

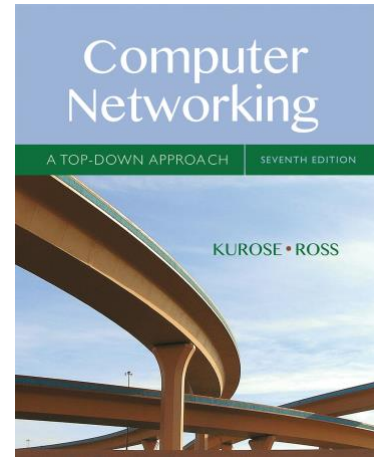
Name: _____

Wireshark Lab: DNS v7.0

Supplement to *Computer Networking: A Top-Down Approach*, 7th ed., J.F. Kurose and K.W. Ross

“Tell me and I forget. Show me and I remember. Involve me and I understand.” Chinese proverb

© 2005-2016, J.F Kurose and K.W. Ross, All Rights Reserved



As described in Section 2.4 of the text¹, 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. Recall that 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. As shown in Figures 2.19 and 2.20 in the textbook, much can go on “under the covers,” invisible to the DNS clients, as the hierarchical DNS servers communicate with each other to either recursively or iteratively resolve the client's DNS query. From the DNS client's standpoint, however, the protocol is quite simple – a query is formulated to the local DNS server and a response is received from that server.

Before beginning this lab, you'll probably want to review DNS by reading Section 2.4 of the text. In particular, you may want to review the material on **local DNS servers**, **DNS caching**, **DNS records and messages**, and the **TYPE field** in the DNS record.

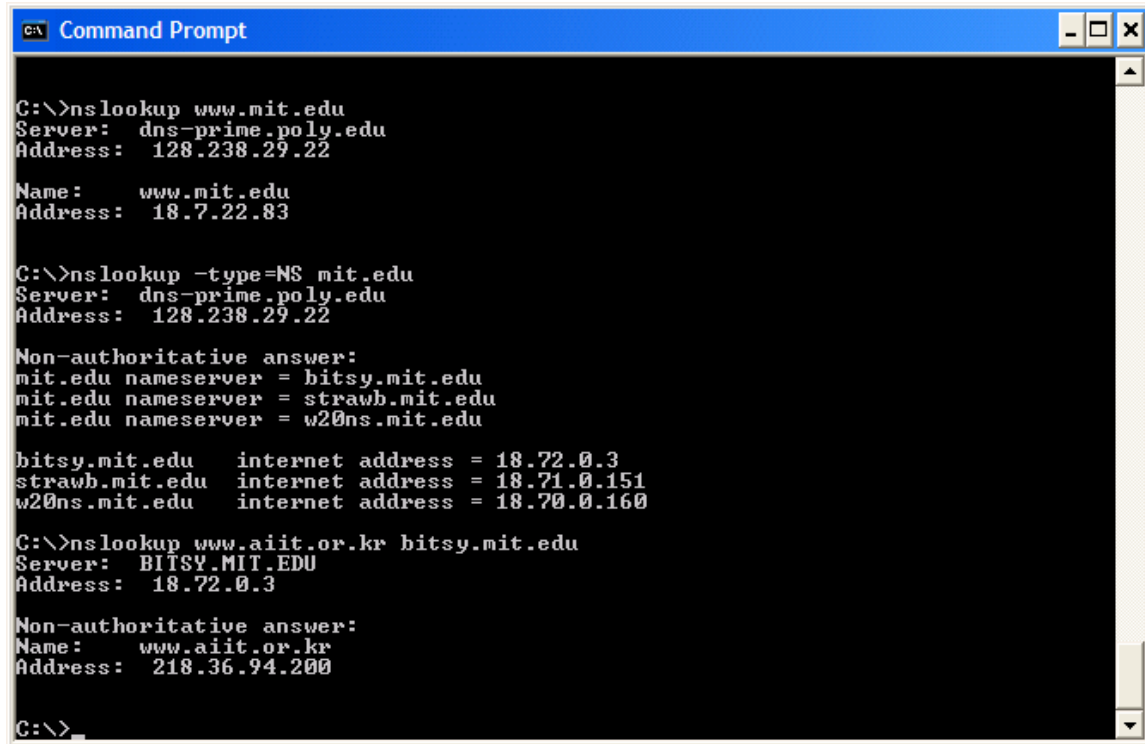
1. nslookup

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 command line. 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.

¹ References to figures and sections are for the 7th edition of our text, *Computer Networks, A Top-down Approach*, 7th ed., J.F. Kurose and K.W. Ross, Addison-Wesley/Pearson, 2016.

Name: _____



```
C:\>nslookup www.mit.edu
Server:  dns-prime.poly.edu
Address: 128.238.29.22

Name:    www.mit.edu
Address: 18.7.22.83

C:\>nslookup -type=NS mit.edu
Server:  dns-prime.poly.edu
Address: 128.238.29.22

Non-authoritative answer:
mit.edu nameserver = bitsy.mit.edu
mit.edu nameserver = strawb.mit.edu
mit.edu nameserver = w20ns.mit.edu

bitsy.mit.edu    internet address = 18.72.0.3
strawb.mit.edu   internet address = 18.71.0.151
w20ns.mit.edu    internet address = 18.70.0.160

C:\>nslookup www.aiit.or.kr bitsy.mit.edu
Server:  BITSY.MIT.EDU
Address: 18.72.0.3

Non-authoritative answer:
Name:    www.aiit.or.kr
Address: 218.36.94.200

C:\>
```

The above screenshot shows the results of three independent *nslookup* commands (displayed in the Windows Command Prompt). In this example, the client host is located on the campus of Polytechnic University in Brooklyn, where the default local DNS server is *dns-prime.poly.edu*. When running *nslookup*, if no DNS server is specified, then *nslookup* sends the query to the default DNS server, which in this case is *dns-prime.poly.edu*. Consider the first command:

```
nslookup www.mit.edu
```

In words, this command is saying “please send me the IP address for the host *www.mit.edu*”. 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 host name and IP address of *www.mit.edu*. Although the response came from the local DNS server at Polytechnic University, it is quite possible that this local DNS server iteratively contacted several other DNS servers to get the answer, as described in Section 2.4 of the textbook.

Now consider the second command:

```
nslookup -type=NS mit.edu
```

In this example, we have provided the option “-type=NS” and the domain “*mit.edu*”. This causes *nslookup* to send a query for a type-NS record to the default local DNS server. In

Name: _____

words, the query is saying, “please send me the host names of the authoritative DNS servers of mit.edu domain”. (When the `-type` option is not used, *nslookup* uses the default, which is to query for type A records, which is IPv4) 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 three DNS servers of MIT domain. Each of these DNS servers is indeed an authoritative DNS server for the hosts on the MIT campus. However, *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 MIT DNS server. Finally, the answer also includes the IP addresses of the authoritative DNS servers at MIT. (Even though the type-NS query generated by *nslookup* did not explicitly ask for the IP addresses, the local DNS server returned these “for free” and *nslookup* displays the result.)

Important note: If you run the same given commands, you may not get the same results. The reason is the names of DNS servers may change over time. Also running these commands from different locations on the Internet may give different results.

Now finally consider the third command:

```
nslookup www.aiit.or.kr bitsy.mit.edu
```

In this example, we indicate that we want to the query sent to the DNS server bitsy.mit.edu 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 bitsy.mit.edu. In this example, the DNS server bitsy.mit.edu provides the IP address of the host www.aiit.or.kr, which is a web server at the Advanced Institute of Information Technology (in Korea).

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.

Now that we have provided an overview of *nslookup*, it is time for you to test drive it yourself. Do the following (and write down the results):

1. Run *nslookup* to obtain the IP address of a Web server in Asia. What is the IP address of that server? **120.108.101.132**
2. Run *nslookup* to determine the authoritative DNS servers for a university in Europe. **dns1.ox.ac.uk** and **dns0.ox.ac.uk**
3. Run *nslookup* so that one of the DNS servers obtained in Question 2 is queried for the mail servers for Yahoo! mail. What is its IP address?

Name: _____

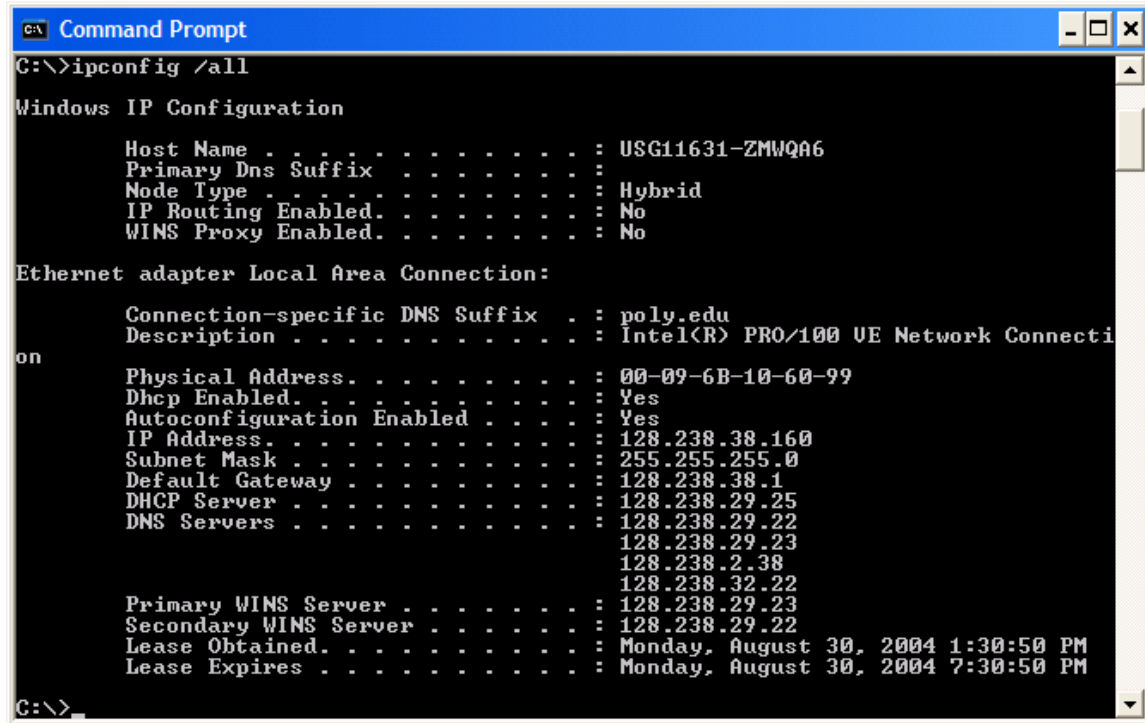
DNS request timed out. timeout was 2 seconds. I kept receiving this error when trying the different DNS servers for ox.ac.uk

2. ipconfig

ipconfig (for Windows) and *ifconfig* (for Linux/Unix) are among the most useful little utilities in your host, especially for debugging network issues. Here we'll only describe *ipconfig*, although the Linux/Unix *ifconfig* is very similar. *ipconfig* can be used to show your current TCP/IP information, including your address, DNS server addresses, adapter type and so on. For example, you get all this information about your host simply by entering

```
ipconfig /all
```

into the Command Prompt, as shown in the following screenshot.



```
Command Prompt
C:\>ipconfig /all

Windows IP Configuration

    Host Name . . . . . : USG11631-ZMWQA6
    Primary Dns Suffix . . . . . :
    Node Type . . . . . : Hybrid
    IP Routing Enabled. . . . . : No
    WINS Proxy Enabled. . . . . : No

Ethernet adapter Local Area Connection:

    Connection-specific DNS Suffix  : poly.edu
    Description . . . . . : Intel(R) PRO/100 VE Network Connecti
on
    Physical Address. . . . . : 00-09-6B-10-60-99
    Dhcp Enabled. . . . . : Yes
    Autoconfiguration Enabled . . . . : Yes
    IP Address. . . . . : 128.238.38.160
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 128.238.38.1
    DHCP Server . . . . . : 128.238.29.25
    DNS Servers . . . . . : 128.238.29.22
                           128.238.29.23
                           128.238.2.38
                           128.238.32.22
    Primary WINS Server . . . . . : 128.238.29.23
    Secondary WINS Server . . . . . : 128.238.29.22
    Lease Obtained. . . . . : Monday, August 30, 2004 1:30:50 PM
    Lease Expires . . . . . : Monday, August 30, 2004 7:30:50 PM

C:\>
```

ipconfig is also very useful for managing the DNS information stored in your host. In Section 2.5 we learned that a host can cache DNS records it recently obtained. To see these cached records, enter the following command:

```
ipconfig /displaydns
```

Each entry shows the remaining Time to Live (TTL) in seconds. Look at the TTL of the last record and run the above command again. You will see that the TTL has decreased, or even the record has disappeared if TTL has been very small. To clear the cache, enter

Name: _____

```
ipconfig /flushdns
```

Flushing the DNS cache clears all entries and reloads the entries from the hosts file.

Now if you run “ipconfig /displaydns” command again, it displays no record.

Name: _____

3. Tracing DNS with Wireshark

Now that we are familiar with *nslookup* and *ipconfig*, we're ready to get down to some serious business. Let's first capture the DNS packets that are generated by ordinary Web-surfing activity.

- Use *ipconfig* to empty the DNS cache in your host.
- Open your browser and empty your browser cache. (With Internet Explorer, go to Tools menu and select Internet Options; then in the General tab select Delete Files.)
- Open Wireshark and enter "ip.addr == your_IP_address" into the filter, where you obtain your_IP_address with *ipconfig*. This filter removes all packets that neither originate nor are destined to your host.
- Start packet capture in Wireshark.
- With your browser, visit the Web page: <http://www.ietf.org>
- Stop packet capture.

If you are unable to run Wireshark on a live network connection, you can download a packet trace file that was captured while following the steps above on one of the author's computers². Answer the following questions. Whenever possible, when answering a question below, you should hand in a printout of the packet(s) within the trace that you used to answer the question asked. Annotate the printout³ to explain your answer. To print a packet, use *File->Print*, choose *Selected packet only*, choose *Packet summary line*, and select the minimum amount of packet detail that you need to answer the question. Note that you can filter DNS traffic by entering "dns" in the filter field.

4. Locate the DNS query and response messages for www.ietf.org. Are they sent over UDP or TCP? **UDP**

² Download the zip file <http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip> and extract the file dns-ethereal-trace-1. The traces in this zip file were collected by Wireshark running on one of the author's computers, while performing the steps indicated in the Wireshark lab. Once you have downloaded the trace, you can load it into Wireshark and view the trace using the *File* pull down menu, choosing *Open*, and then selecting the dns-ethereal-trace-1 trace file.

³ What do we mean by "annotate"? If you hand in a paper copy, please highlight where in the printout you've found the answer and add some text (preferably with a colored pen) noting what you found in what you've highlight. If you hand in an electronic copy, it would be great if you could also highlight and annotate.

Name: _____

ps

ntents

. Displa

1. D

2. Ex

3. G

4. S

5. E

lay fil

re filt

reFilt

nples

r only

tcp.p

r only

.p.src

uffer

tcp.v

on W

smb

r wor

ls_e

h pack

er. No

udp

slice"

restric

eth.

Iso po

h pack

udp

*Ethernet

File Edit View Go Capture Analyze Statistics Telephony Wireless Tools Help

ip.addr==144.37.136.115 and dns

No.	Time	Source	Destination	Protocol	Length	Info
23	2.085572	144.37.136.115	144.37.1.253	DNS	77	Standard query 0x1eea A alertus...
25	2.088470	144.37.1.253	144.37.136.115	DNS	229	Standard query response 0x1eea ...
50	5.320057	144.37.136.115	144.37.1.253	DNS	74	Standard query 0x9014 A www.goo...
51	5.322566	144.37.1.253	144.37.136.115	DNS	162	Standard query response 0x9014 ...
82	6.228731	144.37.136.115	144.37.1.253	DNS	72	Standard query 0x5e59 A www.iet...
83	6.231671	144.37.1.253	144.37.136.115	DNS	239	Standard query response 0x5e59 ...
92	6.245226	144.37.136.115	144.37.1.253	DNS	90	Standard query 0x6b54 A osce12-...
93	6.245268	144.37.136.115	144.37.1.253	DNS	90	Standard query 0x6c54 AAAA osce...
94	6.251503	144.37.1.253	144.37.136.115	DNS	373	Standard query response 0x6b54 ...
95	6.251578	144.37.1.253	144.37.136.115	DNS	397	Standard query response 0x6c54 ...
96	6.251643	144.37.136.115	144.37.1.253	DNS	81	Standard query 0x6d54 A a1441.d...
97	6.254641	144.37.1.253	144.37.136.115	DNS	281	Standard query response 0x6d54 ...

> Destination: IETF-VRRP-VRID_88 (00:00:5e:00:01:88)

> Source: Dell_4a:8a:85 (50:9a:4c:4a:8a:85)

Type: IPv4 (0x0800)

Internet Protocol Version 4, Src: 144.37.136.115, Dst: 144.37.1.253

0100 = Version: 4

.... 0101 = Header Length: 20 bytes (5)

> Differentiated Services Field: 0x00 (DSCP: CS0, ECN: Not-ECT)

Total Length: 58

Identification: 0x49e9 (18921)

> Flags: 0x00

Fragment offset: 0

Time to live: 128

Protocol: UDP (17)

Header checksum: 0x0000 [validation disabled]

[Header checksum status: Unverified]

Source: 144.37.136.115

0000 00 00 5e 00 01 88 50 9a 4c 4a 8a 85 08 00 45 00 ..^...P. LJ....E.

0010 00 3a 49 e9 00 00 80 11 00 00 90 25 88 73 90 25 ..I.... ..%.s.%

0020 01 fd d0 66 00 35 00 26 aa f2 5e 59 01 00 00 01 ...f.5.& ..^Y....

0030 00 00 00 00 00 00 03 77 77 77 04 69 65 74 66 03w ww.ietf.

0040 6f 72 67 00 00 01 00 01 org.....

Domain Name System: Protocol

Packets: 2878 · Displayed: 12 (0.4%) · Dropped: 0 (0.0%) Profile: Default

- h packets where SIP To-header contains the string "a1762" anywhere in the header:
5. What is the destination port for the DNS query message? **Dest port: 53** What is the source port of DNS response message? **Source port: 53**

Name: _____

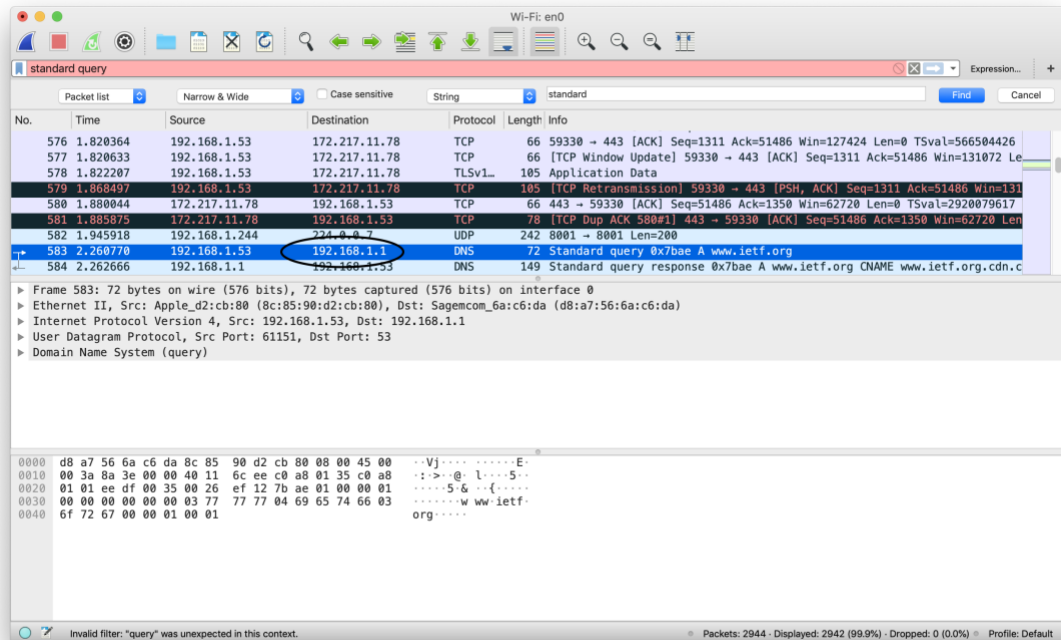
Wireshark packet capture showing a DNS query and response. The packet list shows a query from 144.37.136.115 to 144.37.1.253. The packet details show the destination port is 53. The packet bytes show the query for www.ietf.org.

No.	Time	Source	Destination	Protocol	Length	Info
23	2.085572	144.37.136.115	144.37.1.253	DNS	77	Standard query 0x1eea A alertus...
25	2.088470	144.37.1.253	144.37.136.115	DNS	229	Standard query response 0x1eea ...
50	5.320057	144.37.136.115	144.37.1.253	DNS	74	Standard query 0x9014 A www.goo...
51	5.322566	144.37.1.253	144.37.136.115	DNS	162	Standard query response 0x9014 ...
82	6.228731	144.37.136.115	144.37.1.253	DNS	72	Standard query 0x5e59 A www.iet...
83	6.231671	144.37.1.253	144.37.136.115	DNS	239	Standard query response 0x5e59 ...
92	6.245226	144.37.136.115	144.37.1.253	DNS	90	Standard query 0x6b54 A osce12-...
93	6.245268	144.37.136.115	144.37.1.253	DNS	90	Standard query 0x6c54 AAAA osce...
94	6.251503	144.37.1.253	144.37.136.115	DNS	373	Standard query response 0x6b54 ...
95	6.251578	144.37.1.253	144.37.136.115	DNS	397	Standard query response 0x6c54 ...
96	6.251643	144.37.136.115	144.37.1.253	DNS	81	Standard query 0x6d54 A a1441.d...
97	6.254641	144.37.1.253	144.37.136.115	DNS	281	Standard query response 0x6d54 ...

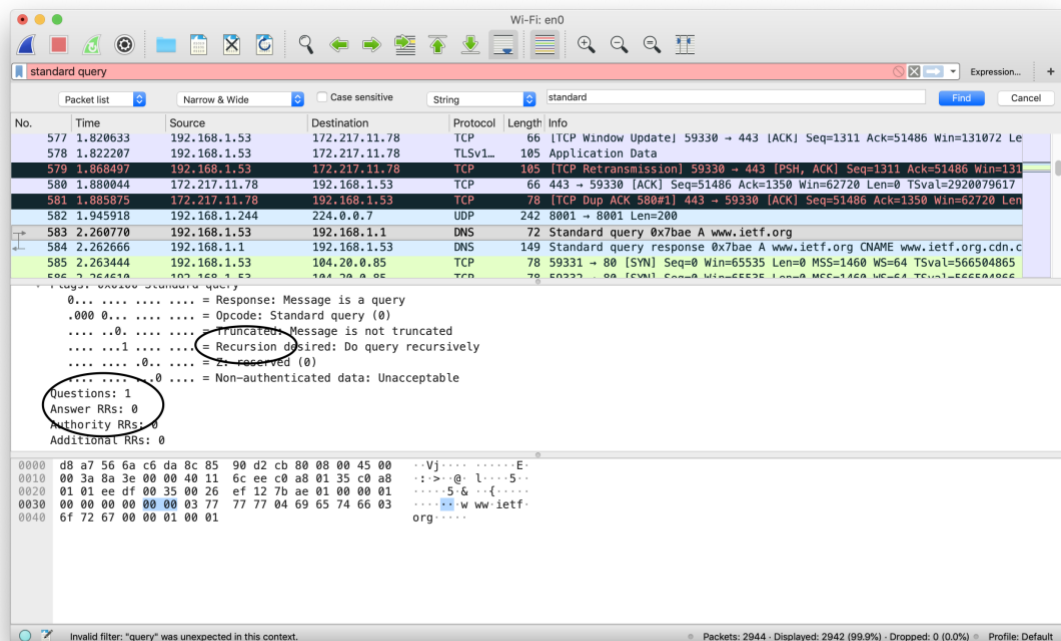
Source: 144.37.136.115
Destination: 144.37.1.253
[Source GeoIP: Unknown]
[Destination GeoIP: Unknown]
User Datagram Protocol, Src Port: 53350, Dst Port: 53
Source Port: 53350
Destination Port: 53
Length: 72
Checksum: 0xaaaf2 [unverified]
[Checksum Status: Unverified]
[Stream index: 13]
Domain Name System (query)
[Response In: 83]
Transaction ID: 0x5e59
Flags: 0x0100 Standard query
Questions: 1
www.ietf.org

6. To what IP address is the DNS query message sent? **192.168.1.1** Use ipconfig to determine the IP address of your local DNS server. Are these two IP addresses the same? **Yes**

Name: _____

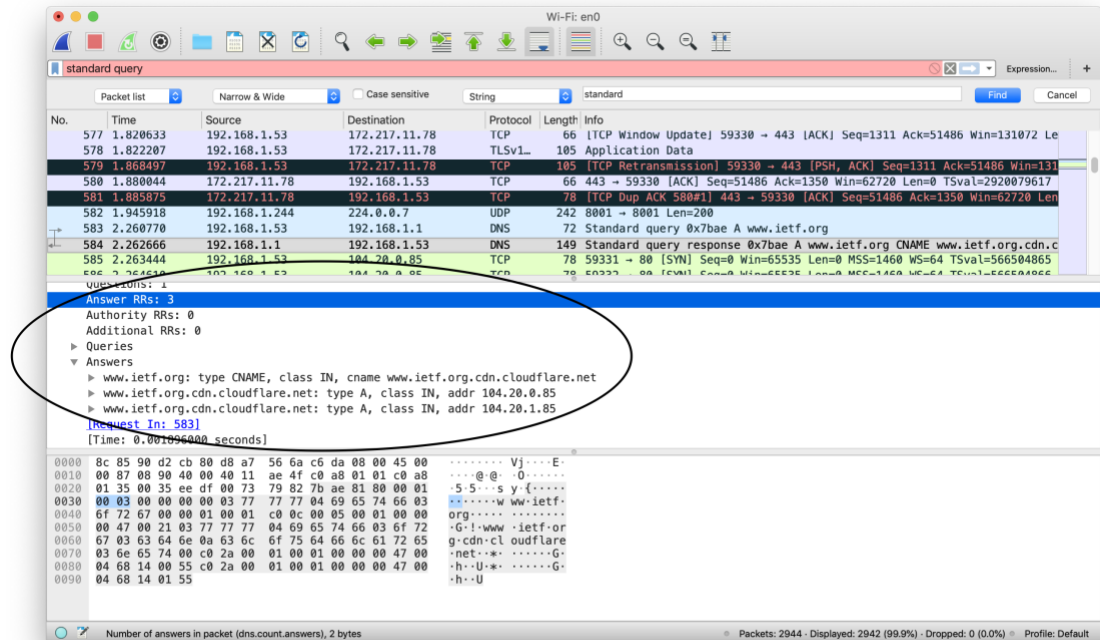


7. Examine the DNS query message. What “Type” of DNS query is it? **Recursive**
Does the query message contain any “answers”? **No**



8. Examine the DNS response message. How many “answers” are provided? **3** What do each of these answers contain? **One type CNAME, and two type A**

Name: _____



9. First write down the frame numbers of the DNS response messages. Then delete anything entered in the filter field. Now consider the subsequent TCP SYN packet sent by your host right after the DNS response messages. Does the destination IP address of the SYN packet correspond to any of the IP addresses provided in the DNS response message? **Filter was not working correctly and could not find the data needed.**
10. This web page contains images. Before retrieving each image, does your host issue new DNS queries? Note: You may filter dns traffic and check whether any DNS message has been exchanged afterwards. **Yes**

Important Note: For the rest of this lab, use the trace files given at footnotes 4, 5, 6. Notice that the names and IP addresses of web servers and DNS servers may change over the time and also the servers may limit their services to off-campus clients. Thus, the following instructions may not work at the time and/or location you run the given commands. Do NOT use live traffic captured by Wireshark for this lab. Use given files.

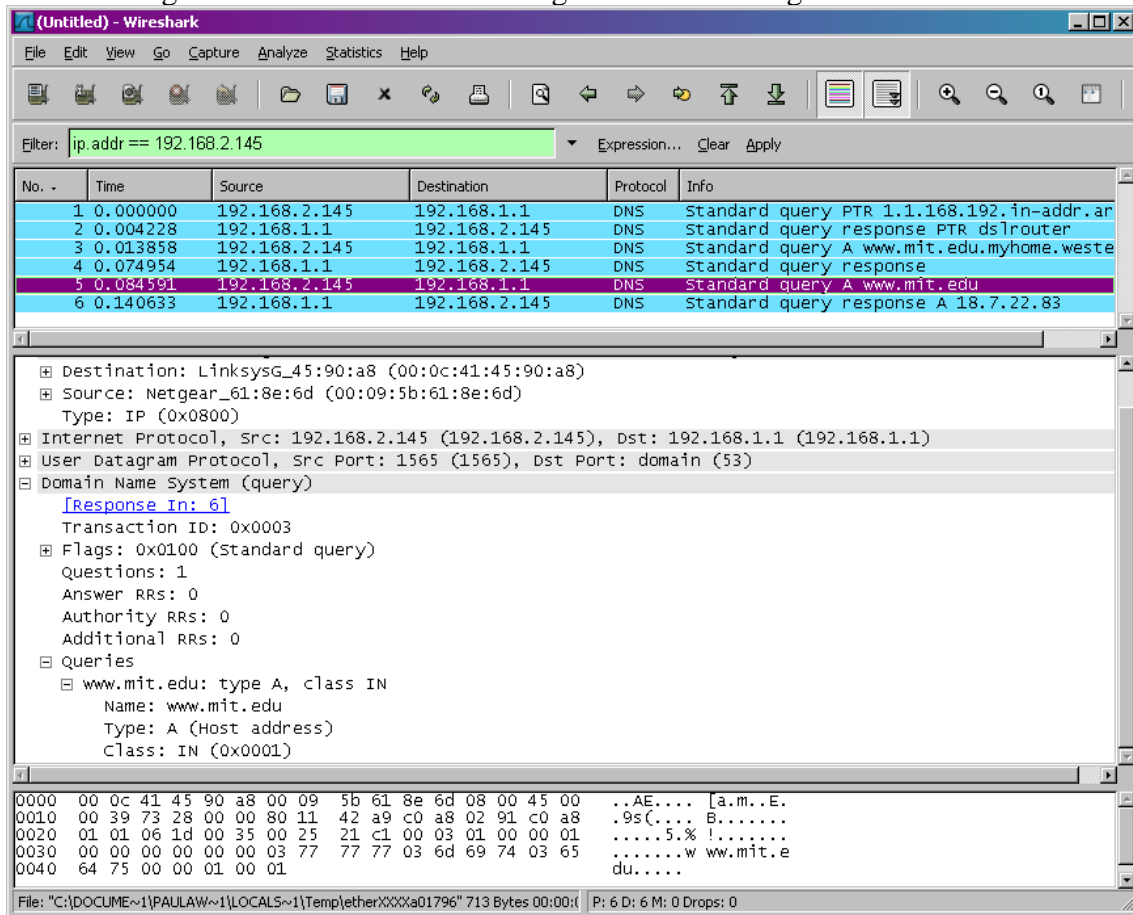
Now let's play with *nslookup*⁴.

- Start packet capture.
- Do an *nslookup* on *www.mit.edu*
- Stop packet capture.

⁴ If you are unable to run Wireshark and capture a trace file, use the trace file *dns-ethereal-trace-2* in the zip file <http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip>

Name: _____

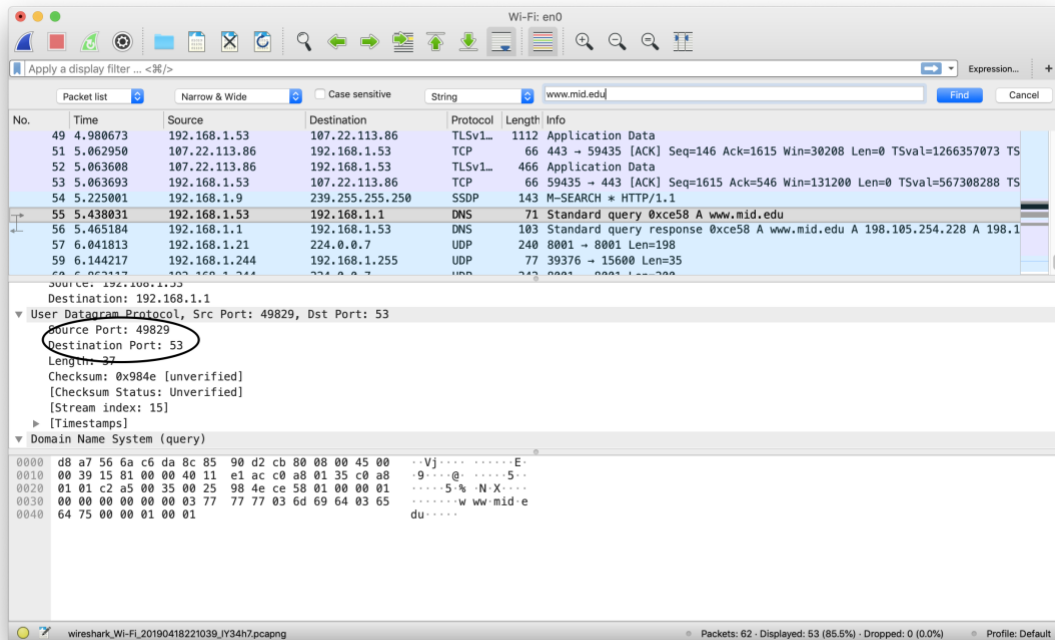
You should get a trace that looks something like the following:



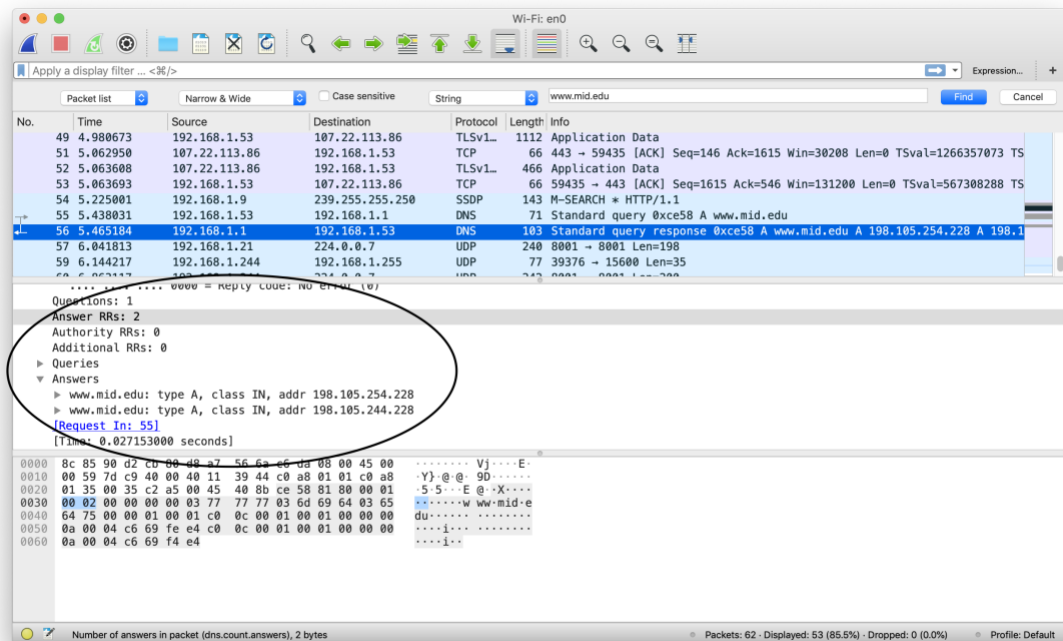
We see from the above screenshot that *nslookup* may actually send three DNS queries and receive three DNS responses. For the purpose of this assignment, in answering the following questions, ignore the first two sets of queries/responses, as they are specific to *nslookup* and are not normally generated by standard Internet applications. You should instead focus on the last query and response messages.

What is the destination port for the DNS query message? **53** What is the source port of DNS response message? **53**

Name: _____



11. To what IP address is the DNS query message sent? **192.168.1.1** Is this the IP address of your default local DNS server? **Yes**
12. Examine the DNS query message. What “Type” of DNS query is it? **Recursive** Does the query message contain any “answers”? **0**
13. Examine the DNS response message. How many “answers” are provided? **2** What do each of these answers contain? **Two type A**
14. Provide a screenshot.



Name: _____

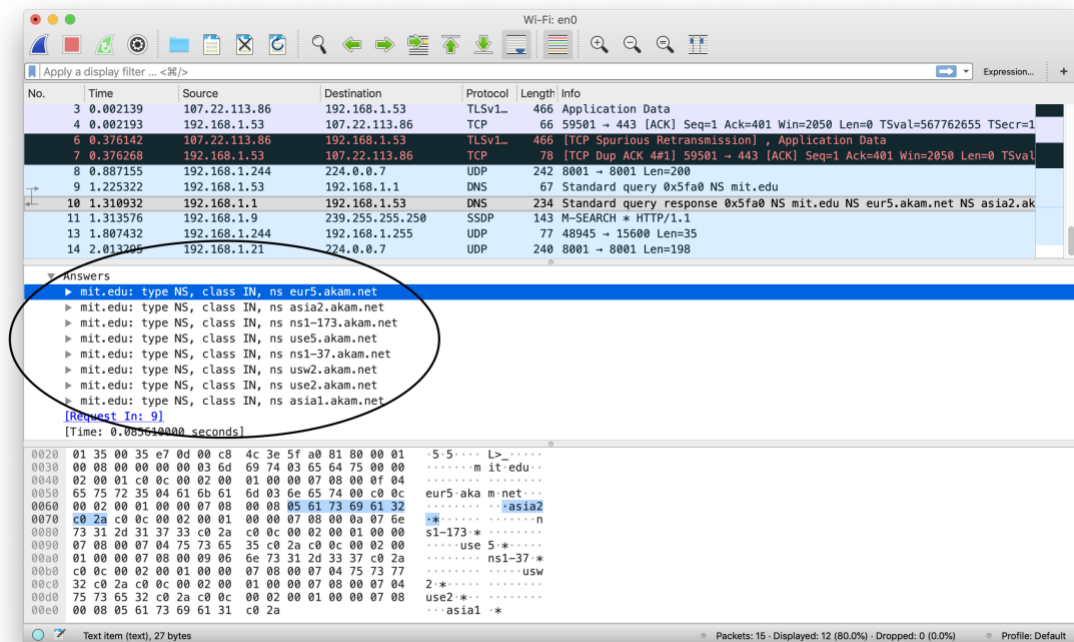
Now repeat the previous experiment, but instead issue the command:
`nslookup -type=NS mit.edu`

Answer the following questions⁵:

15. To what IP address is the DNS query message sent? **192.168.1.1** Is this the IP address of your default local DNS server? **Yes**
16. Examine the DNS query message. What “Type” of DNS query is it? **Recursive**
Does the query message contain any “answers”? **0**
17. Examine the DNS response message. What MIT nameservers does the response message provide?
eur5.akam.net.
asia2.akam.net.
ns1-173.akam.net.
use5.akam.net.
ns1-37.akam.net.
usw2.akam.net.
use2.akam.net.
asia1.akam.net.

Does this response message also provide the IP addresses of the MIT nameservers? **No**

18. Provide a screenshot.



⁵ If you are unable to run Wireshark and capture a trace file, use the trace file dns-ethereal-trace-3 in the zip file <http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip>

Name: _____

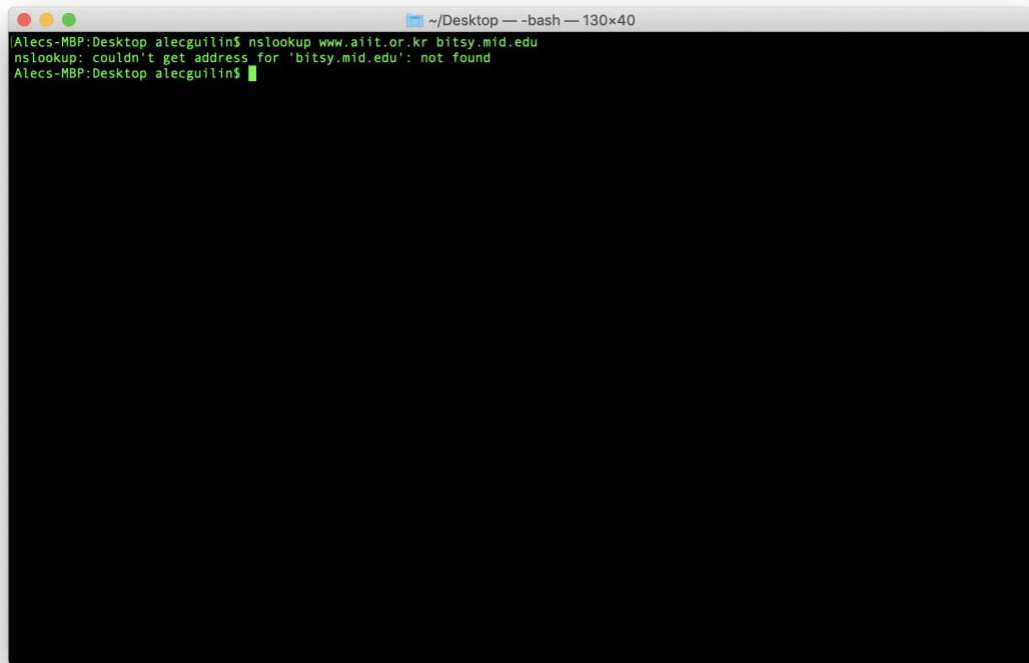
Now repeat the previous experiment, but instead issue the command:

```
nslookup www.aiit.or.kr bitsy.mit.edu
```

Answer the following questions⁶:

19. 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?
20. Examine the DNS query message. What “Type” of DNS query is it? Does the query message contain any “answers”?
21. Examine the DNS response message. How many “answers” are provided? What does each of these answers contain?
22. Provide a screenshot.

Couldn't get this part to work. When I type the nslookup command above I get this error message:

A screenshot of a terminal window on a Mac. The window title is "~/Desktop -- -bash -- 130x40". The prompt is "Alecs-MBP:Desktop alecgulin\$". The user has entered the command "nslookup www.aiit.or.kr bitsy.mit.edu". The output shows the command being executed, followed by an error message: "nslookup: couldn't get address for 'bitsy.mit.edu': not found". The prompt returns to "Alecs-MBP:Desktop alecgulin\$".

```
~/Desktop -- -bash -- 130x40
Alecs-MBP:Desktop alecgulin$ nslookup www.aiit.or.kr bitsy.mit.edu
nslookup: couldn't get address for 'bitsy.mit.edu': not found
Alecs-MBP:Desktop alecgulin$
```

⁶ If you are unable to run Wireshark and capture a trace file, use the trace file dns-ethereal-trace-4 in the zip file <http://gaia.cs.umass.edu/wireshark-labs/wireshark-traces.zip>