsun.mtuls
156 2021-07-20 09:03:17.9396626… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0x488f A hrjbu.dujsa.uklrq
157 2021-07-20 09:03:17.9396653… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0x48d9 A ugvrv.hmqvh.ybqis
158 2021-07-20 09:03:17.9398653… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0x95cc A edvwd.rhpiz.krcgz
159 2021-07-20 09:03:17.9398721… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0x0677 A gmkdv.lelvw.nekrh
160 2021-07-20 09:03:17.9399980… 10.0.2.20  10.0.2.18   DNS    77 Standard query 0x88d4 A zzeky.odawc.zwwhk
161 2021-07-20 09:03:17.9400025… 10.0.2.20  10.0.2.18   DNS    77
162 2021-07-20 09:03:17.9400768… 10.0.2.20  10.0.2.18   DNS    77
163 2021-07-20 09:03:17.9401513… 10.0.2.20  10.0.2.18   DNS    77

▸ Frame 148: 77 bytes on wire (616 bits), 77 bytes captured (616 bits) on interface 0
▸ Ethernet II, Src: PcsCompu_47:a8:07 (08:00:27:47:a8:07), Dst: PcsCompu_cb:9c:0b (08:00:27:c
▸ Internet Protocol Version 4, Src: 10.0.2.20, Dst: 10.0.2.18
▸ User Datagram Protocol, Src Port: 24672, Dst Port: 53
▸ Domain Name System (query)

0000  08 00 27 cb 9c 0b 08 00  27 47 a8 07 08 00 45 00   ..'..... 'G....E.
0010  00 3f 50 b5 00 00 f5 11  5c d3 0a 00 02 14 0a 00   .?P..... \.......
0020  02 12 60 60 00 35 00 2b  00 00 71 e7 01 00 00 01   ..``.5.+ ..q.....
0030  00 00 00 00 00 00 05 6e  62 7a 7a 73 05 7a 61 61   .......n bzzs.zaa
0040  62 77 05 64 76 6a 77 67  00 00 01 00 01            bw.dvjwg .....
```

```
[07/20/21]seed@VM:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:cb:9c:0b
          inet addr:10.0.2.18  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::6aa4:d735:ed27:f110/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:285 errors:0 dropped:0 overruns:0 frame:0
          TX packets:341 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:70951 (70.9 KB)  TX bytes:31059 (31.0 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:132 errors:0 dropped:0 overruns:0 frame:0
          TX packets:132 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:26603 (26.6 KB)  TX bytes:26603 (26.6 KB)

[07/20/21]seed@VM:~$
```

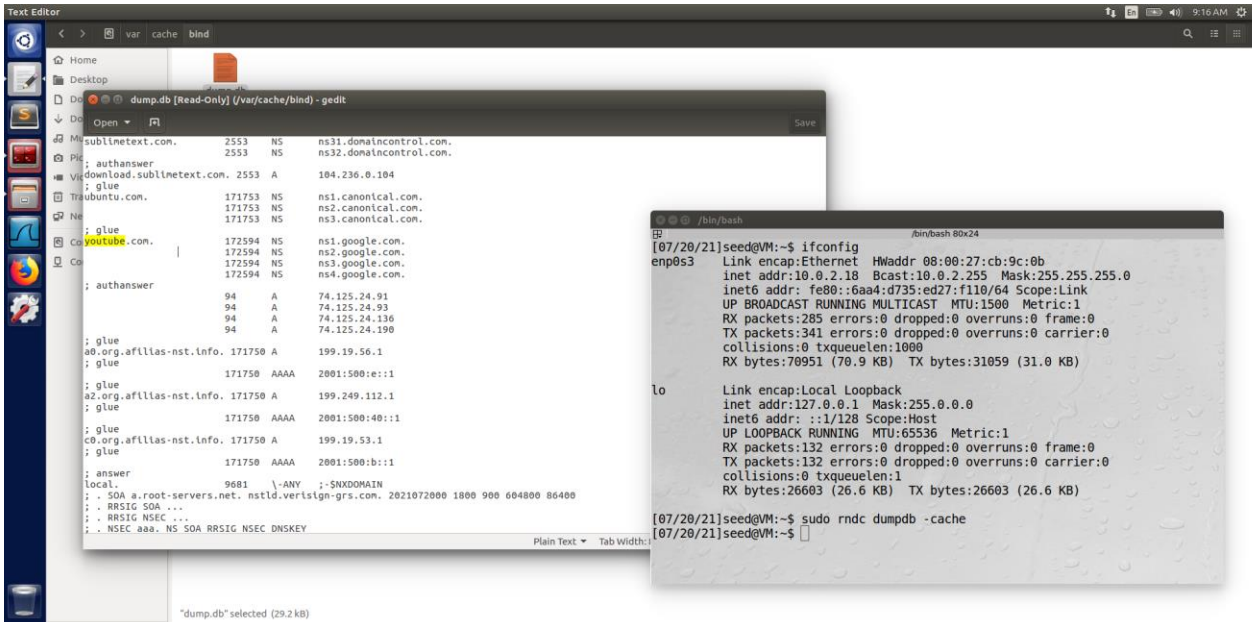
We can see the server is flooded with garbage domain-name queries from IP address 10.0.2.20

Let's check the client machine to see if we can still get response using 'dig' command.


```
;; WHEN: Tue Jul 20 09:05:56 EDT 2021
;; MSG SIZE  rcvd: 300

[07/20/21]seed@VM:~$ dig youtube.com

; <<>> DiG 9.10.3-P4-Ubuntu <<>> youtube.com
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 26416
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 4, ADDITIONAL: 9

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

;; ANSWER SECTION:
youtube.com.            300     IN      A       74.125.24.91
youtube.com.            300     IN      A       74.125.24.136
youtube.com.            300     IN      A       74.125.24.93
youtube.com.            300     IN      A       74.125.24.190

;; AUTHORITY SECTION:
youtube.com.            172800  IN      NS      ns3.google.com.
youtube.com.            172800  IN      NS      ns2.google.com.
youtube.com.            172800  IN      NS      ns4.google.com.
youtube.com.            172800  IN      NS      ns1.google.com.

;; ADDITIONAL SECTION:
ns1.google.com.         172083  IN      A       216.239.32.10
ns1.google.com.         172083  IN      AAAA    2001:4860:4802:32::a
ns2.google.com.         172083  IN      A       216.239.34.10
ns2.google.com.         172083  IN      AAAA    2001:4860:4802:34::a
ns3.google.com.         172083  IN      A       216.239.36.10
ns3.google.com.         172083  IN      AAAA    2001:4860:4802:36::a
ns4.google.com.         172083  IN      A       216.239.38.10
```


```
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 26416
;; flags: qr rd ra; QUERY: 1, ANSWER: 4, AUTHORITY: 4, ADDITIONAL: 9

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

;; ANSWER SECTION:
youtube.com.            300     IN      A       74.125.24.91
youtube.com.            300     IN      A       74.125.24.136
youtube.com.            300     IN      A       74.125.24.93
youtube.com.            300     IN      A       74.125.24.190

;; AUTHORITY SECTION:
youtube.com.            172800  IN      NS      ns3.google.com.
youtube.com.            172800  IN      NS      ns2.google.com.
youtube.com.            172800  IN      NS      ns4.google.com.
youtube.com.            172800  IN      NS      ns1.google.com.

;; ADDITIONAL SECTION:
ns1.google.com.         172083  IN      A       216.239.32.10
ns1.google.com.         172083  IN      AAAA    2001:4860:4802:32::a
ns2.google.com.         172083  IN      A       216.239.34.10
ns2.google.com.         172083  IN      AAAA    2001:4860:4802:34::a
ns3.google.com.         172083  IN      A       216.239.36.10
ns3.google.com.         172083  IN      AAAA    2001:4860:4802:36::a
ns4.google.com.         172083  IN      A       216.239.38.10
ns4.google.com.         172083  IN      AAAA    2001:4860:4802:38::a

;; Query time: 590 msec
;; SERVER: 10.0.2.18#53(10.0.2.18)
;; WHEN: Tue Jul 20 09:11:43 EDT 2021
;; MSG SIZE  rcvd: 359

[07/20/21]seed@VM:~$
```

So, we can see that we are still getting response in reasonable time despite the DoS attack. Let’s stop the attack and check server’s DNS cache.



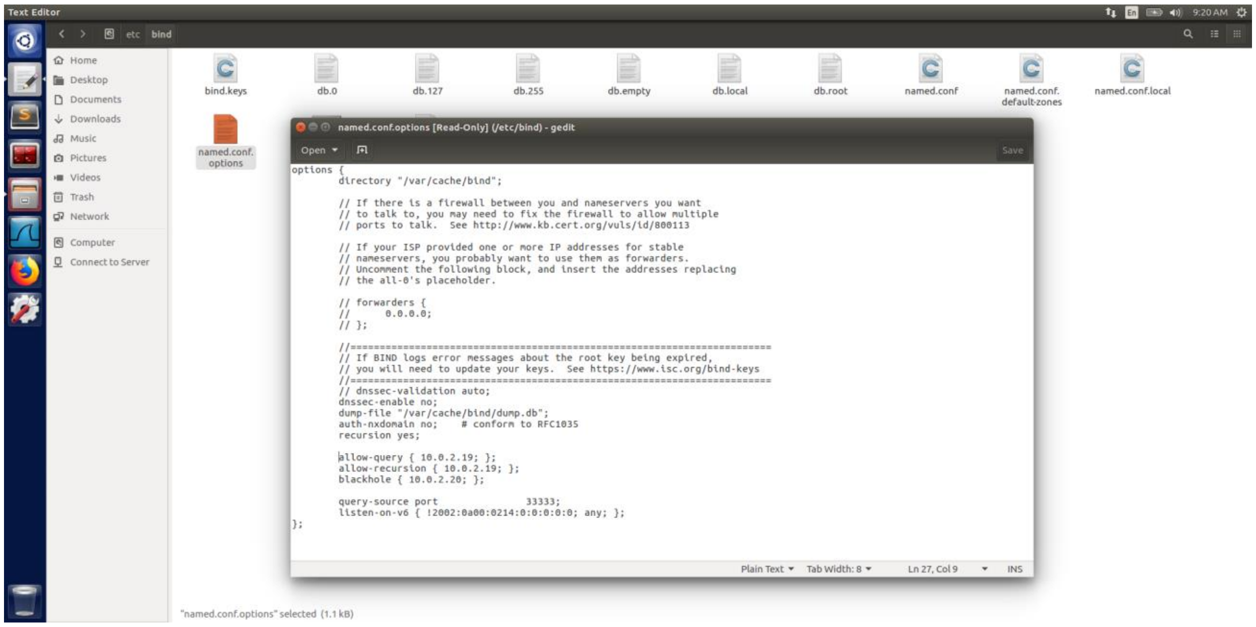
```
sublimetext.com.        2553    NS      ns31.domaincontrol.com.
                        2553    NS      ns32.domaincontrol.com.
; authanswer
download.sublimetext.com. 2553  A       104.236.0.104
; glue
ubuntu.com.             171753  NS      ns1.canonical.com.
                        171753  NS      ns2.canonical.com.
                        171753  NS      ns3.canonical.com.
; glue
youtube.com.            172594  NS      ns1.google.com.
                        172594  NS      ns2.google.com.
                        172594  NS      ns3.google.com.
                        172594  NS      ns4.google.com.
; authanswer
                        94      A       74.125.24.91
                        94      A       74.125.24.93
                        94      A       74.125.24.136
                        94      A       74.125.24.190
; glue
a0.org.afilias-nst.info. 171750 A       199.19.56.1
; glue
                        171750  AAAA    2001:500:e::1
; glue
a2.org.afilias-nst.info. 171750 A       199.249.112.1
; glue
                        171750  AAAA    2001:500:40::1
; glue
c0.org.afilias-nst.info. 171750 A       199.19.53.1
; glue
                        171750  AAAA    2001:500:b::1
; answer
local.                  9681    \-ANY   ;-$NXDOMAIN
; . SOA a.root-servers.net. nstld.verisign-grs.com. 2021072000 1800 900 604800 86400
; . RRSIG SOA ...
; . RRSIG NSEC ...
; . NSEC aaa. NS SOA RRSIG NSEC DNSKEY
```

```
[07/20/21]seed@VM:~$ ifconfig
enp0s3    Link encap:Ethernet  HWaddr 08:00:27:cb:9c:0b
          inet addr:10.0.2.18  Bcast:10.0.2.255  Mask:255.255.255.0
          inet6 addr: fe80::6aa4:d735:ed27:f110/64 Scope:Link
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:285 errors:0 dropped:0 overruns:0 frame:0
          TX packets:341 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:70951 (70.9 KB)  TX bytes:31059 (31.0 KB)

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          inet6 addr: ::1/128 Scope:Host
          UP LOOPBACK RUNNING  MTU:65536  Metric:1
          RX packets:132 errors:0 dropped:0 overruns:0 frame:0
          TX packets:132 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1
          RX bytes:26603 (26.6 KB)  TX bytes:26603 (26.6 KB)

[07/20/21]seed@VM:~$ sudo rndc dumpdb -cache
[07/20/21]seed@VM:~$ 
```

We can see the cache is unaffected by the DoS attack and thus no NXDOMAIN entries. Its size is now only 29.2 KB.

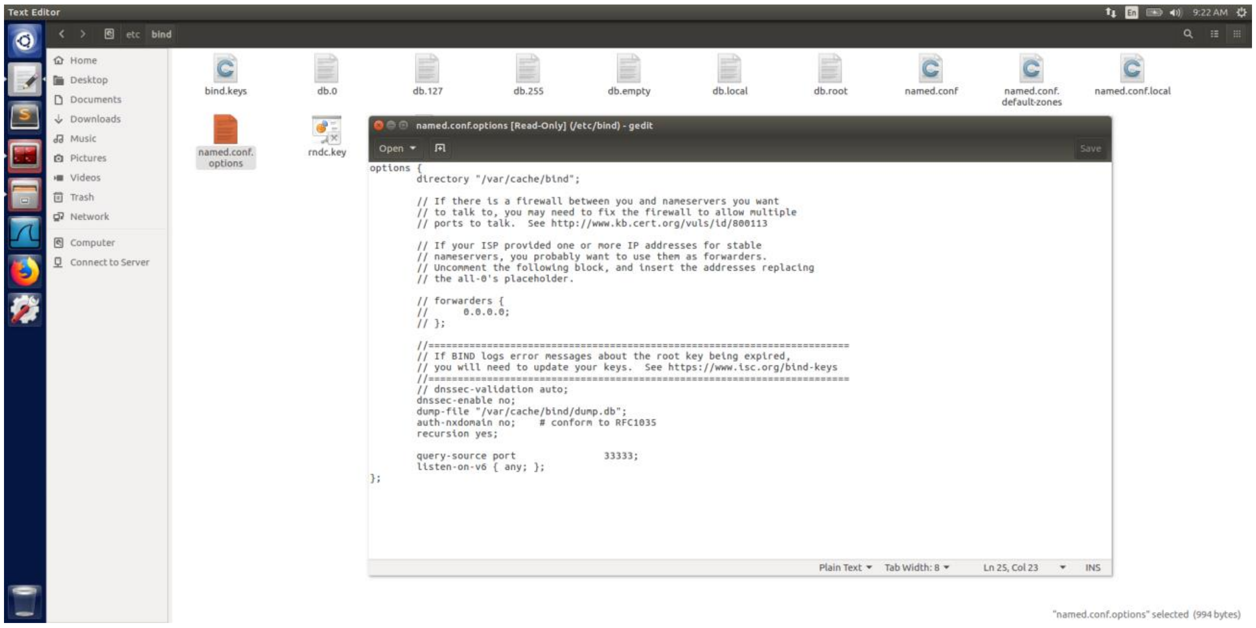
So, from the above observation we can say that the DoS attack is successfully prevented.

To achieve this, I had to make some changes in the /etc/bind/named.conf.options file. Below both normal and secure server's file snapshot is given for comparison.



```
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
        recursion yes;

        allow-query { 10.0.2.19; };
        allow-recursion { 10.0.2.19; };
        blackhole { 10.0.2.20; };

        query-source port               33333;
        listen-on-v6 { !2002:0a00:0214:0:0:0:0:0; any; };
};
```

This is from the **Secure DNS Server** machine.



```
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
        recursion yes;

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

This is from the **Normal DNS Server** machine.

Demonstration Video Link: https://youtu.be/1hZ89F8APoE

References:

- https://datatracker.ietf.org/doc/html/rfc1035
- https://w3.cs.jmu.edu/kirkpams/OpenCSF/Books/csf/html/index.html
- Wenliang Du - Computer & Internet Security_ A Hands-on Approach-Wenliang Du (2019)