COL334 Assignment:1

Sachin 2019CS10722

August 2021

1 Networking Tools

In this question we were asked to use some basic networking commands, first let me define what each command does:-

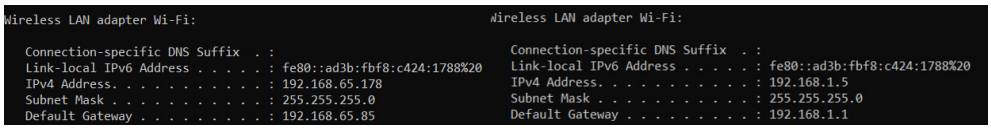
1. **ipconfig:** Gives TCP/IP Configuration details of the machine.
2. **nslookup:** Through this command we can ask DNS server the IP address of some domain.
3. **ping:** Checks whether given domain is reachable or not by sending given amount of packets(by default 4 in windows).
4. **traceroute:** Shows several details of the path taken by packets to reach the final server.

Now lets see the output of several commands asked in the question:-

1.1 IP Address of machine

I tried connecting to different service providers and noted the IP Address, here are the results:-

1. **Airtel** Wireless LAN adapter Wi-Fi: **192.168.65.178**
2. **BSNL** Wireless LAN adapter Wi-Fi: **192.168.1.5**



```
Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::ad3b:fbf8:c424:1788%20
   IPv4 Address. . . . . . . . . . . : 192.168.65.178
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.65.85
```

```
Wireless LAN adapter Wi-Fi:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::ad3b:fbf8:c424:1788%20
   IPv4 Address. . . . . . . . . . . : 192.168.1.5
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.1.1
```

Figure 1: Proof

1.2 DNS server

I tried using different DNS servers and noted the IP Address of asked domains, here are the results:-

DNS Sever	Server Address	www.google.com	www.facebook.com
multiplay.bsnl.in	218.248.114.193	142.250.192.228	157.240.239.35
dns.google	8.8.8.8	142.250.193.68	157.240.198.35
dns9.quad9.net	9.9.9.9	142.250.217.228	157.240.16.35
one.one.one.one	1.1.1.1	172.217.167.36	157.240.198.35

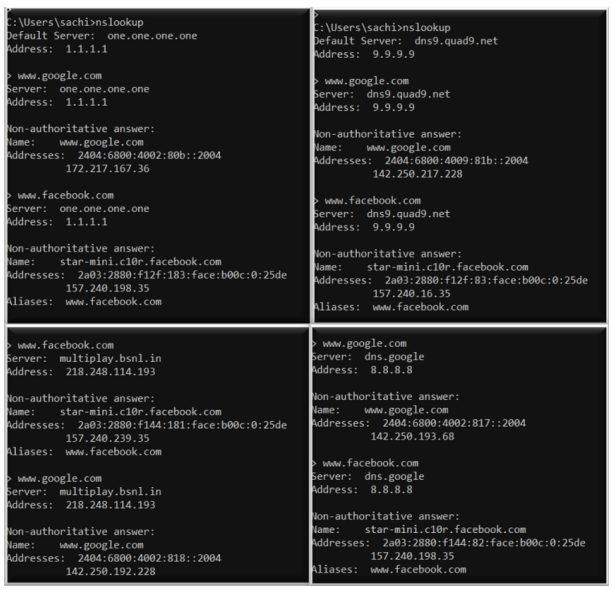


Figure 2: Proof

1.3 Ping

Following are the results of sending ping packets of different sizes to www.iitd.ac.in:-

Packet size(bytes)	RTT(avg) in ms
32	53
64	55
640	57
6400	62
14000	63

Max packet size that can be sent to www.iitd.ac.in= 17752 bytes.
Max packet size that can be sent to www.google.com= 1432 bytes.
Max packet size that can be sent to www.facebook.com= 1432 bytes.

Following are the results of sending ping packets with different TTL values to www.iitd.ac.in:-

TTL	Output
1	TTL Expired
10	TTL Expired
12	TTL Expired
13	Packets Sent
14	Packets Sent
49	Packets Sent

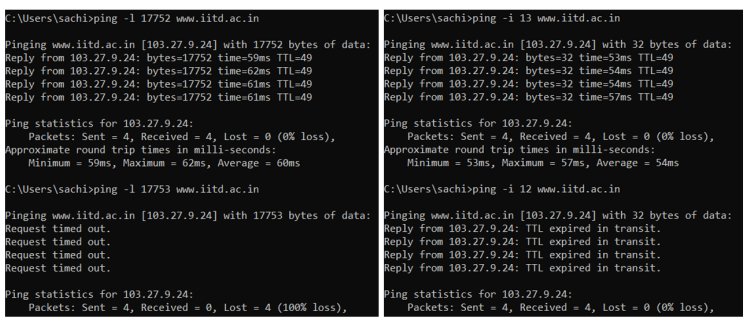


Figure 3: Proof

1.4 Traceroute

I tried traceroute on the all 3 domains once with bsnl Wifi and again with airtel. One difference that I noticed was that airtel traceroute was a bit faster than bsnl.
Although using both service providers, for any of the domains I did not get IPv6 address but if one wants to use only IPv4 address, this can be done by using command: tracert -4 <domainname> in windows.
While using airtel, first 4 IP addresses were private whereas while using bsnl, only first IP address was private.
There were some routers in the path (comapratively more in bsnl than airtel) that did not reply. There can be many reasons for it, like the router itself doesn't allow the same (for security purposes) or maybe there is some firewall blocking the request or maybe there is a lot of traffic and the router canno be reached.
One can use ICMP or TCP to overcome this problem by using command:
tracert -T <domainname> for TCP and tracert -I <domainname> for ICMP.

2 Packet Analysis

2.1 Apache: DNS

Start time = 18:31:10.634895
End Time = 18:31:10.680367
Time taken to get the IP address of http://apache.org = 0.045472 s

Note: This is the time taken for DNS request to get IP address of apache.org only, to load the site there were multiple dns requests to load different components of site like fonts.googleapi.com, youtube.com etc. Last DNS request completed at time 18:32:36.232817.

2.2 Apache: HTTP

There were total of 28 http GET requests and their corresponding responses. This tells that there are around 28 different components that constitute to make up the webpage, and for a client to display the site it has to get all those 28 components from the server. More specifically, by checking the nature of each packet we can know the structure of the webpage. Here are the details how apache.org is structured:-

Extension	Count
css	4
jpg	10
js	3
png	8
ico	1

And out of remaining 2, 1 is the first GET request of HTTP 1.1 and other is named generate_204 (since there is no extension, it might be some executable)

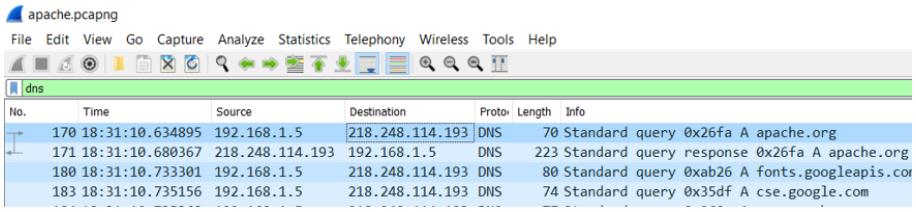
2.3 Apache: Total time

Total time taken to download the entire webpage was 1.948461 seconds.

2.4 CSE vs Apache

Yes there was one http GET request but in response I got 301 moved permanently. And when I checked the body of the response packet it was a html file that had a message to redirect to https://www.iitd.ac.in (http changed to https)

Apart from that there were no HTTP traffic like what we had in apache.org where we had http get requests for each component of the webpage. This is because once we were redirected to the new site we are no longer using the http protocol.



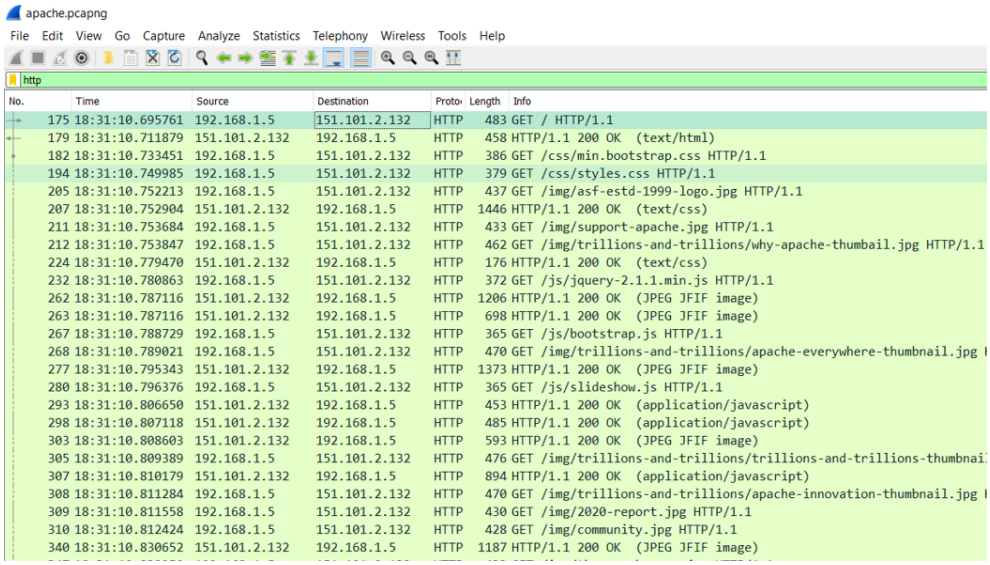
apache.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

dns

No.	Time	Source	Destination	Proto	Length	Info
170	18:31:10.634895	192.168.1.5	218.248.114.193	DNS	70	Standard query 0x26fa A apache.org
171	18:31:10.680367	218.248.114.193	192.168.1.5	DNS	223	Standard query response 0x26fa A apache.org
180	18:31:10.733301	192.168.1.5	218.248.114.193	DNS	80	Standard query 0xab26 A fonts.googleapis.co
183	18:31:10.735156	192.168.1.5	218.248.114.193	DNS	74	Standard query 0x35df A cse.google.com

Figure 4: Apache: dns



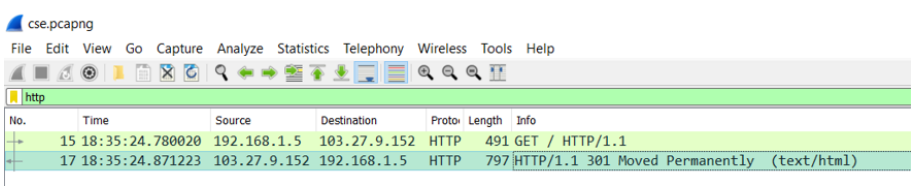
apache.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

http

No.	Time	Source	Destination	Proto	Length	Info
175	18:31:10.695761	192.168.1.5	151.101.2.132	HTTP	483	GET / HTTP/1.1
179	18:31:10.711879	151.101.2.132	192.168.1.5	HTTP	458	HTTP/1.1 200 OK (text/html)
182	18:31:10.733451	192.168.1.5	151.101.2.132	HTTP	386	GET /css/min.bootstrap.css HTTP/1.1
194	18:31:10.749985	192.168.1.5	151.101.2.132	HTTP	379	GET /css/styles.css HTTP/1.1
205	18:31:10.752213	192.168.1.5	151.101.2.132	HTTP	437	GET /img/asf-estd-1999-logo.jpg HTTP/1.1
207	18:31:10.752904	151.101.2.132	192.168.1.5	HTTP	1446	HTTP/1.1 200 OK (text/css)
211	18:31:10.753684	192.168.1.5	151.101.2.132	HTTP	433	GET /img/support-apache.jpg HTTP/1.1
212	18:31:10.753847	192.168.1.5	151.101.2.132	HTTP	462	GET /img/trillions-and-trillions/why-apache-thumbail.jpg HTTP/1.1
224	18:31:10.779470	151.101.2.132	192.168.1.5	HTTP	176	HTTP/1.1 200 OK (text/css)
232	18:31:10.780863	192.168.1.5	151.101.2.132	HTTP	372	GET /js/jquery-2.1.1.min.js HTTP/1.1
262	18:31:10.787116	151.101.2.132	192.168.1.5	HTTP	1206	HTTP/1.1 200 OK (JPEG JFIF image)
263	18:31:10.787116	151.101.2.132	192.168.1.5	HTTP	698	HTTP/1.1 200 OK (JPEG JFIF image)
267	18:31:10.788729	192.168.1.5	151.101.2.132	HTTP	365	GET /js/bootstrap.js HTTP/1.1
268	18:31:10.789021	192.168.1.5	151.101.2.132	HTTP	470	GET /img/trillions-and-trillions/apache-everywhere-thumbnail.jpg
277	18:31:10.795343	151.101.2.132	192.168.1.5	HTTP	1373	HTTP/1.1 200 OK (JPEG JFIF image)
280	18:31:10.796376	192.168.1.5	151.101.2.132	HTTP	365	GET /js/slideshow.js HTTP/1.1
293	18:31:10.806650	151.101.2.132	192.168.1.5	HTTP	453	HTTP/1.1 200 OK (application/javascript)
298	18:31:10.807118	151.101.2.132	192.168.1.5	HTTP	485	HTTP/1.1 200 OK (application/javascript)
303	18:31:10.808603	151.101.2.132	192.168.1.5	HTTP	593	HTTP/1.1 200 OK (JPEG JFIF image)
305	18:31:10.809389	192.168.1.5	151.101.2.132	HTTP	476	GET /img/trillions-and-trillions/trillions-and-trillions-thumbnai
307	18:31:10.810179	151.101.2.132	192.168.1.5	HTTP	894	HTTP/1.1 200 OK (application/javascript)
308	18:31:10.811284	192.168.1.5	151.101.2.132	HTTP	470	GET /img/trillions-and-trillions/apache-innovation-thumbnail.jpg
309	18:31:10.811558	192.168.1.5	151.101.2.132	HTTP	430	GET /img/2020-report.jpg HTTP/1.1
310	18:31:10.812424	192.168.1.5	151.101.2.132	HTTP	428	GET /img/community.jpg HTTP/1.1
340	18:31:10.830652	151.101.2.132	192.168.1.5	HTTP	1187	HTTP/1.1 200 OK (JPEG JFIF image)

Figure 5: Apache: http



cse.pcapng

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

http

No.	Time	Source	Destination	Proto	Length	Info
15	18:35:24.780020	192.168.1.5	103.27.9.152	HTTP	491	GET / HTTP/1.1
17	18:35:24.871223	103.27.9.152	192.168.1.5	HTTP	797	HTTP/1.1 301 Moved Permanently (text/html)

Figure 6: CSE: http

Detailed proof of Question 2 (click to open link of wireshark saved files)

3 Traceroute Implementation

Language used: Python

3.1 Requirements:

1. The code is written in windows so might not run in other OS.
2. Make sure you have python installed in your system.
3. Also check for python packages matplotlib, sys and subprocess.

3.2 How to run

1. After unzipping a python file q3.py will be created, this is the code file.
2. Run q3.py using command python q3.py in terminal.
3. You can give the domain name as an argument in command line and if you don't program will ask to input the domain name once you run it.
4. Output format will be same as windows tracert command, only the number of packets send is 4 here.
5. After all the hops are done, a graph will be displayed in a new window displaying the RTT vs hop graph (RTT values are averaged over all 4 packets).

3.3 Approach

1. Fist I find out the IP address of the domain name provided using nslookup command.
2. Once I have the domain name I keep on calling ping on this IP address with ttl values starting from 1 till (Maximum 30 hops).
3. If the packet expires at some IP address then get that IP address (say currentIP) from the ping's output and call ping on currentIP, and displays the time for all the 4 packets of ping.
4. Also I store the avg RTT values of all the 4 packets. If some IP Address is not reached (timeout) then as per the directions provided on piazza I took RTT value to be 0.
5. I will stop once I get the currentIP to be equal to the destination IP (that is we have reached to the destination) or ttl value is more than 30.

3.4 Output Screenshot

```
sachi@MSI MINGW64 /e/Semesters/Sem5/COL334/Assignments/A1
$ python q3.py www.google.com

Tracing route to www.google.com [142.250.193.4]
over a maximum of 30 hops:

1    1ms     1ms     2ms     4ms     192.168.1.1
2    2ms     4ms     5ms     4ms     117.207.176.1
3    2ms     5ms     4ms     6ms     218.248.174.113
4    2ms     5ms     5ms     4ms     218.248.165.190
5    24ms    25ms    24ms    26ms    218.248.255.22
6    37ms    54ms    45ms    28ms    218.248.255.23
7    *       *       *       *       142.250.161.230
8    33ms    35ms    35ms    34ms    108.170.248.177
9    37ms    39ms    37ms    38ms    108.170.248.186
10   31ms    33ms    34ms    34ms    216.239.54.93
11   37ms    51ms    44ms    34ms    142.250.232.91
12   20ms    21ms    27ms    47ms    74.125.243.97
13   14ms    16ms    14ms    15ms    142.251.54.89
14   35ms    39ms    35ms    38ms    142.250.193.4

Trace Complete.
```

Figure 7: Code

3.5 Graph

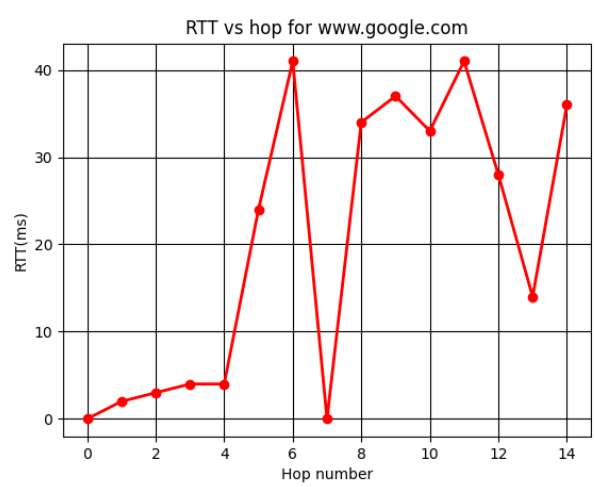


Figure 8: Plot