CS 315: Computer Networks Lab
Spring 2022-23, IIT Dharwad
Assignment-4

Wireshark Lab: DNS

January 24, 2023

Lab Instructions

- Please leave your bags on the Iron shelf near the SP16 entrance.
- Login to the Ubuntu OS on your machine. The login credentials are as follows:
  - Username: user
  - Password: 123456
- Mark your attendance in the attendance sheet before leaving the lab.
- Handle the lab resources with utmost care.
- Please go through the following exercises in today's lab.
- It is recommended that you complete all the following exercises during the lab slot itself.
- If you face any difficulties, please feel free to seek help online or from your peers or TAs.
- After finishing all exercises, please carry your solutions with you (via email/pen drive) for future reference, and delete the files from the desktop.

Introduction

The Domain Name System (DNS) translates hostnames to IP addresses, fulfilling a critical role in the Internet infrastructure. In this lab, we'll take a closer look at the client side of DNS. The client's role in the DNS is relatively simple – a client sends a *query* to its local DNS server, and receives a *response* back.

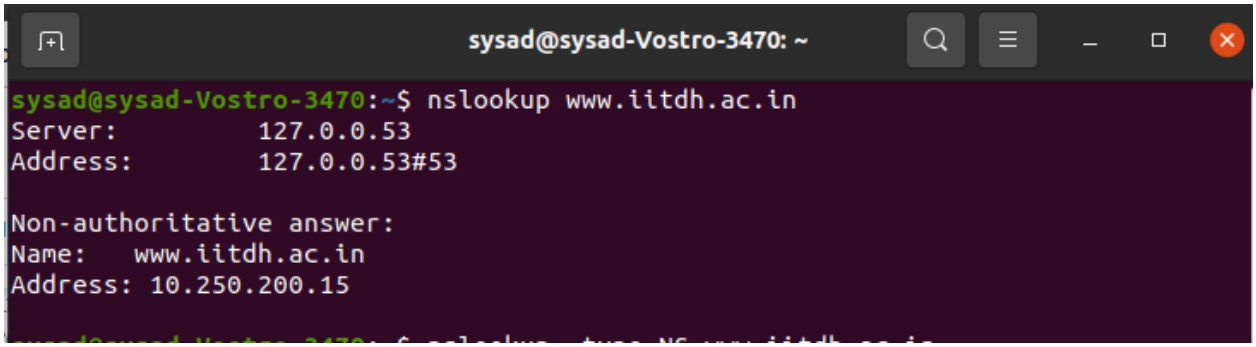
Part-1: nslookup

In this lab, we'll make extensive use of the *nslookup* tool, which is available in most Linux/Unix and Microsoft platforms today. To run *nslookup* in Linux/Unix, you just type the *nslookup* command on the terminal. To run it in Windows, open the Command Prompt and run *nslookup* on the command line.

In its most basic operation, *nslookup* tool allows the host running the tool to query any specified DNS server for a DNS record. The queried DNS server can be a root DNS server, a top-level-domain DNS server, an authoritative DNS server, or an intermediate DNS server (see the textbook for definitions of these terms). To accomplish this task, *nslookup* sends a DNS query to the specified DNS server, receives a DNS reply from that same DNS server, and displays the result.

Consider the first command:

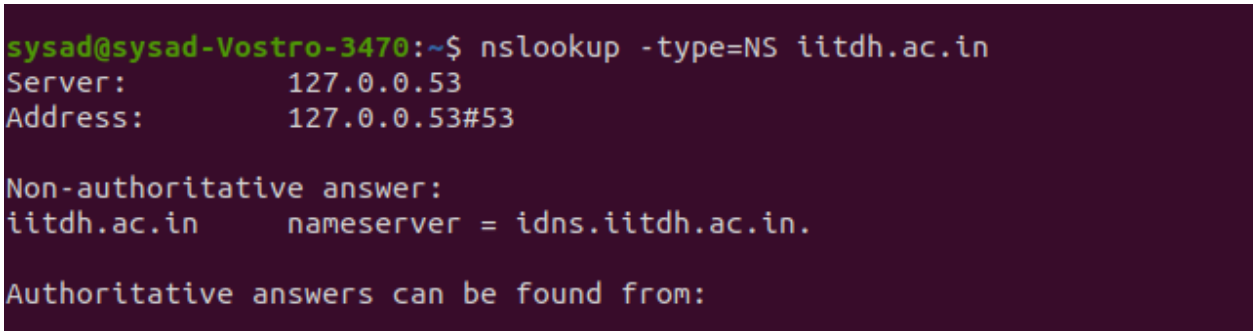
nslookup www.iitdh.ac.in



In other words, this command is saying "please send me the IP address for the host www.iitdh.ac.in". As shown in the screenshot, the response from this command provides two pieces of information: (1) the name and IP address of the DNS server that provides the answer; and (2) the answer itself, which is the hostname and IP address of www.iitdh.ac.in.

Now consider the second command:

nslookup –type=NS iitdh.ac.in



In this example, we have provided the option "-type=NS" and the domain "iitdh.ac.in". This causes nslookup to send a query for a type-NS record to the default local DNS server. In other words, the query is saying, "please send me the host names of the DNS for iitdh.ac.in". (When the –type option is not used, nslookup uses the default, which is to query for type A records.) The answer, displayed in the above screenshot, first indicates the DNS server that is providing the answer (which is the default local DNS server) along with iitdh nameservers, nslookup also indicates that the answer is "non-authoritative," meaning that this answer came from the cache of some server rather than from an authoritative IITDH DNS server.

Consider the third command:

nslookup google.com ns3.google.com

```
sysad@sysad-Vostro-3470:~$ nslookup google.com ns3.google.com
Server:         ns3.google.com
Address:        216.239.36.10#53

Name:   google.com
Address: 172.217.31.206
Name:   google.com
Address: 2404:6800:4007:809::200e
```

In this example, we indicate that we want the query sent to the DNS server ns3.google.com rather than to the default DNS server (dns-prime.poly.edu). Thus, the query and reply transaction takes place directly between our querying host and ns3.google.com. In this example, the DNS server ns3.google.com provides the IP address of the host google.com.

Lastly, we sometimes might be interested in discovering the name of the host associated with a given IP address, i.e., the reverse of the lookup shown below (where the host's name was known/specified and the host's IP address was returned). `nslookup` can also be used to perform this so-called "reverse DNS lookup." In Figure below, for example, we specify an IP address as the `nslookup` argument (128.119.245.12 in this example) and `nslookup` returns the host name with that address (gaia.cs.umass.edu in this example)

```
sysad@sysad-OptiPlex-7080:~$ nslookup 10.250.200.15
15.200.250.10.in-addr.arpa      name = www.iitdh.ac.in.

Authoritative answers can be found from:
```

Now that we have gone through a few illustrative examples, you are perhaps wondering about the general syntax of nslookup commands. The syntax is:

nslookup –option1 –option2 host-to-find dns-server

In general, nslookup can be run with zero, one, two or more options. And as we have seen in the above examples, the dns-server is optional as well; if it is not supplied, the query is sent to the default local DNS server.

Do the following (and write down the results):

1. Run `nslookup` to obtain the IP address of the web server for the Indian Institute of Technology Dharwad, India: www.iitdh.ac.in. What is the IP address of www.iitdh.ac.in
2. Run *nslookup* to determine the DNS servers for google.com.
3. Run *nslookup* so that one of the DNS servers obtained in Question 2 is queried for gmail.com. What is its IP address?

Part-2: The DNS cache on your computer

From the description of iterative and recursive DNS query resolution in our textbook, you might think that the local DNS server must be contacted *every* time an application needs to translate from a hostname to an IP address. That's not always true in practice!

Most hosts (e.g., your personal computer) keep a *cache* of recently retrieved DNS records (sometimes called a DNS *resolver cache*), just like many Web browsers keep a cache of objects recently retrieved by HTTP. When DNS services need to be invoked by a host, that host will first check if the DNS record needed is resident in this host's DNS cache; if the record is found, the host will not even bother to contact the local DNS server and will instead use this cached DNS record. A DNS record in a resolver cache will eventually timeout and be removed from the resolver cache, just as records cached in a local DNS server (see Figures 2.19, 2.20) will timeout.

You can also explicitly clear the records in your DNS cache. There's no harm in doing so – it will just mean that your computer will need to invoke the distributed DNS service next time it needs to use the DNS name resolution service, since it will find no records in the cache.

On a Mac computer, you can enter the following command into a terminal window to clear your DNS resolver cache:

```
sudo killall -HUP mDNSResponder
```

On Windows computer you can enter the following command at the command prompt:

```
ipconfig /flushdns
```

and on a Linux computer, enter:

```
sudo systemd-resolve --flush-caches
```

Part-3: Tracing DNS with Wireshark

Let's first capture the DNS packets that are generated by ordinary Web-surfing activity.

Do the following:

- Empty the DNS cache in your host.
- Open your browser and empty your browser cache.
- Open Wireshark and enter *dns*.
- Start packet capture in Wireshark.
- With your browser, visit the Web page: http://www.ietf.org
- Stop packet capture.

Answer the following questions:

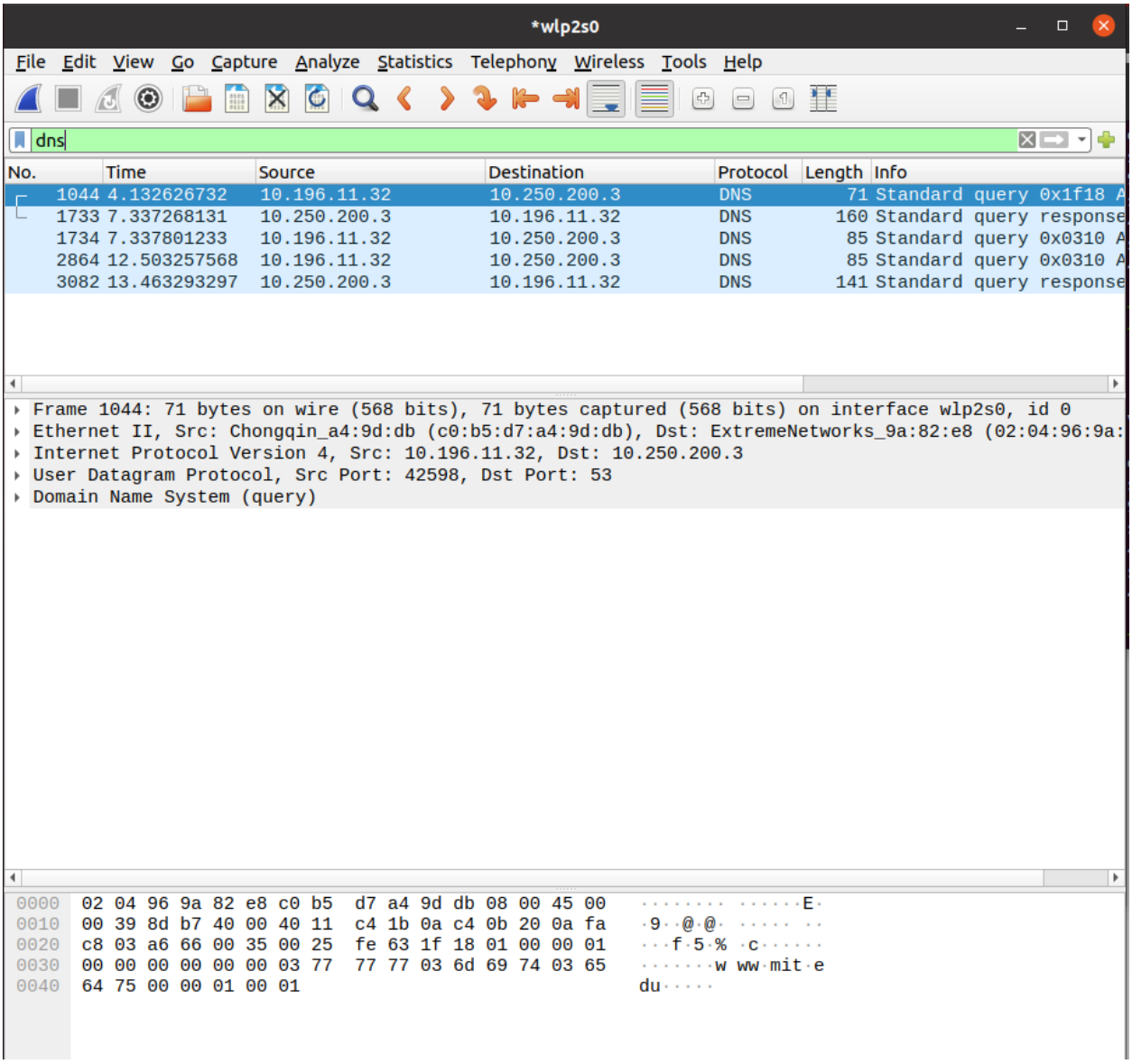
1. Locate the DNS query and response messages. Are then sent over UDP or TCP?
2. What is the destination port for the DNS query message? What is the source port of DNS response messages?
3. To what IP address is the DNS query message sent? Use *ipconfig(Windows)/dig(Linux)* to determine the IP address of your local DNS server. Are these two IP addresses the same?
4. Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"?
5. Examine the DNS response message. How many "answers" are provided? What do each of these answers contain?
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?
7. This web page contains images. Before retrieving each image, does your host issue new DNS queries?

Part-4: Wireshark and nslookup

1: Do the following:

- Start packet capture.
- In terminal do an *nslookup* on www.mit.edu
- Stop packet capture.

You should get a trace that looks something like the following:



Answer the following questions:

1. What is the destination port for the DNS query message? What is the source port of DNS response messages?
2. To what IP address is the DNS query message sent? Is this the IP address of your default local DNS server?
3. Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"?
4. Examine the DNS response message. How many "answers" are provided? What do each of these answers contain?
5. Provide a screenshot.

2: Now repeat the previous experiment, but instead issue the command:

nslookup –type=NS mit.edu

Answer the following questions

1. To what IP address is the DNS query message sent? Is this the IP address of your default local DNS server?
2. Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"?
3. Examine the DNS response message. What MIT nameservers does the response message provide? Does this response message also provide the IP addresses of the MIT namesers?
4. Provide a screenshot.

3: Now repeat the previous experiment, but instead issue the command:

nslookup gmail.com ns3.google.com

Answer the following questions

1. To what IP address is the DNS query message sent? Is this the IP address of your default local DNS server? If not, what does the IP address correspond to?
2. Examine the DNS query message. What "Type" of DNS query is it? Does the query message contain any "answers"?
3. Examine the DNS response message. How many "answers" are provided? What does each of these answers contain?
4. Provide a screenshot.

Submission Details

- Write your answers in a single doc/tex file, and submit its PDF named after your IIT Dharwad roll number, which contains all answers (with screenshots, if necessary).