Computer Networks Lab

UE19CS256

Week 4

Name: Sreenath Saikumar

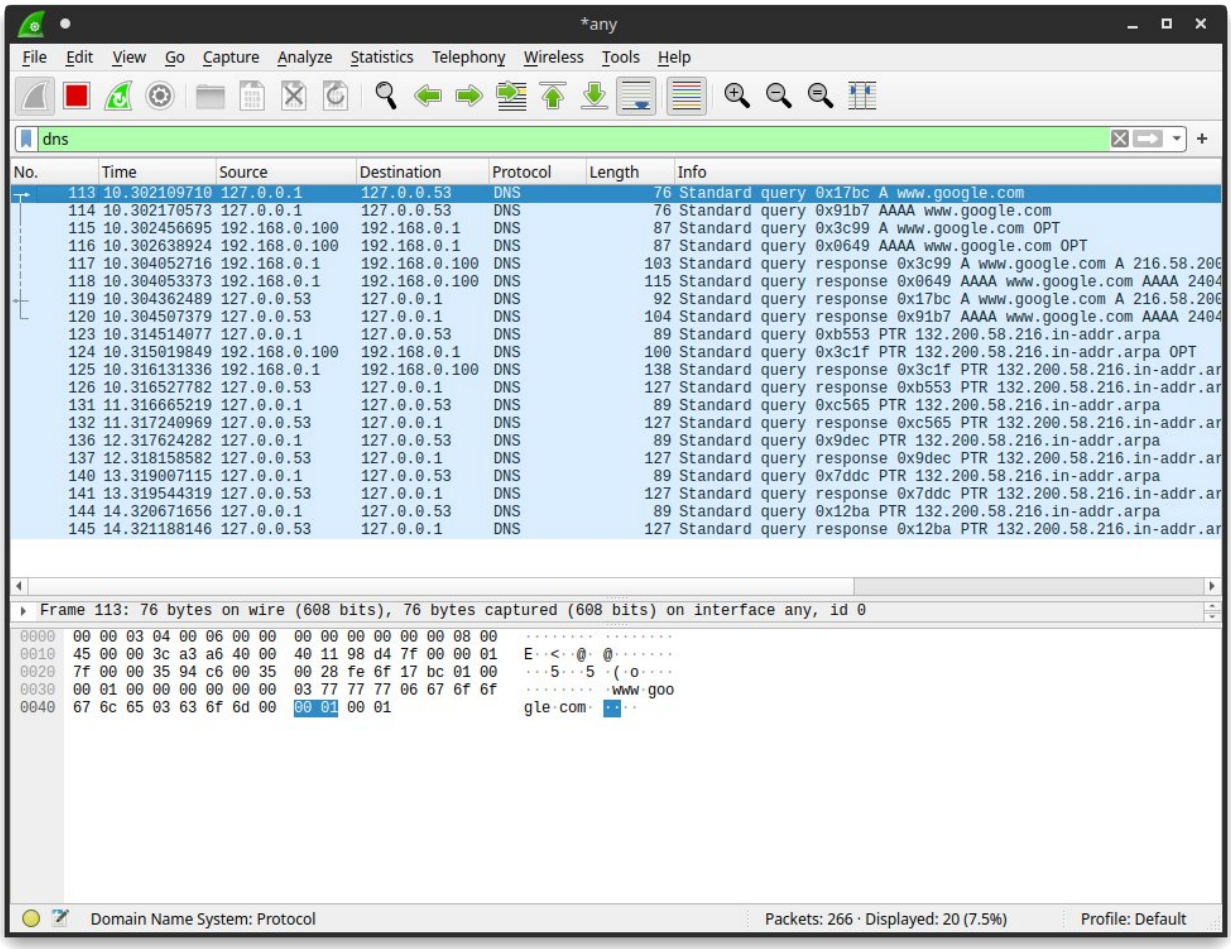
Semester: 4 Section :G

SRN: PES2UG19CS406

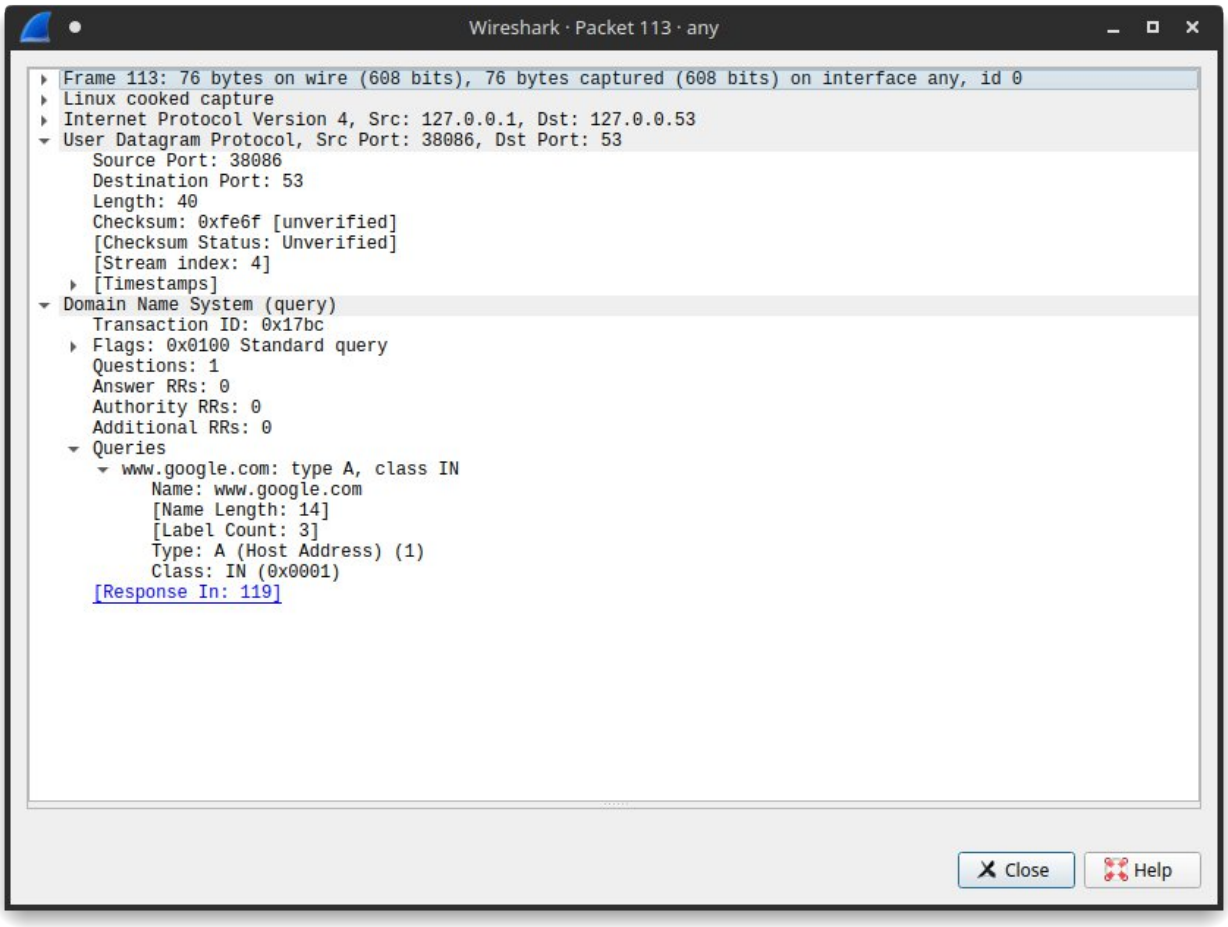
Date: 21/02/2021

1. Pinging Using default DNS

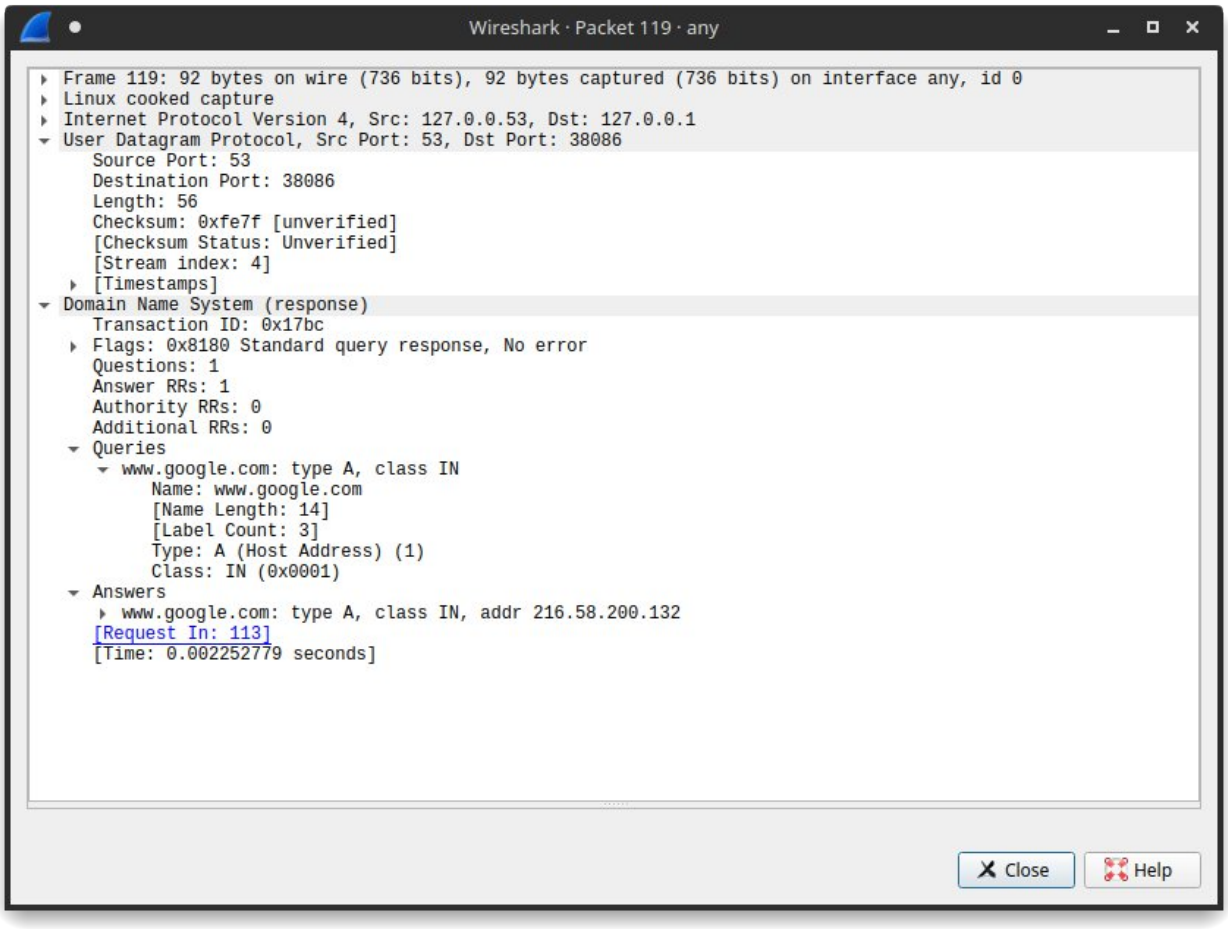
We use the `ping` command to connect to `www.google.com` and use Wireshark to capture the packets. The IP address of the default/ local DNS server is `127.0.0.53`.



Wireshark capture


Wireshark · Packet 113 · any
Frame 113: 76 bytes on wire (608 bits), 76 bytes captured (608 bits) on interface any, id 0
Linux cooked capture
Internet Protocol Version 4, Src: 127.0.0.1, Dst: 127.0.0.53
User Datagram Protocol, Src Port: 38086, Dst Port: 53
Source Port: 38086
Destination Port: 53
Length: 40
Checksum: 0xfe6f [unverified]
[Checksum Status: Unverified]
[Stream index: 4]
[Timestamps]
Domain Name System (query)
Transaction ID: 0x17bc
Flags: 0x0100 Standard query
Questions: 1
Answer RRs: 0
Authority RRs: 0
Additional RRs: 0
Queries
www.google.com: type A, class IN
Name: www.google.com
[Name Length: 14]
[Label Count: 3]
Type: A (Host Address) (1)
Class: IN (0x0001)
[Response In: 119]
Close
Help

DNS Query



DNS Query Response

The query is type A (authoritative). The response has an answers field which contains the IP address of the destination (`www.google.com) 216.58.200.132.`

2. Setting up a local DNS Server

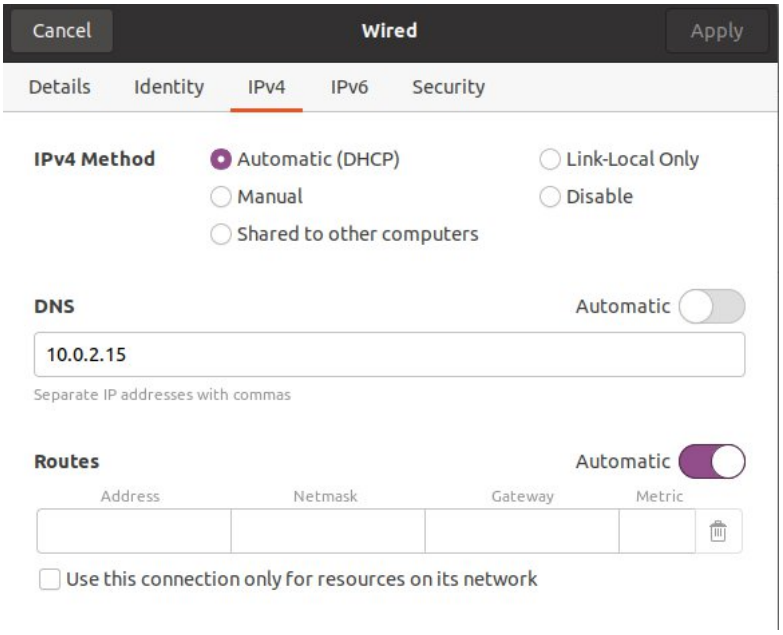
a) Configure Client Machine

The IP address of the server is `192.168.0.100` and the IP address of the client used for this task is `192.168.0.104.`

We can add a custom DNS Server by changing the resolver configuration in the `/etc/resolvconf/resolv.conf.d/head.`

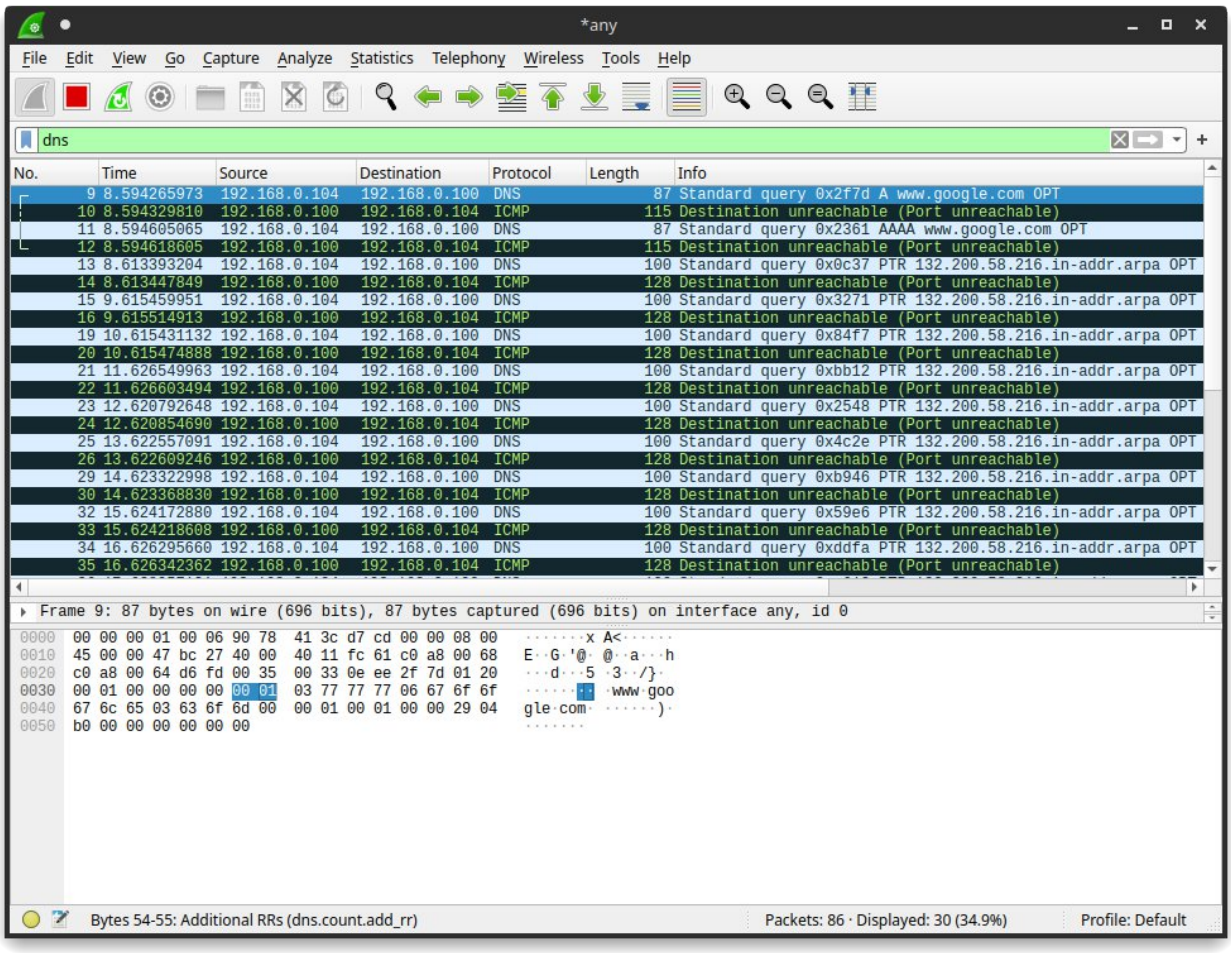
We also change the DNS Server manually in the IPv4 Settings GUI.

To apply the changes, use the command `sudo resolvconf -u`



Here, the IP is 10.0.2.15 but prior to this, it was `192.168.0.104`. This change was made since I went from using 2 physical machines to using 2 VM's.

Now, we ping `www.google.com` again and observe the wireshark capture.



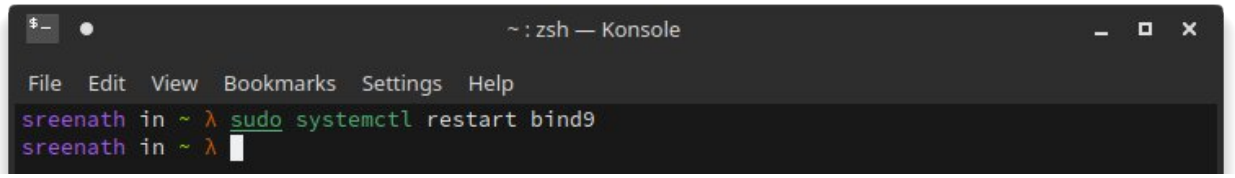
We get a `Destination Unreachable Error` since there is no record of Google's IP on the server i.e. our server machine and will try to fallback onto the default `127.0.0.53 IP.`

b) Setting up a Local DNS Server

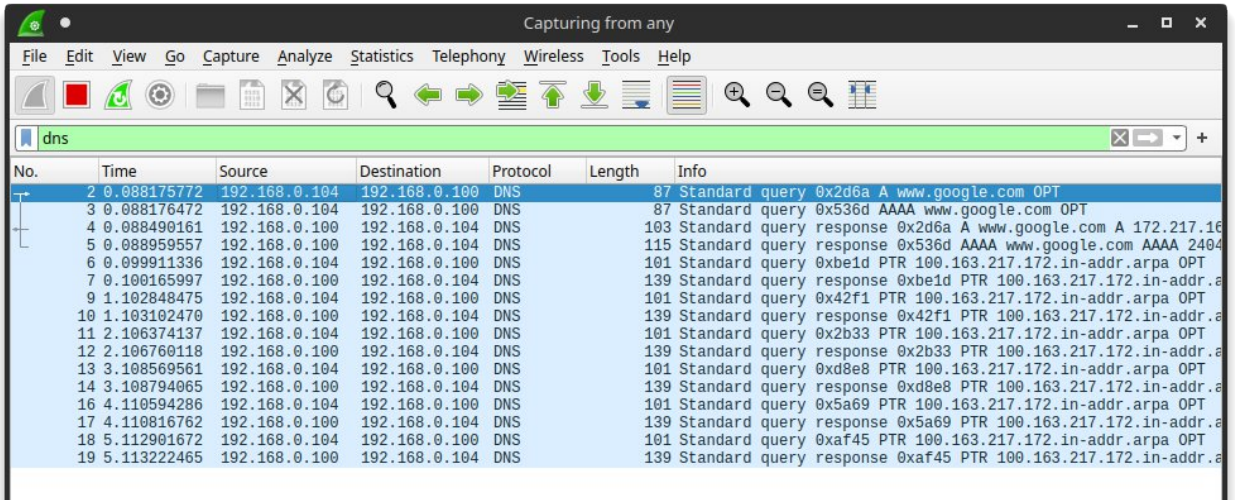
We use the `bind9` server as the DNS server on our local server machine. Install it using the command `sudo <packagemanager> install bind9.`

We can also edit the `named.conf.options` file in the `/etc/bind/` directory to specify where we want to store our DNS dump files but the default file stored in the `/var/cache/bind/named_dump.db` will be used.

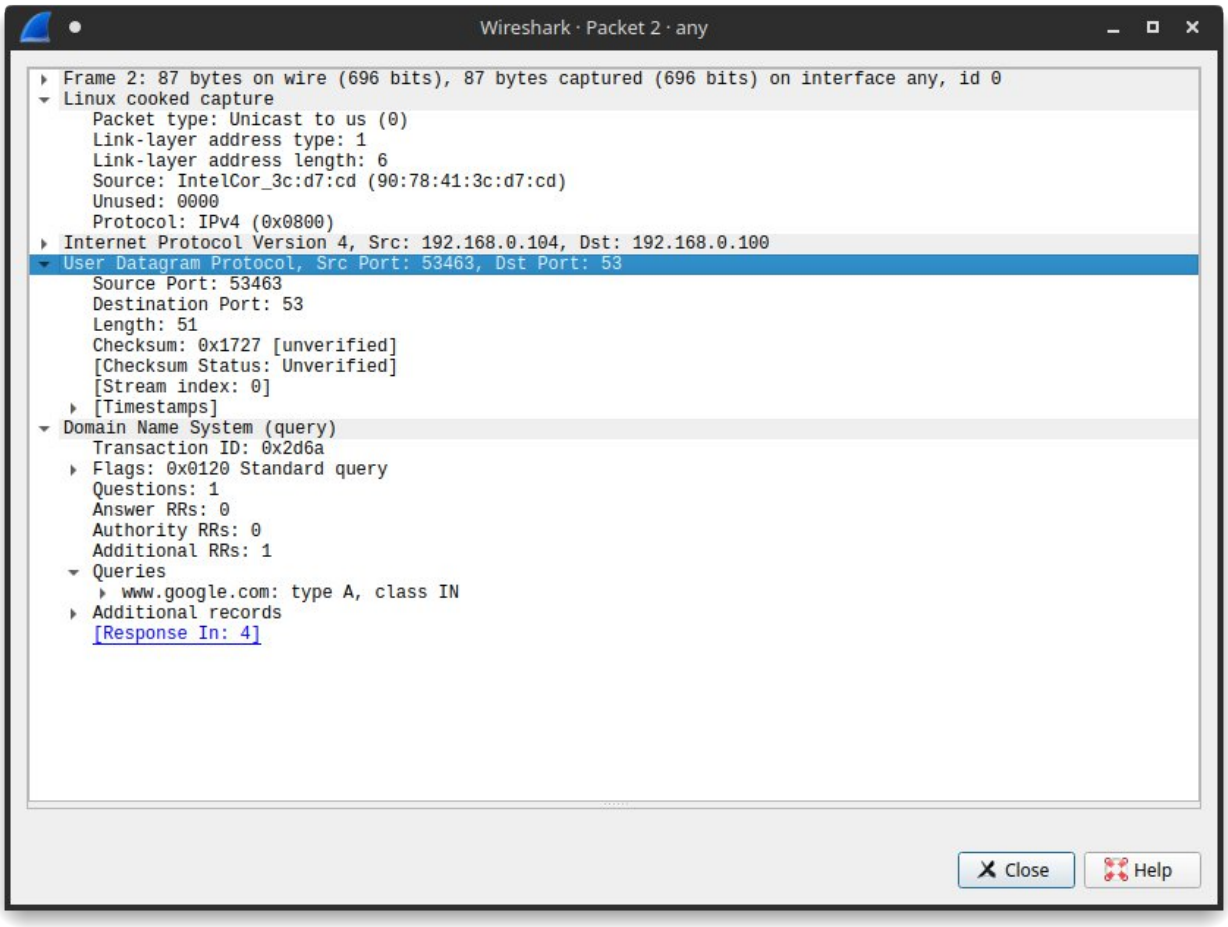
Now restart the server using `sudo systemctl restart bind9` or `sudo service bind9 restart.`



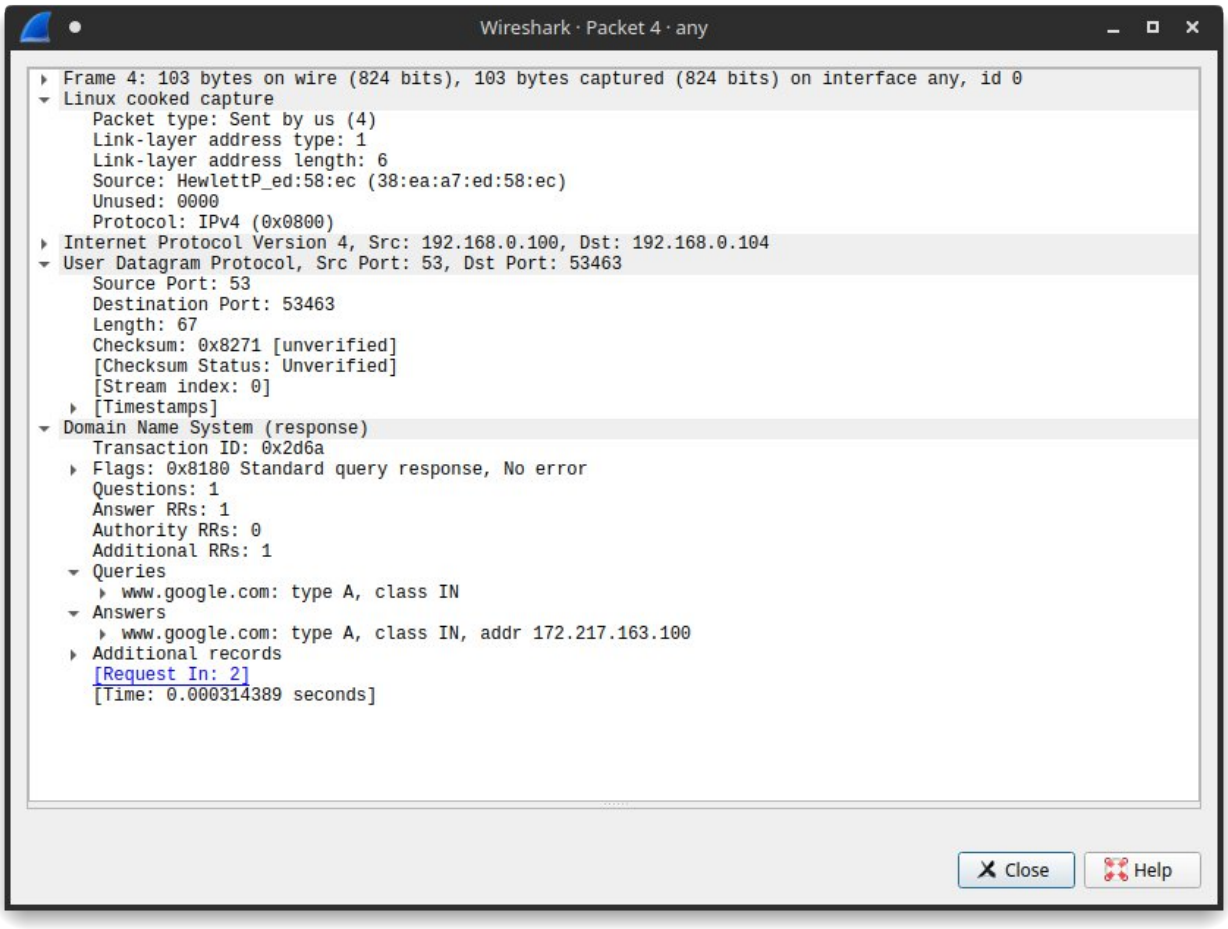
Now, we ping `www.google.com` again from our client machine(192.168.0.104) and observe the Wireshark capture.



We now see that our client machine clearly uses the local DNS Server that we have set up with A type records.

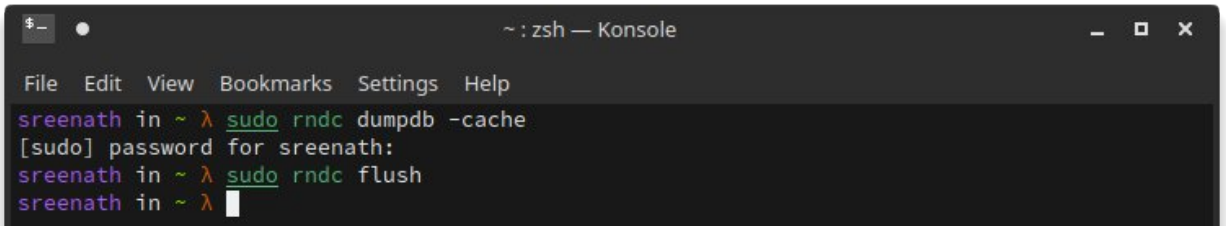

Wireshark · Packet 2 · any
Frame 2: 87 bytes on wire (696 bits), 87 bytes captured (696 bits) on interface any, id 0
Linux cooked capture
Packet type: Unicast to us (0)
Link-layer address type: 1
Link-layer address length: 6
Source: IntelCor_3c:d7:cd (90:78:41:3c:d7:cd)
Unused: 0000
Protocol: IPv4 (0x0800)
Internet Protocol Version 4, Src: 192.168.0.104, Dst: 192.168.0.100
User Datagram Protocol, Src Port: 53463, Dst Port: 53
Source Port: 53463
Destination Port: 53
Length: 51
Checksum: 0x1727 [unverified]
[Checksum Status: Unverified]
[Stream index: 0]
[Timestamps]
Domain Name System (query)
Transaction ID: 0x2d6a
Flags: 0x0120 Standard query
Questions: 1
Answer RRs: 0
Authority RRs: 0
Additional RRs: 1
Queries
www.google.com: type A, class IN
Additional records
[Response In: 4]
Close
Help

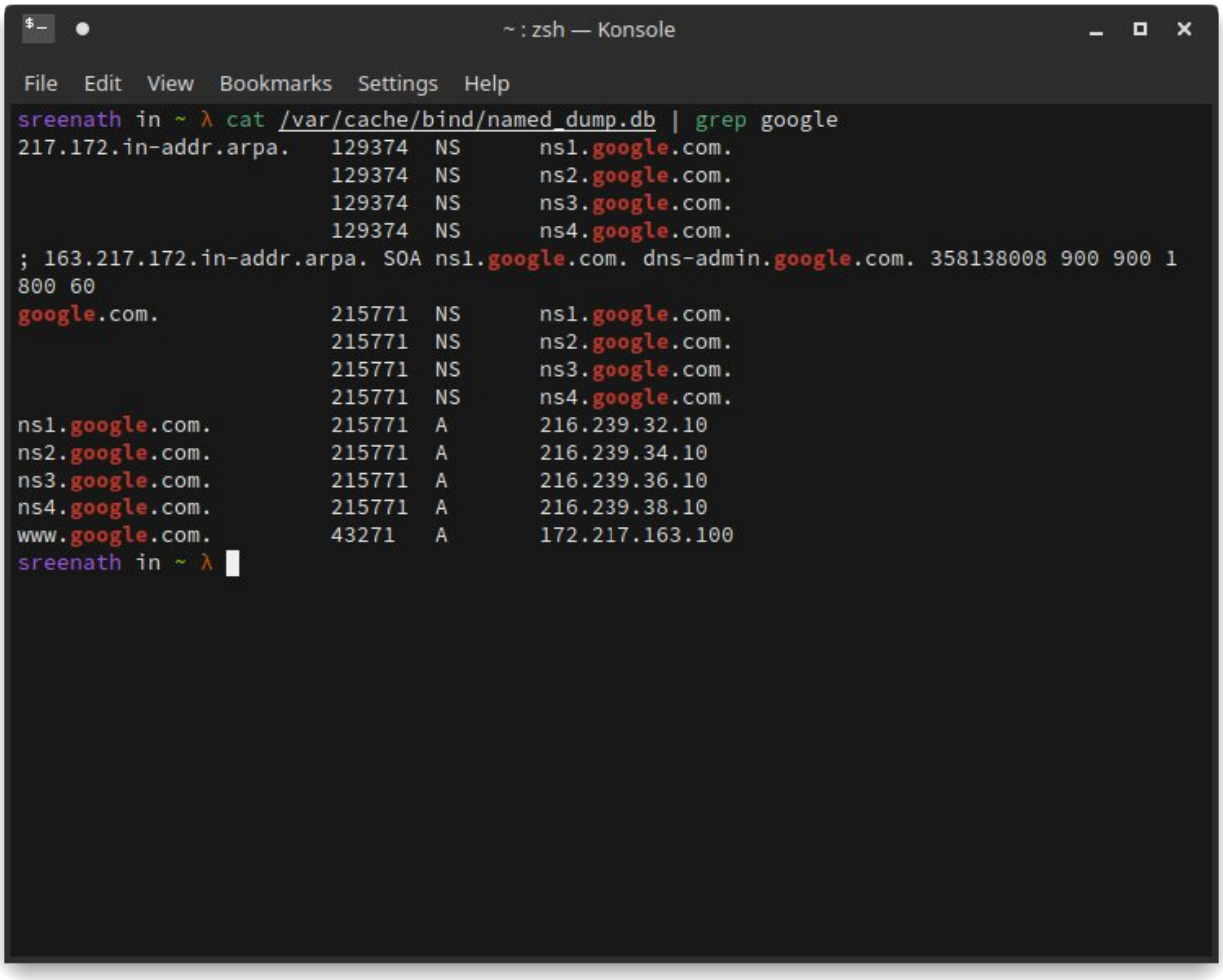
DNS Query



DNS Response

We now use the commands `sudo rndc dumpdb -cache & sudo rndc flush` to dump the cache to the `named_dump.db` file and then flush the DNS cache.





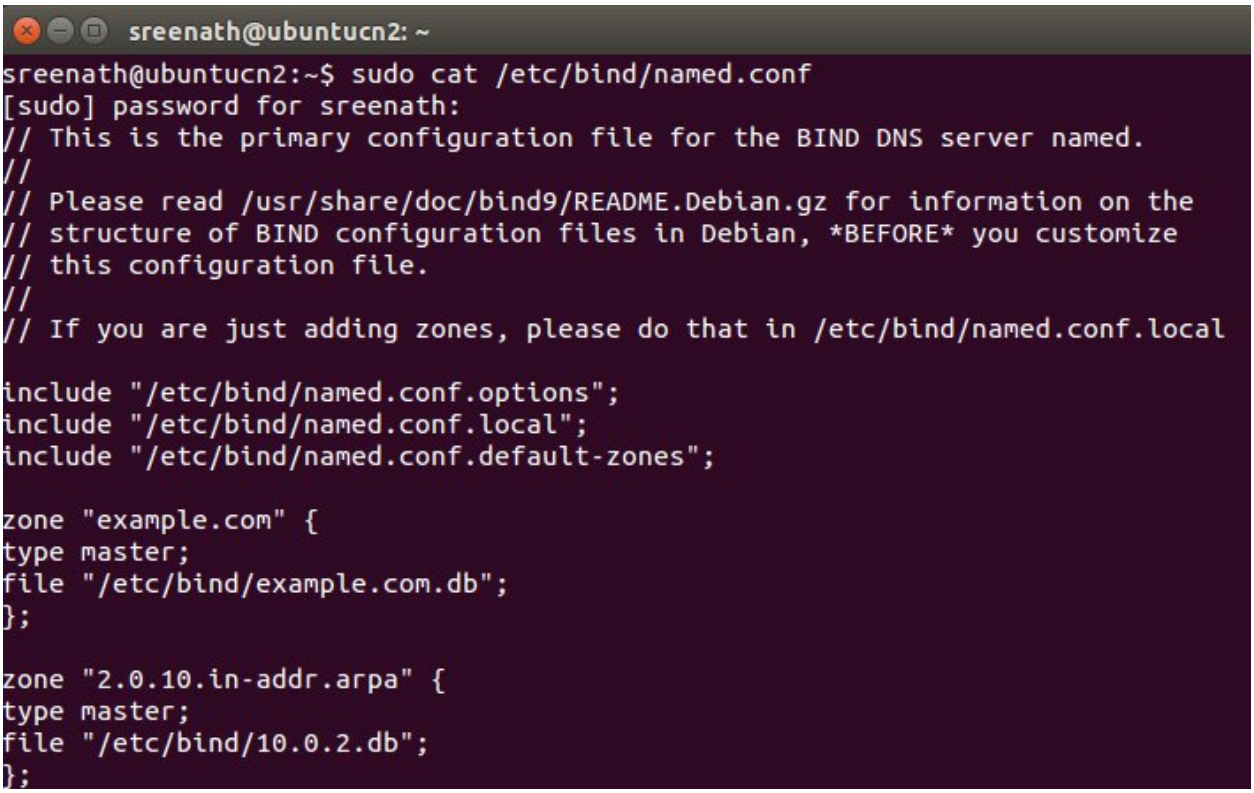
On grepping for google in the DNS cachedump

3. Setting up an Authoritative Nameserver for the example.com Domain

a) Creating a Zone in the local DNS Server

We add 2 zone entries to the `/etc/bind/named.conf` files on the server to create 2 zones. The 1st zone corresponds to the forward lookup (from hostname to IP) and the 2nd zone is for reverse lookup (from IP to hostname).

The IP's used from now onwards are `10.0.2.15` for the Server and `10.0.2.4` for the Client machine as they are running on VM's.



10.0.2.0 is the subnet mask of our IP address, hence the reverse of that is used in the reverse lookup.

Forward and Reverse Lookup

The forward lookup file is located in `/etc/bind/example.com.db.`

The @ symbol is used to specify the origin, which is example.com in this case. There are 7 resource records in the lookup file, including an SOA (Start of Authority) RR, an NS(Name Server) RR, an MX(Mail Exchanger) RR and 4 A(host Addresses) RR's.

example.com.db file

The reverse lookup file is located in `/etc/bind/10.0.2.db.`

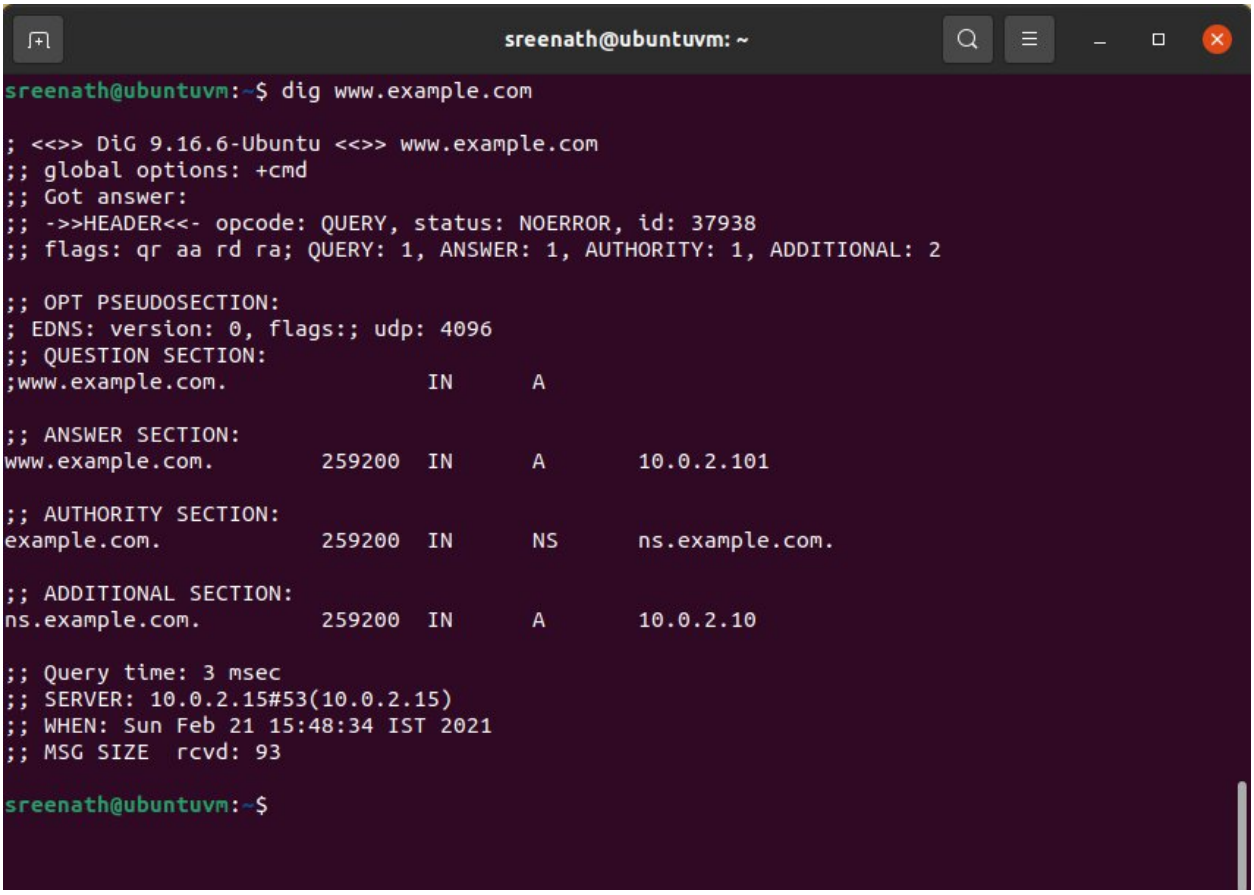
This is used to translate IP addresses to hostnames for a given domain.

For each IP address defined in the forward lookup file, a corresponding hostname is specified here. The record type here is PTR or DNS Pointer Record.

Once the files have been created, restart the `bind9` server.

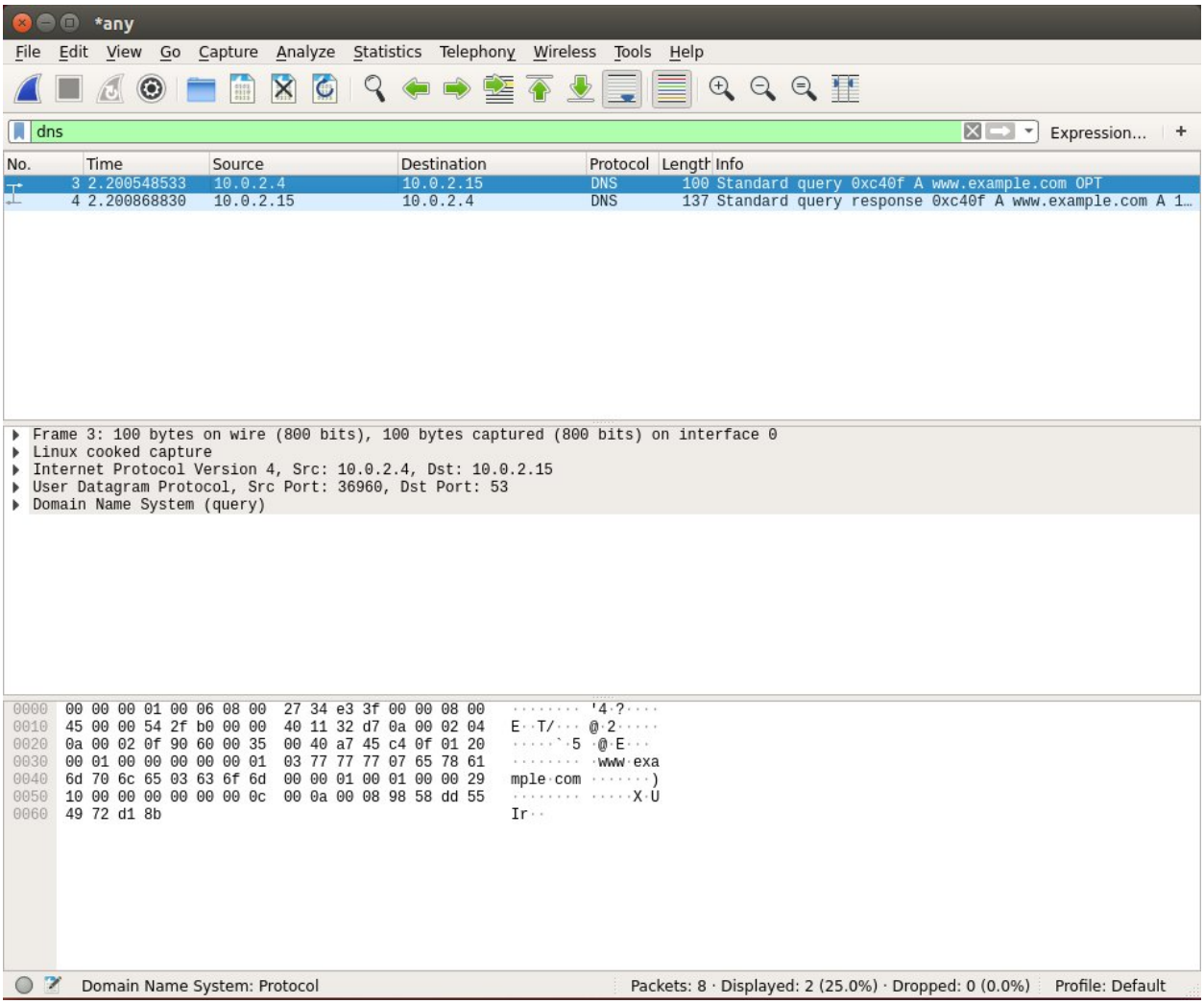
Testing the DNS Server using DIG

On the client machine, use the `dig` command which is used to lookup name servers stored in the `/etc/resolv.conf` file. We use Wireshark on the server to capture the packets received when we run the command `dig www.example.com`

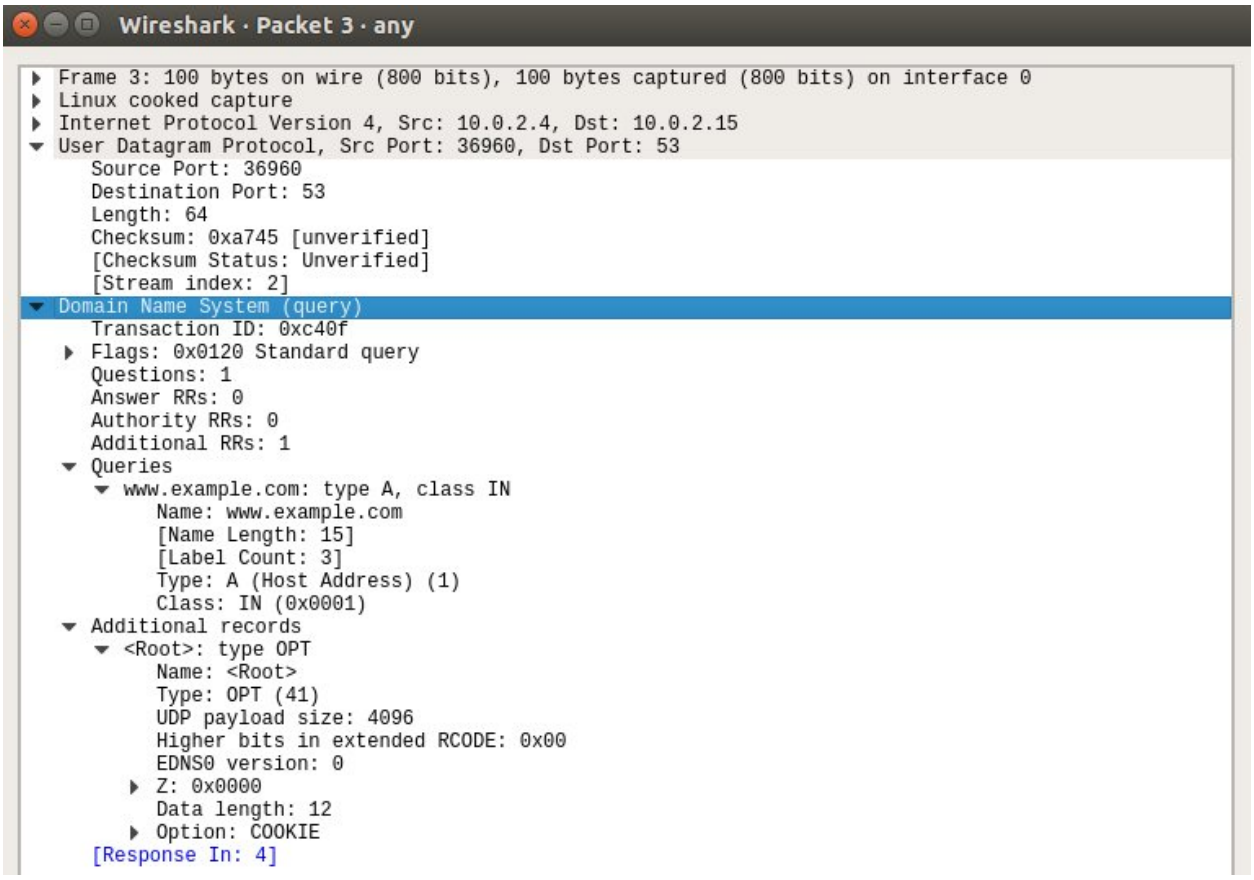


`dig www.example.com`

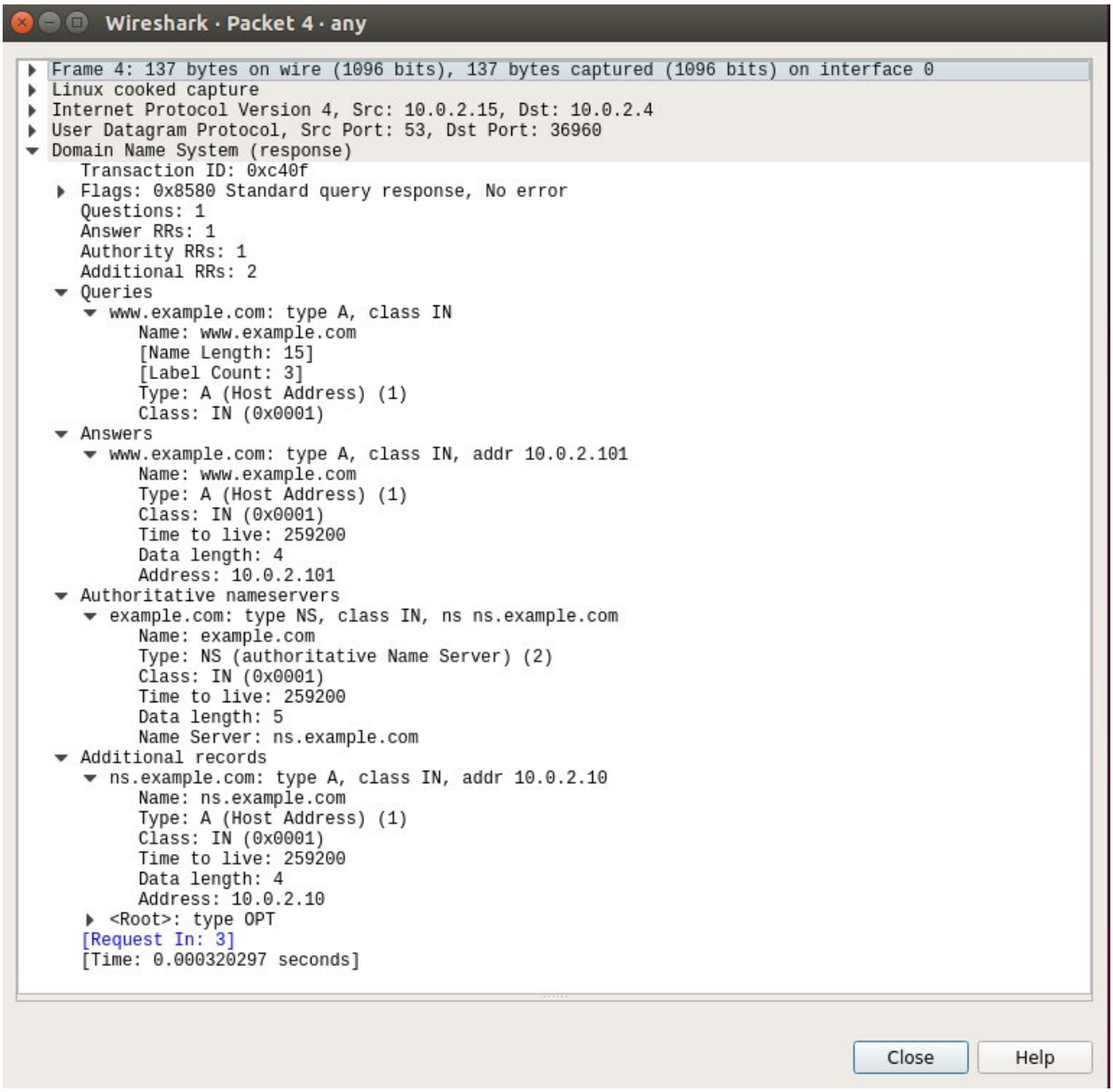
On inspecting the output of the `dig` command, we can see that the ANSWER section contains the DNS mapping of our query which is `10.0.2.101,` which corresponds to the IP that we had specified in the lookup files.



Wireshark Capture


Wireshark · Packet 3 · any
Frame 3: 100 bytes on wire (800 bits), 100 bytes captured (800 bits) on interface 0
Linux cooked capture
Internet Protocol Version 4, Src: 10.0.2.4, Dst: 10.0.2.15
User Datagram Protocol, Src Port: 36960, Dst Port: 53
Source Port: 36960
Destination Port: 53
Length: 64
Checksum: 0xa745 [unverified]
[Checksum Status: Unverified]
[Stream index: 2]
Domain Name System (query)
Transaction ID: 0xc40f
Flags: 0x0120 Standard query
Questions: 1
Answer RRs: 0
Authority RRs: 0
Additional RRs: 1
Queries
www.example.com: type A, class IN
Name: www.example.com
[Name Length: 15]
[Label Count: 3]
Type: A (Host Address) (1)
Class: IN (0x0001)
Additional records
<Root>: type OPT
Name: <Root>
Type: OPT (41)
UDP payload size: 4096
Higher bits in extended RCODE: 0x00
EDNS0 version: 0
Z: 0x0000
Data length: 12
Option: COOKIE
[Response In: 4]

DNS Query



DNS Query Response

Unfortunately, on checking the DNS cache after this task, there was only garbage information, so it has been omitted.

Questions:

1. Locate the DNS query and response messages. Are they sent over TCP or UDP?

We notice from the above packet captures that they are sent over UDP.

2. What is the destination port for the DNS query message? What is the source port of the DNS response message?

Both ports are the same. The port used for DNS is 53.

3. To what IP address is the DNS query message sent? Using ipconfig, determine the IP address of your local DNS server. Are these 2 IP's the same?

The DNS query is sent to 10.0.2.15 which is what is configured.

4. Examine the DNS query message. What 'Type' of DNS query is it? Does the query message contain any 'answers'?

The DNS query is of type A since it is authoritative. There is no answer section since queries won't have answers.

5. Examine the DNS response message. How many 'answers' are provided? What do each of these answers contain?

The DNS response contained 1 RR: an A type RR for `www.google.com` with its IP address and canonical hostname.

6. Consider the subsequent TCP SYN packet sent by your host. Does the destination IP address of the SYN packet correspond to any of the IP addresses provided in the DNS response message?

The destination IP address corresponds to the IP address of the hostname `www.google.com` obtained from the response message.