DTU Technical University of Denmark

62577 Data Communication Spring 21

Softwareteknologi
Final Submission

Gruppe 21

Submission date:11.05.2021

Muhammad Ali Khan Bangash

STUD
Studerende

Danmarks Tekniske Universitet

E-mail s092512@student.dtu.dk

Indholdsfortegnelse

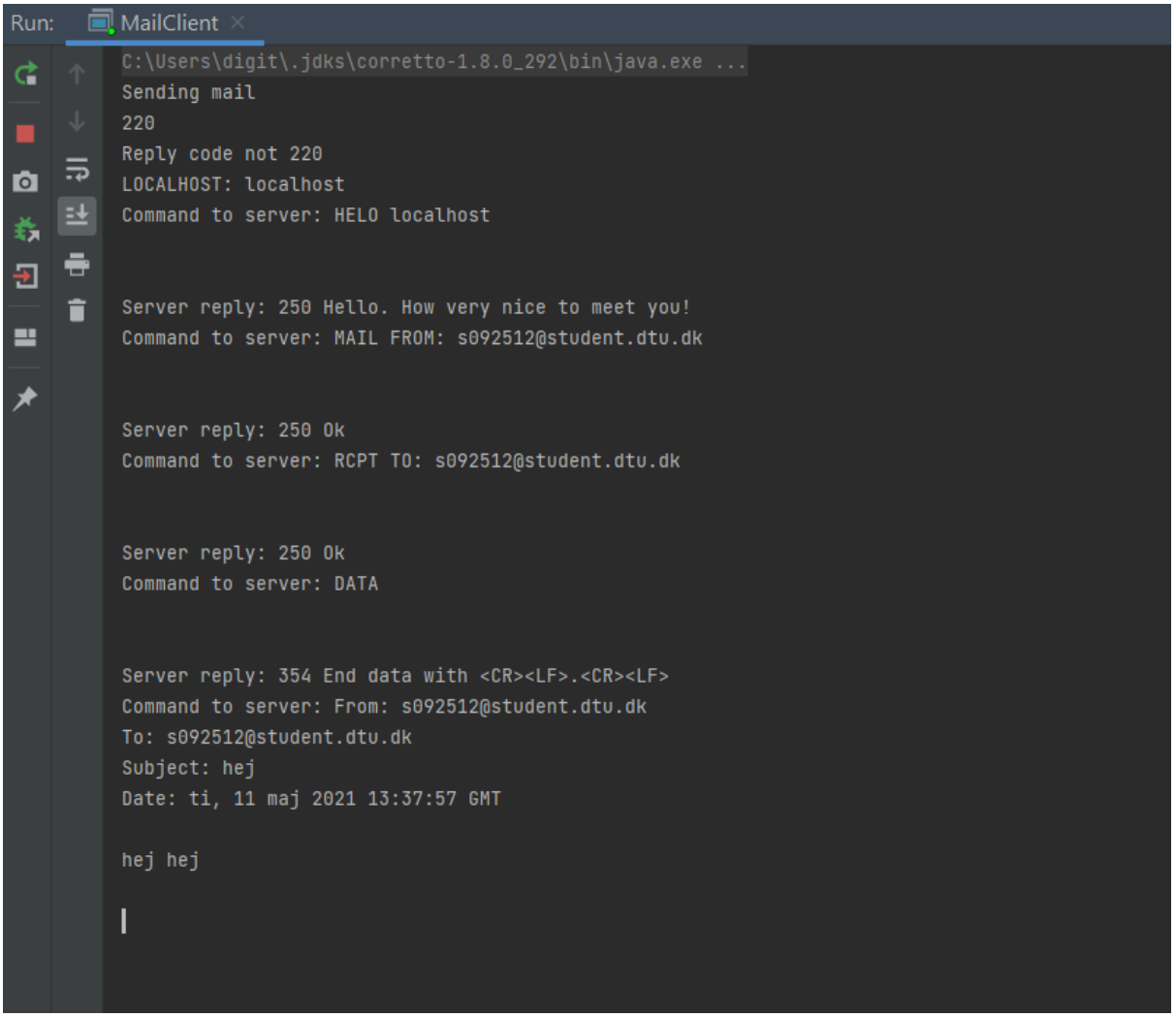
3. Assignment 1: Infographic

Link to our infographic - Click here

4. Summary of Assignment 2

We were assigned to make a programme that could create a mail client that sends e-mail to any recipient. The mail client will need to establish a TCP connection with a mail server (localhost:25) and a dialogue with mail server using SMTP protocol.To be able to do that, we had to connect to the provided SMTP server that will give us access to the server. We used SSH to forward the ports of the mailserver to our machine. Once the connection was established, we used the Java Programme to send some commands that enable us to send the mail to the recipient . For each message the client begins with the process with the new **MAIL FROM:** A 220 code is sent in response to a new user connecting to the server to indicate that the server is ready for the new client. It can also be sent in response to a REIN command, which is meant to reset the connection to the moment the client first connected to the server.Then the command with recipient's email address is sent using **RCPT TO.** And finally the message is sent by using command **DATA.** The details of the programmes will be given in appendix A.

I tried to implement the feedback given on assignment, as my programme works fine, but still no message is received. The suggestions from the feedback did not help and therefore no improvements are made on the behalf of suggestions given.

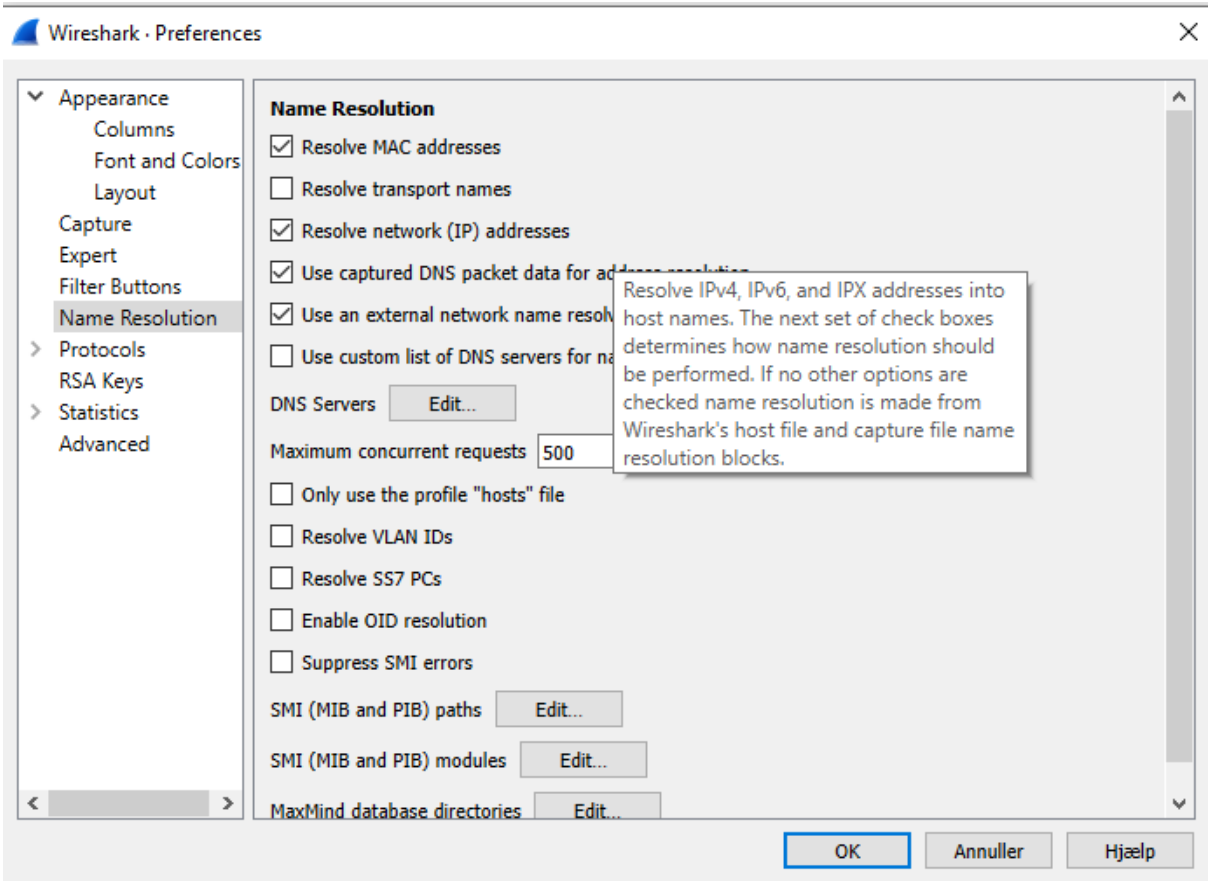


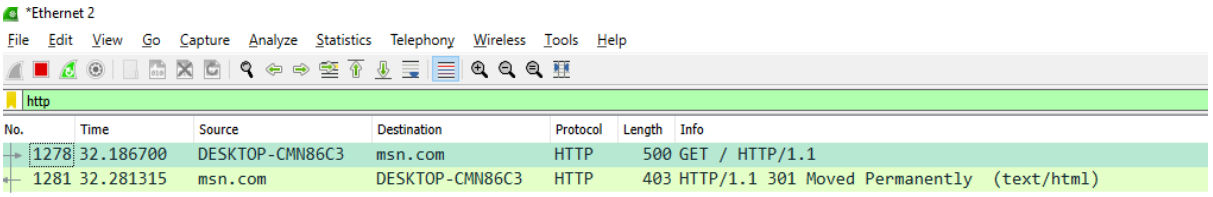
5.a Summary of mandatory assignment 3

Wireshark is a network packet analyzer. A network packet analyzer presents captured packet data in as much detail as possible. This software lets you analyze network traffic in real time, and is often the best tool for troubleshooting issues on your network. Common problems that Wireshark can help troubleshoot include dropped packets, latency issues, and malicious activity on your network. It lets you put your network traffic under a microscope, and provides tools to filter and drill down into that traffic, zooming in on the root cause of the problem.

The feature I want to highligt is While capturing packets, you might be annoyed that Wireshark only displays IP addresses. You can convert the IP addresses to domain names yourself. You can enable this setting by opening the preferences window from *Edit* ->

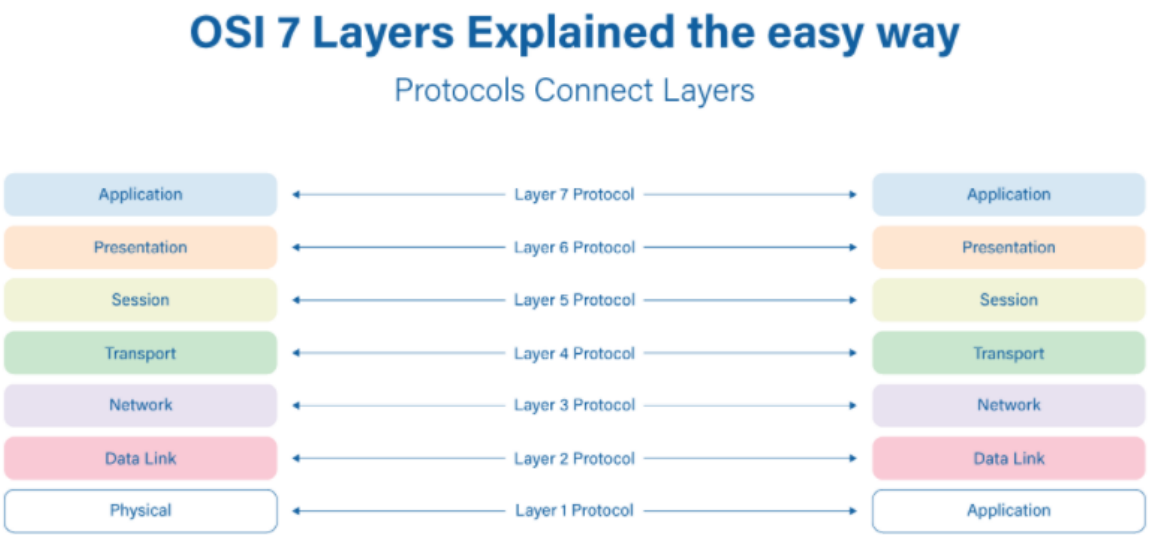
Preferences, clicking the *Name Resolution* panel and clicking the "*Resolve Network Name Resolution*" check box.





5.b Summary of data communication

Data communication networks deal with the transfer of data between two points. Data originates at the source and is finally delivered to the destination, which is also called a sink. It can also be classified according to the type of medium over which the signal propagates. In this case, there are two types of transmissions: guided transmission and wireless transmission; wireless transmission is also called unguided transmission. There are two ways in which data can be transferred from source to destination: switching and broadcasting. In a switched network, data is transferred from source to destination through a series of intermediate switching nodes. In a broadcast network, a transmission from a source is received by all nodes in the network. There are two network access techniques that are closely related to the transfer technique used: switched network access and broadcast network access. The OSI Model is used today as a means to describe Network Architecture. It divides network communication into seven layers. In this model, layers 1-4 are considered the lower layers, and mostly concern themselves with moving data around. Layers 5-7, called the upper layers, and contain application-level data. Networks operate on one basic principle: "pass it on". Each layer takes care of a very specific job, and then passes the data onto the next layer.



Physical Layer

The lowest layer of the OSI Model is concerned with electrically or optically transmitting raw unstructured data bits across the network from the physical layer of the sending device to the physical layer of the receiving device. It can include specifications such as voltages, pin

layout, cabling, and radio frequencies. At the physical layer, one might find “physical” resources such as network hubs, cabling, repeaters, network adapters or modems.

Data Link Layer

At the data link layer, directly connected nodes are used to perform node-to-node data transfer where data is packaged into frames. The data link layer also corrects errors that may have occurred at the physical layer.

The data link layer encompasses two sub-layers of its own. The first, media access control (MAC), provides flow control and multiplexing for device transmissions over a network. The second, the logical link control (LLC), provides flow and error control over the physical medium as well as identifies line protocols.

Network Layer

The network layer is responsible for receiving frames from the data link layer, and delivering them to their intended destinations among based on the addresses contained inside the frame. The network layer finds the destination by using logical addresses, such as IP (internet protocol). At this layer, routers are a crucial component used to quite literally route information where it needs to go between networks.

Transport Layer

The transport layer manages the delivery and error checking of data packets. It regulates the size, sequencing, and ultimately the transfer of data between systems and hosts. One of the most common examples of the transport layer is TCP or the Transmission Control Protocol.

Session Layer

The session layer controls the conversations between different computers. A session or connection between machines is set up, managed, and termined at layer 5. Session layer services also include authentication and reconnections.

Presentation Layer

The presentation layer formats or translates data for the application layer based on the syntax or semantics that the application accepts. Because of this, it at times also called the syntax layer. This layer can also handle the encryption and decryption required by the application layer.

Application Layer

At this layer, both the end user and the application layer interact directly with the software application. This layer sees network services provided to end-user applications such as a web browser or Office 365. The application layer identifies communication partners, resource availability, and synchronizes communication.

Appendix

a. Socket Programming

```
package Datacom;


import java.net.*;
import java.io.*;
import java.util.*;

/*Open an SMTP connection to a mailserver and send one
mail.*/
public class SMTPConnection {
   /* The socket to the server */
   private final Socket connection;

   /* Streams for reading and writing the socket */
   private  BufferedReader fromServer;
   private DataOutputStream toServer;

   private static final int SMTP_PORT = 2025;
   private static final String CRLF = "\r\n";

   /* Are we connected? Used in close() to determine
what to do. */
```

```
    private boolean isConnected = false;

    /* Create an SMTPConnection object. Create the socket
and the
       associated streams. Initialize SMTP connection. */
    public SMTPConnection(Envelope envelope) throws
IOException {

        connection = new Socket("localhost",SMTP_PORT);
        fromServer = new BufferedReader(new
InputStreamReader(connection.getInputStream()));
        toServer = new
DataOutputStream(connection.getOutputStream());


        /* Fill in */
/* Read a line from server and check that the reply code
is 220.
  If not, throw an IOException. */
        String text = fromServer.readLine();
        System.out.println(parseReply(text));
        if (parseReply(text) != 220)
            throw new IOException("Reply code not 220");
        System.out.println("Reply code not 220");



/* SMTP handshake. We need the name of the local
machine.
  Send the appropriate SMTP handshake command. */
        String localhost = "localhost";
        System.out.println ("LOCALHOST: "+localhost);
        sendCommand( "HELO " + localhost + CRLF, 250);
        isConnected = true;
    }

    /* Send the message. Write the correct SMTP-commands
in the
       correct order. No checking for errors, just throw
them to the
       caller. */
    public void send(Envelope envelope) throws
IOException {
/* Send all the necessary commands to send a message.
Call
  sendCommand() to do the dirty work. Do _not_ catch the
  exception thrown from sendCommand(). */
        /* Fill in */
        sendCommand("MAIL FROM: " + envelope.Sender +
CRLF,250);
```

```
        sendCommand("RCPT TO: " + envelope.Recipient +
CRLF ,250);
        sendCommand("DATA"+ CRLF ,354);
        sendCommand(envelope.Message.toString(),250);
        sendCommand(". ", 250);
    }

    /* Close the connection. First, terminate on SMTP
level, then
       close the socket. */
    public void close() {
        isConnected = false;
        try {
            sendCommand("QUIT" + CRLF, 221);
            connection.close();
        }
        catch (IOException e) {
            System.out.println("The system can not close
connection: " + e);
            isConnected = true;
        }
    }

    /* Send an SMTP command to the server. Check that the
reply code is
       what is is supposed to be according to RFC 821. */
    private void sendCommand(String command, int rc)
throws IOException {

        /* Write command to server and read reply from
server. */
        System.out.println("Command to server: " +
command +CRLF);
        toServer.writeBytes(command+CRLF);



/* Check that the server's reply code is the same as the
parameter
   rc. If not, throw an IOException. */
        String text = fromServer.readLine();
        System.out.println("Server reply: " + text);

//         if (parseReply(text) != rc) {
//             System.out.println("The reply code is not
the same as the rc");
//             throw new IOException("The reply code is
not the same as the rc");
//         }
```

```java
    }

    /* Parse the reply line from the server. Returns the
reply code. */
    private int parseReply(String reply) {

        StringTokenizer tokens = new
StringTokenizer(reply," ");
        String rc = tokens.nextToken();
        int x = Integer.parseInt(rc);

        return x;

    }
    /* Destructor. Closes the connection if something bad
happens. */
    protected void finalize() throws Throwable {
        if(isConnected) {
            close();
        }
        super.finalize();
    }
}
```

Assignments › View Feedback

Feedback for Mandatory Assignment 2

Add to ePortfolio

Submission Feedback

Overall Feedback

An error in the message is you add an CRLF to much when establishing
an MAIL FROM.
Also ReCIPIENT TO should be RCPT TO.
Since you commented out the error catch, you can see it dies after
trying to send RCPT TO.

Feedback Date

21 April, 2021 5:41 PM

Assignment

Mandatory Assignment 2

Submission ID	Submission(s)	Date Submitted
174235	SMTP1.0.zip (62.79 KB) The program works well but i dont get any mails. Picture attached to source files.	06 April, 2021 12:19 PM

Done

Wireshark Exercises

Uge 1

R12. What advantage does a circuit-switched network have over a packet-switched network? What advantages does TDM have over FDM in a circuit-switched network?

12. A circuit-switched network can guarantee a certain amount of end-to-end bandwidth for the duration of a call. Most packet-switched networks today (including the Internet) cannot make any end-to-end guarantees for bandwidth. FDM requires sophisticated analog hardware to shift signal into appropriate frequency bands.

Uge 2

R5. What information is used by a process running on one host to identify a process running on another host?

1. the IP address, to identify the machine

2. the Port number, to identify the program/application.

Wireshark Lab:HTTPv7.0

8. Inspect the contents of the first HTTP GET request from your browser to the server. Do you see an "IF-MODIFIED-SINCE" line in the HTTP GET?

No

9. Inspect the contents of the server response. Did the server explicitly return the contents of the file? How can you tell?

Yes because we can see the contents in the Line-based text data field.

```
File Data: 371 bytes
Line-based text data: text/html (10 lines)
    \n
    <html>\n
    \n
    Congratulations again!  Now you've downloaded the file lab2-2.html. <br>\n
    This file's last modification date will not change.  <p>\n
    Thus  if you download this multiple times on your browser, a complete copy <br>\n
    will only be sent once by the server due to the inclusion of the IN-MODIFIED-SINCE<br>\n
    field in your browser's HTTP GET request to the server.\n
    \n
```

10. Now inspect the contents of the second HTTP GET request from your browser to the server. Do you see an "IF-MODIFIED-SINCE:" line in the HTTP GET? If so, what information follows the "IF-MODIFIED-SINCE:" header?

Yes the information is following

```
Tue, 30 Mar 2021 05:59:01 GMT\r\n
```

11. What is the HTTP status code and phrase returned from the server in response to this second HTTP GET? Did the server explicitly return the contents of the file? Explain.

```
HTTP/1.1 304 Not Modified\r\n
```

HTTP/1.1 304 Not Modified.. The server did not return the contents of the file since that browser loaded it from its catche.

Uge 3

P18. a. What is a whois database?

WHOIS is a database containing contact and registration information for domain names

b. Use various whois databases on the Internet to obtain the names of two DNS servers. Indicate which whois databases you used.

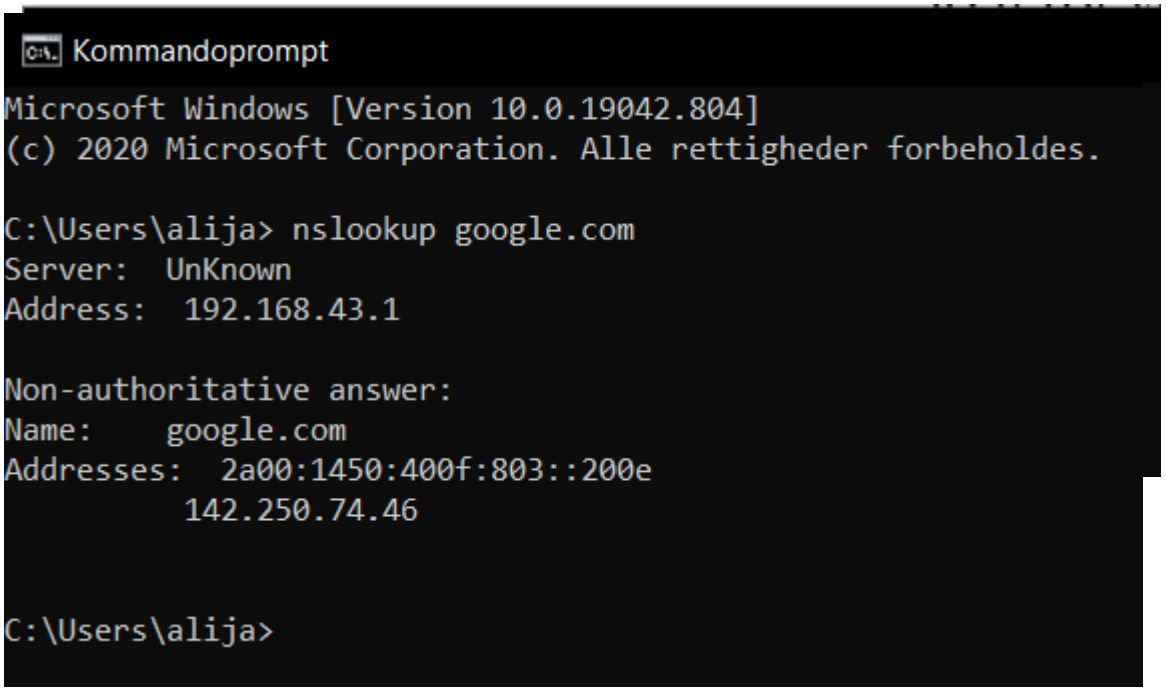
There is a lot of Domain Naming System servers in Google, I used the following:

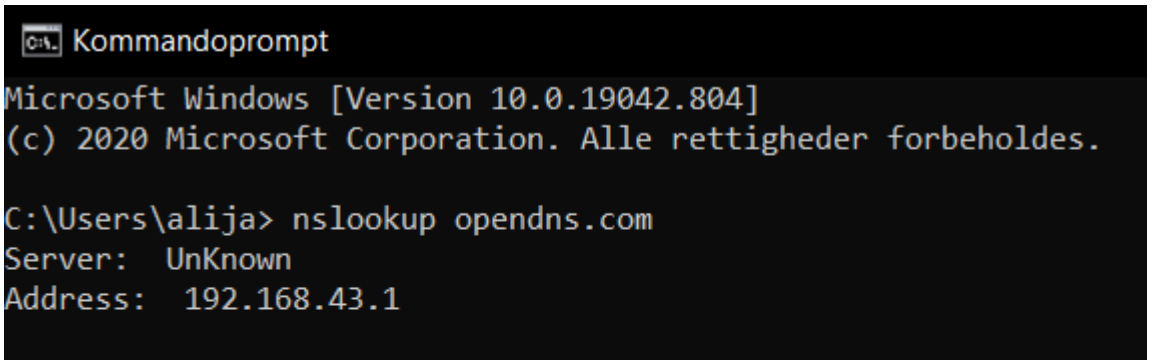
https://whois.domaintools.com/

https://lookup.icann.org/

c. Use nslookup on your local host to send DNS queries to three DNS servers: your local DNS server and the two DNS servers you found in part (b).

Try querying for Type A, NS, and MX reports. Summarize your findings.





d. Use nslookup to find a Web server that has multiple IP addresses. Does the Web server of your institution (school or company) have multiple IP addresses?

Yes.

e. Use the ARIN whois database to determine the IP address range used by your university.

```
Connection-specific DNS Suffix  . :
Link-local IPv6 Address . . . . . : fe80::85d9:7337:cfd5:1fc2%18
IPv4 Address. . . . . . . . . . . : 192.168.43.50
Subnet Mask . . . . . . . . . . . : 255.255.255.0
Default Gateway . . . . . . . . . : 192.168.43.1
```

f. Describe how an attacker can use whois databases and the nslookup tool to perform reconnaissance on an institution before launching an attack.

He can find out every IP Address for the institution we are working for and target those IP Addresses during his attack.

g. Discuss why whois databases should be publicly available.

Because Whois databases are used to find out registration and IP information about domains.

P19.

In this problem, we use the useful dig tool available on Unix and Linux hosts to explore the hierarchy of DNS servers. Recall that in Figure 2.19, a DNS server in the DNS hierarchy delegates a DNS query to a DNS server lower in the

hierarchy, by sending back to the DNS client the name of that lower-level DNS server. First read the man page for dig, and then answer the following questions.

a. Starting with a root DNS server (from one of the root servers [a-m]. root-servers.net), initiate a sequence of queries for the IP address for your department's Web server by using dig. Show the list of the names of DNS servers in the delegation chain in answering your query.

First command: dig +norecurse @a.root-servers.net any gaia.cs.umass.edu

```
; <<>> DiG 9.11.4-P2-RedHat-9.11.4-26.P2.el7_9.4 <<>> +norecurse @a.root-servers.net any gaia.cs.umass.edu
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 58195
;; flags: qr; QUERY: 1, ANSWER: 0, AUTHORITY: 13, ADDITIONAL: 27

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;gaia.cs.umass.edu.             IN      ANY

;; AUTHORITY SECTION:
edu.                    172800  IN      NS      b.edu-servers.net.
edu.                    172800  IN      NS      f.edu-servers.net.
edu.                    172800  IN      NS      i.edu-servers.net.
edu.                    172800  IN      NS      a.edu-servers.net.
edu.                    172800  IN      NS      g.edu-servers.net.
edu.                    172800  IN      NS      j.edu-servers.net.
edu.                    172800  IN      NS      k.edu-servers.net.
edu.                    172800  IN      NS      m.edu-servers.net.
edu.                    172800  IN      NS      l.edu-servers.net.
edu.                    172800  IN      NS      h.edu-servers.net.
edu.                    172800  IN      NS      c.edu-servers.net.
edu.                    172800  IN      NS      e.edu-servers.net.
edu.                    172800  IN      NS      d.edu-servers.net.

;; ADDITIONAL SECTION:
b.edu-servers.net.      172800  IN      A       192.33.14.30
b.edu-servers.net.      172800  IN      AAAA    2001:503:231d::2:30
f.edu-servers.net.      172800  IN      A       192.35.51.30
f.edu-servers.net.      172800  IN      AAAA    2001:503:d414::30
i.edu-servers.net.      172800  IN      A       192.43.172.30
i.edu-servers.net.      172800  IN      AAAA    2001:503:39c1::30
a.edu-servers.net.      172800  IN      A       192.5.6.30
a.edu-servers.net.      172800  IN      AAAA    2001:503:a83e::2:30
g.edu-servers.net.      172800  IN      A       192.42.93.30
```

Among all returned edu DNS servers, we send a query to the first one. dig +norecurse @b.edu-servers.net any gaia.cs.umass.edu

```
; <<>> DiG 9.11.4-P2-RedHat-9.11.4-26.P2.el7_9.4 <<>> +norecurse @b.edu-servers.net any gaia.cs.umass.edu
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 54434
;; flags: qr; QUERY: 1, ANSWER: 0, AUTHORITY: 3, ADDITIONAL: 4

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;gaia.cs.umass.edu.             IN      ANY

;; AUTHORITY SECTION:
umass.edu.              172800  IN      NS      ns1.umass.edu.
umass.edu.              172800  IN      NS      ns3.umass.edu.
umass.edu.              172800  IN      NS      ns2.umass.edu.

;; ADDITIONAL SECTION:
ns1.umass.edu.          172800  IN      A       128.119.10.27
ns3.umass.edu.          172800  IN      A       69.16.40.18
ns2.umass.edu.          172800  IN      A       128.119.10.28

;; Query time: 11 msec
;; SERVER: 192.33.14.30#53(192.33.14.30)
;; WHEN: Tue Mar 30 15:31:29 CEST 2021
;; MSG SIZE  rcvd: 148
```

Among all three returned authoritative DNS servers, we send a query to the first one.

dig +norecurse @ns1.umass.edu any gaia.cs.umass.edu

```
;; MSG SIZE  rcvd: 148

~/Desktop
n-62-27-19(s092512) $ dig +norecurse @ns1.umass.edu  any gaia.cs.umass.edu

; <<>> DiG 9.11.4-P2-RedHat-9.11.4-26.P2.el7_9.4 <<>> +norecurse @ns1.umass.edu any gaia.cs.umass.edu
; (1 server found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 21240
;; flags: qr aa; QUERY: 1, ANSWER: 2, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 4096
;; QUESTION SECTION:
;gaia.cs.umass.edu.             IN      ANY

;; ANSWER SECTION:
gaia.cs.umass.edu.      21600   IN      MX      0 barramail.cs.umass.edu.
gaia.cs.umass.edu.      21600   IN      A       128.119.245.12

;; Query time: 105 msec
;; SERVER: 128.119.10.27#53(128.119.10.27)
;; WHEN: Tue Mar 30 15:33:30 CEST 2021
;; MSG SIZE  rcvd: 88

~/Desktop
n-62-27-19(s092512) $
```

b. Repeat part (a) for several popular Web sites, such as google.com, yahoo

.com, or amazon.com.

For google.com

The answer for google.com could be:

a.root-servers.net

A.GTLD-SERVERS.NET

ns1.google.com(authoritative)

```
~/Desktop
n-62-27-19(s092512) $ dig +norecurse @ns1.google.comt any google.com
dig: couldn't get address for 'ns1.google.comt': not found
~/Desktop
n-62-27-19(s092512) $ dig +norecurse @ns1.google.com any google.com

; <<>> DiG 9.11.4-P2-RedHat-9.11.4-26.P2.el7_9.4 <<>> +norecurse @ns1.google.com any google.com
; (2 servers found)
;; global options: +cmd
;; Got answer:
;; ->>HEADER<<- opcode: QUERY, status: NOERROR, id: 9160
;; flags: qr aa; QUERY: 1, ANSWER: 20, AUTHORITY: 0, ADDITIONAL: 1

;; OPT PSEUDOSECTION:
; EDNS: version: 0, flags:; udp: 512
;; QUESTION SECTION:
;google.com.                    IN      ANY

;; ANSWER SECTION:
google.com.             300     IN      A       172.217.22.174
google.com.             300     IN      AAAA    2a00:1450:400f:807::200e
google.com.             600     IN      MX      30 alt2.aspmx.l.google.com.
google.com.             600     IN      MX      40 alt3.aspmx.l.google.com.
google.com.             60      IN      SOA     ns1.google.com. dns-admin.google.com. 365535376 900 900 1800 60
google.com.             3600    IN      TXT     "docusign=05958488-4752-4ef2-95eb-aa7ba8a3bd0e"
google.com.             600     IN      MX      50 alt4.aspmx.l.google.com.
google.com.             345600  IN      NS      ns4.google.com.
google.com.             3600    IN      TXT     "facebook-domain-verification=22rm551cu4k0ab0bxsw536tlds4h95"
google.com.             86400   IN      CAA     0 issue "pki.goog"
google.com.             345600  IN      NS      ns2.google.com.
google.com.             3600    IN      TXT     "docusign=1b0a6754-49b1-4db5-8540-d2c12664b289"
google.com.             600     IN      MX      10 aspmx.l.google.com.
google.com.             3600    IN      TXT     "v=spf1 include:_spf.google.com ~all"
google.com.             600     IN      MX      20 alt1.aspmx.l.google.com.
google.com.             3600    IN      TXT     "google-site-verification=wD8N7i1JTNTkezJ49swvWW48f8_9xveREV4oB-0Hf5o"
google.com.             345600  IN      NS      ns3.google.com.
google.com.             345600  IN      NS      ns1.google.com.
google.com.             3600    IN      TXT     "apple-domain-verification=30afIBcvSuDV2PLX"
google.com.             3600    IN      TXT     "globalsign-smime-dv=CDYX+XFHUw2wml6/Gb8+59BsH31KzUr6c1l2BPvqKX8="

;; Query time: 24 msec
;; SERVER: 216.239.32.10#53(216.239.32.10)
;; WHEN: Tue Mar 30 15:43:25 CEST 2021
;; MSG SIZE  rcvd: 785
```

Wireshark DNS. 7.01

16.To what IP address is the DNS query message sent? Is this the IP address of your default local DNS server?

It was sent to 192.168.43.1 which is my default DNS server

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

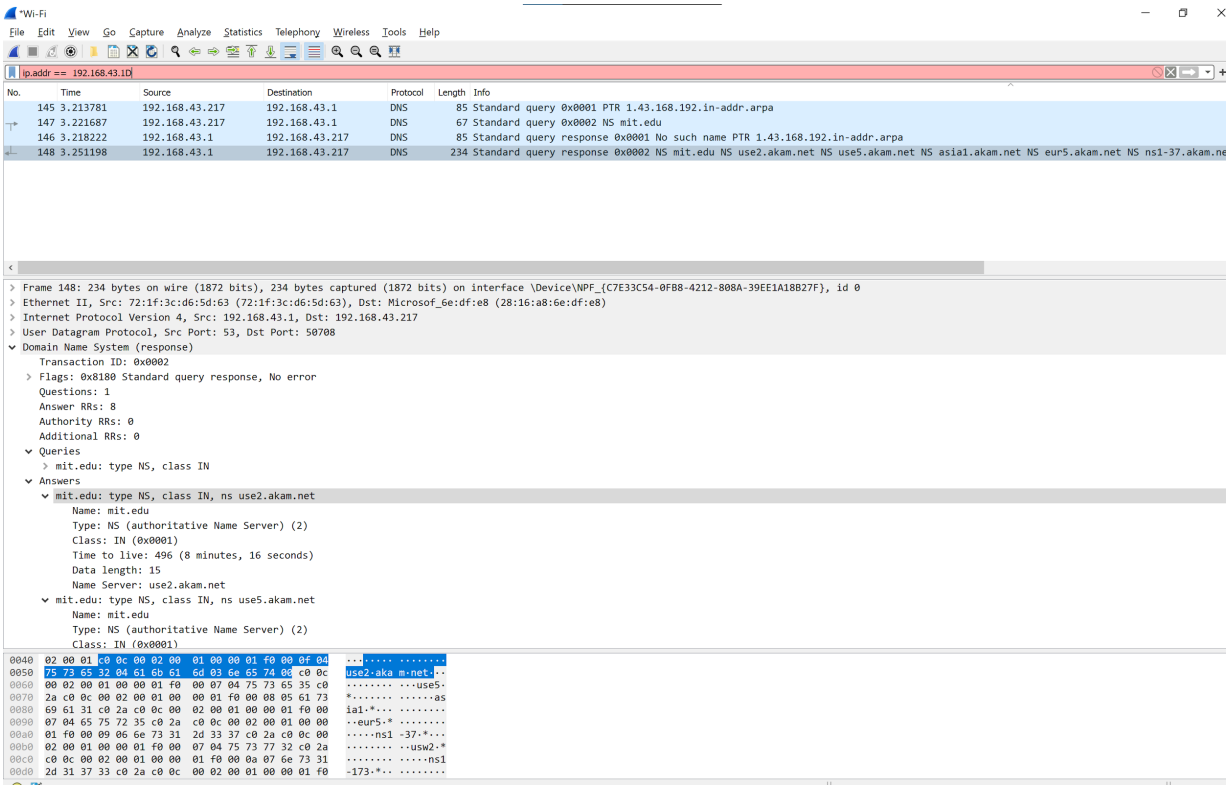
Type is NS. It does not contain any answers.

18.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?

The names are following but i don't get any IP addresses.

```
Answers
    mit.edu: type NS, class IN, ns use2.akam.net
        Name: mit.edu
        Type: NS (authoritative Name Server) (2)
        Class: IN (0x0001)
        Time to live: 496 (8 minutes, 16 seconds)
        Data length: 15
        Name Server: use2.akam.net
    mit.edu: type NS, class IN, ns use5.akam.net
        Name: mit.edu
        Type: NS (authoritative Name Server) (2)
        Class: IN (0x0001)
        Time to live: 496 (8 minutes, 16 seconds)
        Data length: 7
        Name Server: use5.akam.net
    mit.edu: type NS, class IN, ns asia1.akam.net
        Name: mit.edu
        Type: NS (authoritative Name Server) (2)
        Class: IN (0x0001)
        Time to live: 496 (8 minutes, 16 seconds)
        Data length: 8
```

19.Provide a screenshot.



Uge 4

R1. Suppose the network layer provides the following service. The network layer in the source host accepts a segment of maximum size 1,200 bytes and a destination host address from the transport layer. The network layer then guarantees to deliver the segment to the transport layer at the destination host. Suppose many network application processes can be running at the destination host.

a. Design the simplest possible transport-layer protocol that will get application data to the desired process at the destination host. Assume the operating system in the destination host has assigned a 4-byte port number to each running application process.

Call this protocol Simple Transport Protocol (STP). At the sender side, STP accepts from the sending process a chunk of data not exceeding 1196 bytes, a destination host address, and a destination port number. STP adds a four-byte header to each chunk and puts the port number of the destination process in this header. STP then gives the destination host address and the resulting segment to the network layer. The network layer delivers the segment to STP at the destination host. STP then examines the port number in the segment, extracts the data from the segment, and passes the data to the process identified by the port number.

b. Modify this protocol so that it provides a "return address" to the destination process.

The segment now has two header fields: a source port field and destination port field. At the sender side, STP accepts a chunk of data not exceeding 1192 bytes, a destination host address, a source port number, and a destination port number. STP creates a segment which contains the application data, source port number, and destination port number. It then gives the segment and the destination host address to the network layer. After receiving the segment, STP at the receiving host gives the application process the application data and the source port number.

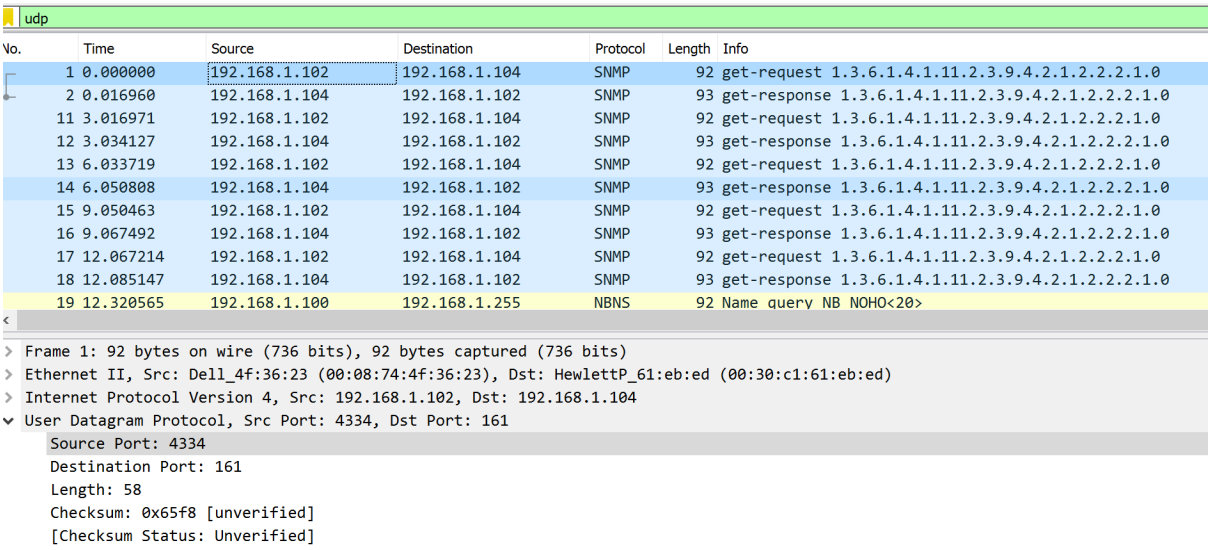
c. In your protocols, does the transport layer "have to do anything" in the core of the computer network?

No, the transport layer does not have to do anything in the core; the transport layer "lives" in the end systems.

Wireshark DP 7.0

7. Examine a pair of UDP packets in which your host sends the first UDP packet and the second UDP packet is a reply to this first UDP packet. (Hint: for a second packet to be sent in response to a first packet, the sender of the first packet should be the destination of the second packet). Describe the relationship between the port numbers in the two packets.

Let's look at packets 1 and 2 in the trace. These packets carry SNMP application -level messages encapsulated inside of UDP. The IP address of the sender of packet 1 is the IP address of the destination of packet 2, and the IP address of the destination of packet 1 is the IP address of the sender of packet 2, The names in the Info field of get-request and get-response suggests that the second packet (a response) is sent in reply to the first packet (a request). Indeed this is the case. The value of the source port in packet 1 is the value of the destination port of packet 2; the value of the destination port of packet 1 is the value of the source port of packet 2,



Uge 5

R14. True or false?

a. Host A is sending Host B a large file over a TCP connection. Assume Host B has no data to send Host A. Host B will not send acknowledgments to Host A because Host B cannot piggyback the acknowledgments on data.

False.

b. The size of the TCP rwnd never changes throughout the duration of the connection.

False.

c. Suppose Host A is sending Host B a large file over a TCP connection. The number of unacknowledged bytes that A sends cannot exceed the size of the receive buffer.

True.

d. Suppose Host A is sending a large file to Host B over a TCP connection. If the sequence number for a segment of this connection is m, then the sequence number for the subsequent segment will necessarily be m + 1.

False.

e. The TCP segment has a field in its header for rwnd.

True

f. Suppose that the last SampleRTT in a TCP connection is equal to 1 sec. The current value of TimeoutInterval for the connection will necessarily be Ú 1 sec.

True.

g. Suppose Host A sends one segment with sequence number 38 and 4 bytes of data over a TCP connection to Host B. In this same segment the acknowledgment number is necessarily 42.

True.

R15

Suppose Host A sends two TCP segments back to back to Host B over a TCP connection. The first segment has sequence number 90; the second has sequence number 110.

a. How much data is in the first segment?

110 - 90 = 20.

b. Suppose that the first segment is lost but the second segment arrives at B. In the acknowledgment that Host B sends to Host A, what will be the acknowledgment number?

In this situation the acknowledgement number will be the first segment of sequence number, and it is 90.

Wireshark TCP 7.0

8. What is the length of each of the first six TCP segments?

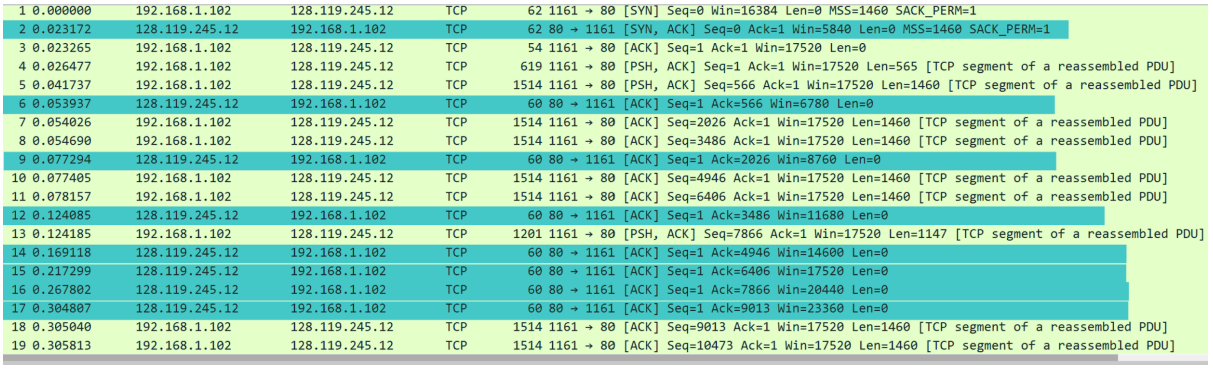
Length of the first TCP segment = 565 bytes

Length of each other five TCP segments = 1460 Bytes

1	0.000000	192.168.1.102	128.119.245.12	TCP	62	1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1
2	0.023172	128.119.245.12	192.168.1.102	TCP	62	80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1
3	0.023265	192.168.1.102	128.119.245.12	TCP	54	1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
4	0.026477	192.168.1.102	128.119.245.12	TCP	619	1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
5	0.041737	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
6	0.053937	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
7	0.054026	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
8	0.054690	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
9	0.077294	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10	0.077405	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
11	0.078157	192.168.1.102	128.119.245.12	TCP	1514	1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
12	0.124085	128.119.245.12	192.168.1.102	TCP	60	80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13	0.124185	192.168.1.102	128.119.245.12	TCP	1201	1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]

9.What is the minimum amount of available buffer space advertised at the received for the entire trace? Does the lack of receiver buffer space ever throttle the sender?

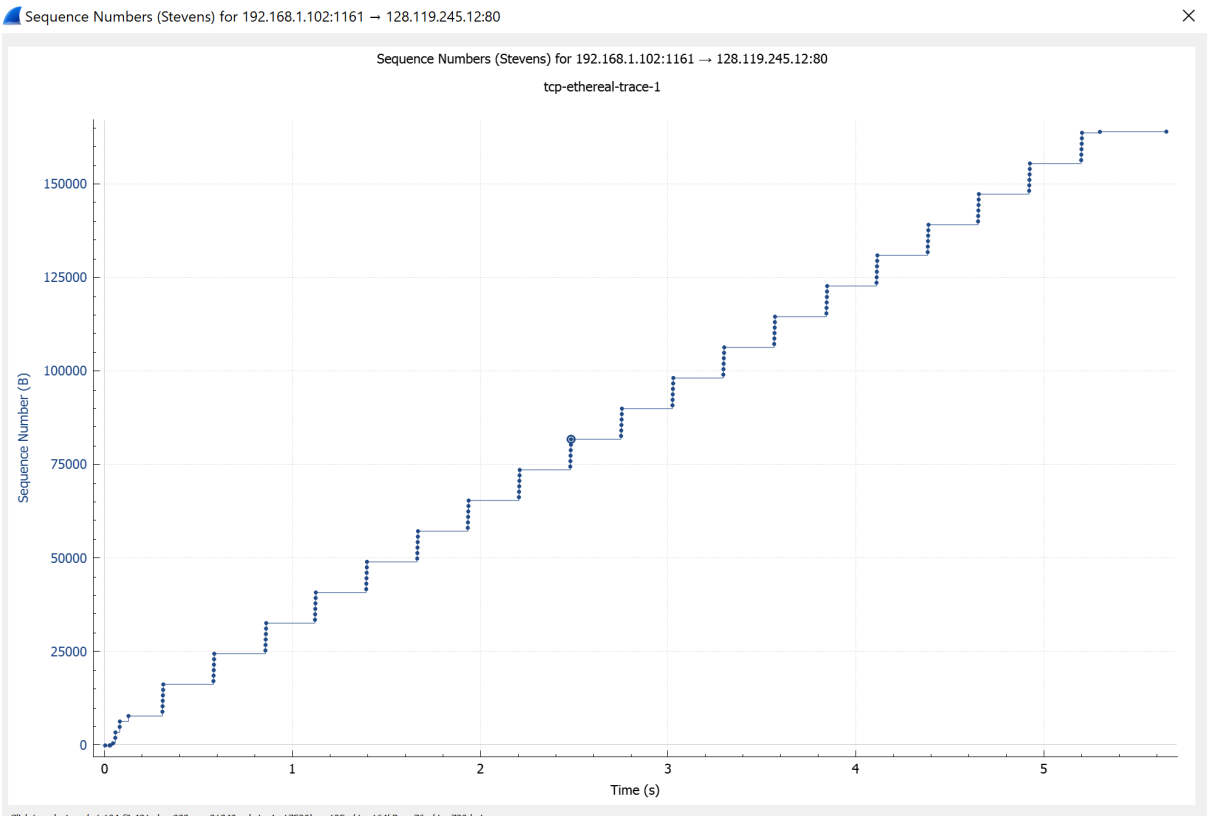
The minimum amount of buffer space advertised at gaia.cs.umass.edu for the entire trace is 5840 bytes, which shows in the first acknowledgement from the server. The receiver window grows steadily until a maximum receiver buffer size of 62870 bytes. The sender is never throttled due to lacking of receiver buffer space by inspecting this trace.



1	0.000000	192.168.1.102	128.119.245.12	TCP	62 1161 → 80 [SYN] Seq=0 Win=16384 Len=0 MSS=1460 SACK_PERM=1
2	0.023172	128.119.245.12	192.168.1.102	TCP	62 80 → 1161 [SYN, ACK] Seq=0 Ack=1 Win=5840 Len=0 MSS=1460 SACK_PERM=1
3	0.023265	192.168.1.102	128.119.245.12	TCP	54 1161 → 80 [ACK] Seq=1 Ack=1 Win=17520 Len=0
4	0.026477	192.168.1.102	128.119.245.12	TCP	619 1161 → 80 [PSH, ACK] Seq=1 Ack=1 Win=17520 Len=565 [TCP segment of a reassembled PDU]
5	0.041737	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [PSH, ACK] Seq=566 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
6	0.053937	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=566 Win=6780 Len=0
7	0.054026	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=2026 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
8	0.054690	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=3486 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
9	0.077294	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=2026 Win=8760 Len=0
10	0.077405	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=4946 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
11	0.078157	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=6406 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
12	0.124085	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=3486 Win=11680 Len=0
13	0.124185	192.168.1.102	128.119.245.12	TCP	1201 1161 → 80 [PSH, ACK] Seq=7866 Ack=1 Win=17520 Len=1147 [TCP segment of a reassembled PDU]
14	0.169118	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=4946 Win=14600 Len=0
15	0.217299	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=6406 Win=17520 Len=0
16	0.267802	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=7866 Win=20440 Len=0
17	0.304807	128.119.245.12	192.168.1.102	TCP	60 80 → 1161 [ACK] Seq=1 Ack=9013 Win=23360 Len=0
18	0.305040	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=9013 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]
19	0.305813	192.168.1.102	128.119.245.12	TCP	1514 1161 → 80 [ACK] Seq=10473 Ack=1 Win=17520 Len=1460 [TCP segment of a reassembled PDU]

10.Are there any retransmitted segments in the trace file? What did you check for (in the trace) in order to answer this question?

There are no retransmitted segments in the trace file. We can verify this by checking the sequence numbers of the TCP segments in the trace file. In the Time-Sequence-Graph (Stevens) of this trace, all sequence numbers from the source (192.168.1.102) to the destination (128.119.245.12) are increasing monotonically with respect to time. If there is a retransmitted segment, the sequence number of this retransmitted segment should be smaller than those of its neighboring segments



Uge 6

R3. We made a distinction between the forwarding function and the routing function performed in the network layer. What are the key differences between routing and forwarding?

Routing: determine route taken by packets from source to destination. Usually routing runs continuously to establish the routing tables before the packets are forwarded and in anticipation of packets.

Forwarding: move packets from router's input to appropriate router output.

R4. What is the role of the forwarding table within a router?

It maps destination addresses to that router's outbound links.

P5

P5. Consider a datagram network using 32-bit host addresses. Suppose a router has four links, numbered 0 through 3, and packets are to be forwarded to the link interfaces as follows:

Destination Address Range	Link Interface
11100000 00000000 00000000 00000000 through 11100000 00000000 11111111 11111111	0
11100000 00000001 00000000 00000000 through 11100000 00000001 11111111 11111111	1
11100000 00000010 00000000 00000000 through 11100001 11111111 11111111 11111111	2
otherwise	3

a. Provide a forwarding table that has five entries, uses longest prefix matching, and forwards packets to the correct link interfaces.

b. Describe how your forwarding table determines the appropriate link interface for datagrams with destination addresses:

11111000 10010001 01010001 01010101
11100000 00000000 11000011 00111100
11100001 10000000 00010001 01110111

a)

Prefix Match	Link Interface
11100000 00	0
11100000 01000000	1
1110000	2
11100001 1	3
otherwise	3

b)

Prefix match for first address is 5th entry: link interface 3
Prefix match for second address is 3nd entry: link interface 2
Prefix match for third address is 4th entry: link interface 3

P11. Consider a subnet with prefix 192.168.56.128/26. Give an example of one IP address (of form xxx.xxx.xxx.xxx) that can be assigned to this network. Suppose an ISP owns the block of addresses of the form 192.168.56.32/26. Suppose it wants to create four subnets from this block, with each block having the same number of IP addresses. What are the prefixes (of form a.b.c.d/x) for the four subnets?

Any IP address in range 128.119.40.128 to 128.119.40.191

Four equal size subnets: 128.119.40.64/28, 128.119.40.80/28, 128.119.40.96/28, 128.119.40.112/28

Wireshark IP v 7.0

2. Within the IP packet header, what is the value in the upper layer protocol field?

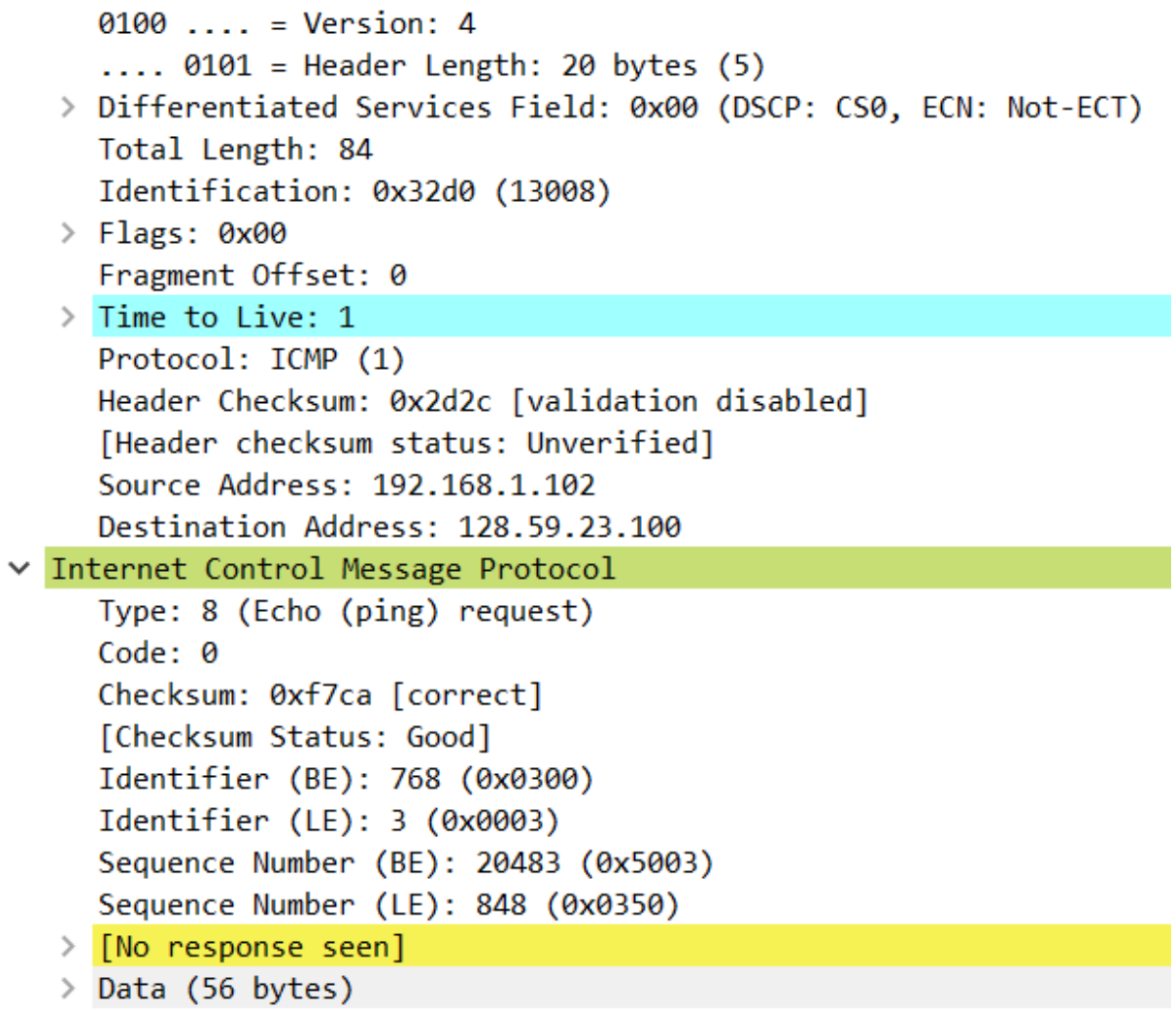
ICMP (1).

```
> Time to Live: 1
  Protocol: ICMP (1)
  Header Checksum: 0x2d2c [validation disabled]
  [Header checksum status: Unverified]
  Source Address: 192.168.1.102
  Destination Address: 128.59.23.100
```

3. How many bytes are in the IP header? How many bytes are in the payload of the IP datagram? Explain how you determined the number of payload bytes.

IP header = 20 bytes

The payload of the IP datagram = 36. (The total length = 56. so 56 - 20 =36)



4. Has this IP datagram been fragmented? Explain how you determined whether or not the datagram has been fragmented.

No, it is not fragmented, because the fragment bit = 0.

UGE 7

R3. Compare and contrast the properties of a centralized and a distributed routing algorithm. Give an example of a routing protocol that takes a centralized and a decentralized approach.

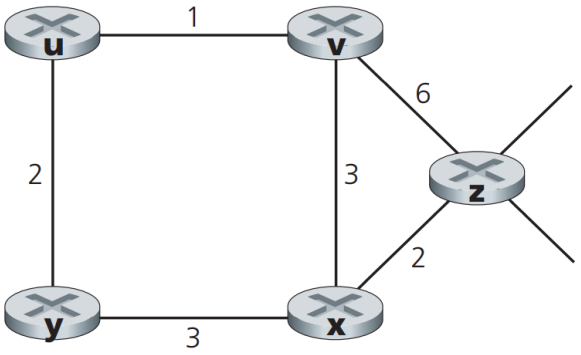
A centralized routing algorithm computes the least-cost path between a source and destination by using complete, global knowledge about the network. The algorithm needs to have the complete knowledge of the connectivity between all nodes and all links' costs. The actual calculation can be run at one site or could be replicated in the routing component of each and every router. A distributed routing algorithm calculates the lease-cost path in an

iterative, distributed manner by the routers. With a decentralized algorithm, no node has the complete information about the costs of all network links. Each node begins with only the knowledge of the costs of its own directly attached links, and then through an iterative process of calculation and information exchange with its neighboring nodes, a node gradually calculates the least-cost path to a destination or a set of destinations.

R5. What is the "count to infinity" problem in distance vector routing?

The count-to-infinity problem refers to a problem of distance vector routing. The problem means that it takes a long time for a distance vector routing algorithm to converge when there is a link cost increase. For example, consider a network of three nodes x, y, and z. Suppose initially the link costs are c(x,y)=4, c(x,z)=50, and c(y,z)=1. The result of distance-vector routing algorithm says that z's path to x is z→y→ x and the cost is 5(=4+1). When the cost of link (x,y) increases from 4 to 60, it will take 44 iterations of running the distance-vector routing algorithm for node z to realize that its new least-cost path to x is via its direct link to x, and hence y will also realize its least-cost path to x is via z.

P5. Consider the network shown below, and assume that each node initially knows the costs to each of its neighbors. Consider the distance-vector algorithm and show the distance table entries at node *z*.



		Cost to				
		u	v	x	y	z
From	v	∞	∞	∞	∞	∞
	x	∞	∞	∞	∞	∞
	z	∞	6	2	∞	0

		Cost to				
		u	v	x	y	z
	v	1	0	3	∞	6
From	x	∞	3	0	3	2
	z	7	5	2	5	0

		Cost to				
		u	v	x	y	z
	v	1	0	3	3	5
From	x	4	3	0	3	2
	z	6	5	2	5	0

		Cost to				
		u	v	x	y	z
	v	1	0	3	3	5
From	x	4	3	0	3	2
	z	6	5	2	5	0

P11. Consider Figure 5.7. Suppose there is another router w, connected to router y and z. The costs of all links are given as follows: c(x,y) = 4, c(x,z) = 50, c(y,w) = 1, c(z,w) = 1, c(y,z) = 3. Suppose that poisoned reverse is used in the distance-vector routing algorithm.

a. When the distance vector routing is stabilized, router w, y, and z inform their distances to x to each other. What distance values do they tell each other?

a)

Router z	Informs w, $D_z(x)=\infty$
	Informs y, $D_z(x)=6$
Router w	Informs y, $D_w(x)=\infty$
	Informs z, $D_w(x)=5$

Router y	Informs w, $D_y(x)=4$
	Informs z, $D_y(x)=4$

b. Now suppose that the link cost between x and y increases to 60. Will there be a count-to-infinity problem even if poisoned reverse is used? Why or why not? If there is a count-to-infinity problem, then how many iterations are needed for the distance-vector routing to reach a stable state again? Justify your answer.

b) Yes, there will be a count-to-infinity problem. The following table shows the routing converging process. Assume that at time t0, link cost change happens. At time t1, y updates its distance vector and informs neighbors w and z. In the following table, "à" stands for "informs"**.**

time	**t0**	**t1**	**t2**	**t3**	**t4**
Z	**→w, $D_z(x)=\infty$** **→y, $D_z(x)=6$**		**No change**	**→w, $D_z(x)=\infty$** **→y, $D_z(x)=11$**	
W	**→y, $D_w(x)=\infty$** **→z, $D_w(x)=5$**		**→y, $D_w(x)=\infty$** **→z, $D_w(x)=10$**		**No change**
Y	**→w, $D_y(x)=4$** **→z, $D_y(x)=4$**	**→w, $D_y(x)=9$** **→z, $D_y(x)=\infty$**		**No change**	**→w, $D_y(x)=14$** **→ z, $D_y(x)=\infty$**

We see that w, y, z form a loop in their computation of the costs to router x. If we continue the iterations shown in the above table, then we will see that, at t27, z detects that its least cost to x is 50, via its direct link with x. At t29, w learns its least cost to x is 51 via z. At t30, y updates its least cost to x to be 52 (via w). Finally, at time t31, no updating, and the routing is stabilized.

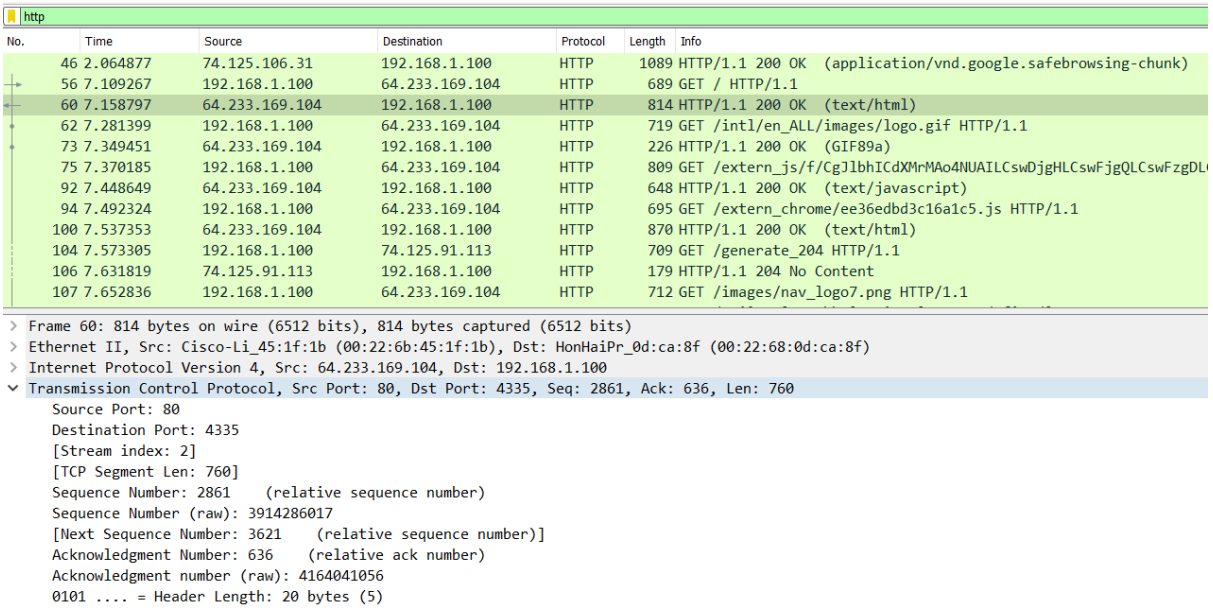
time	t27	t28	t29	t30	t31
Z	**→w, Dz(x)=50** **→y, Dz(x)=50**				**via w, ∞** **via y, 55** **via z, 50**
W		**→y, Dw(x)=∞** **→z, D_w(x)=50**	**→y, Dw(x)=51** **→z, D_w(x)= ∞**		**via w, ∞** **via y, ∞** **via z, 51**
		→w, D_y(x)=53 **→z, D_y(x)= ∞**		**→w, D_y(x)= ∞** **→z, D_y(x)= 52**	**via w, 52** **via y, 60** **via z, 53**

c. How do you modify c(y,z) such that there is no count-to-infinity problem at all if c(y,x) changes from 4 to 60?
Cut the link between y and z**.**

Wireshark NAT v 7.0

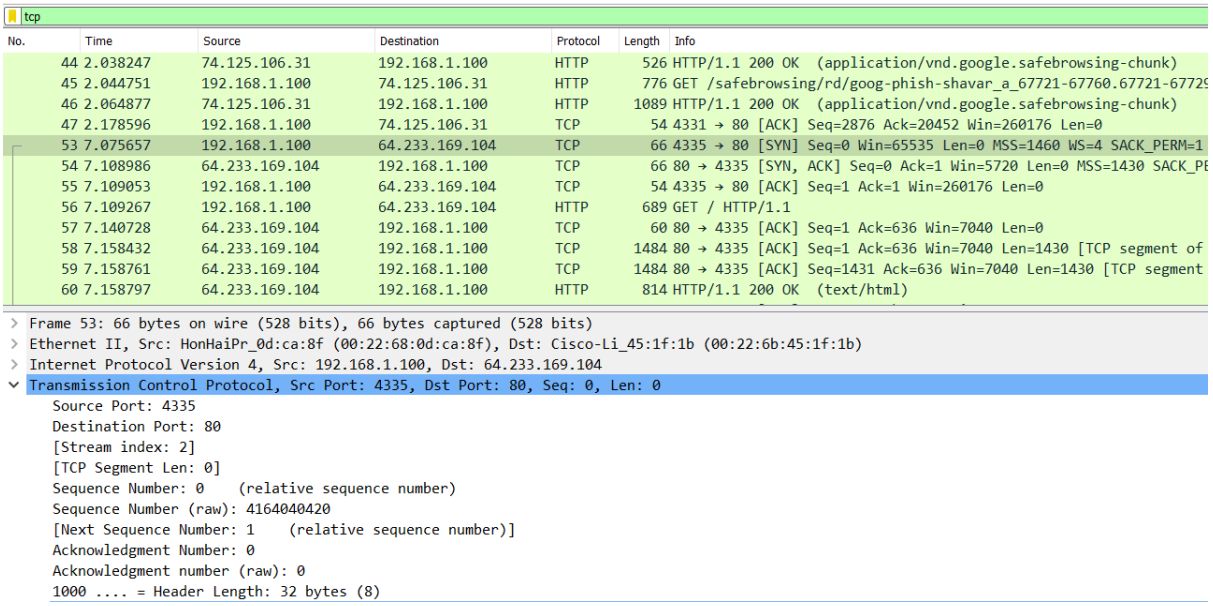
4. At what time is the corresponding 200 OK HTTP message received from the Google server? What are the source and destination IP addresses and TCP source and destination ports on the IP datagram carrying this HTTP 200 OK message?

The time is 7.158797 sec.
The source IP is 64.233.169.104, the port 80.
The destination IP is 192.168.1.100 , the port 4335.

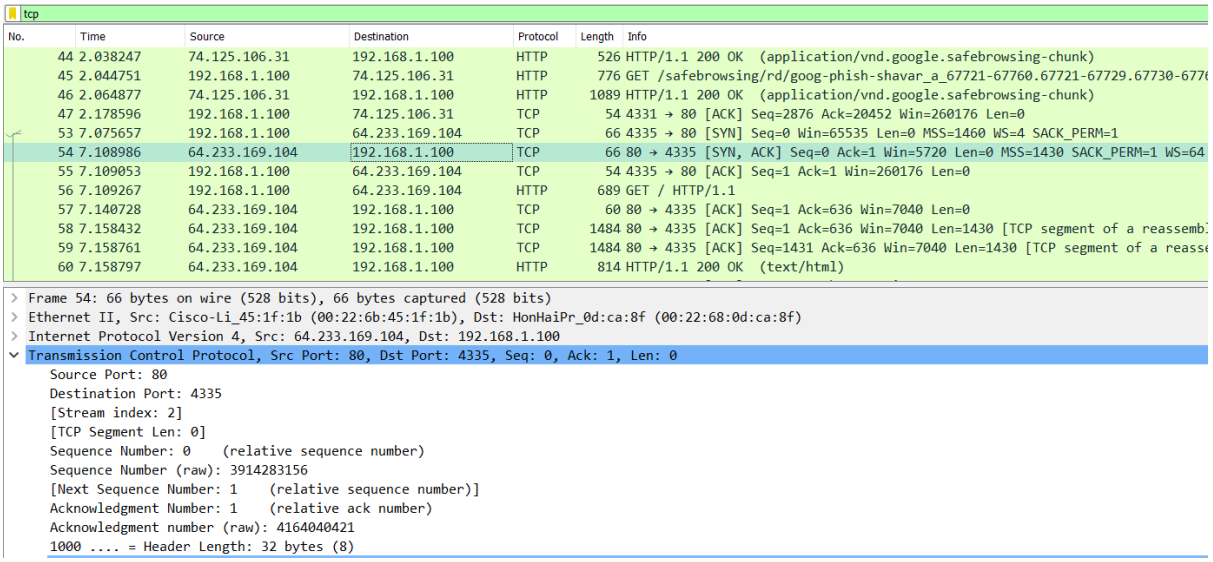


5. Recall that before a GET command can be sent to an HTTP server, TCP must first set up a connection using the three-way SYN/ACK handshake. At what time is the client-to-server TCP SYN segment sent that sets up the connection used by the GET sent at time 7.109267? What are the source and destination IP addresses and source and destination ports for the TCP SYN segment? What are the source and destination IP addresses and source and destination ports of the ACK sent in response to the SYN. At what time is this ACK received at the client? (Note: to find these segments you will need to clear the Filter expression you entered above in step 2. If you enter the filter "tcp", only TCP segments will be displayed by Wireshark).

TCP SYN segment:
The time is 7.075657 sec.
The source IP is 192.168.1.100, the port 4335.
The destination IP is 64.233.169.104, the port 80.



ACK
The time is 7.108986 sec.
The source IP is 64.233.169.104, the port 80.
The destination IP is 192.168.1.100, the port 4335.



8. In the NAT_ISP_side trace file, at what time is the first 200 OK HTTP message received from the Google server? What are the source and destination IP addresses and TCP source and destination ports on the IP datagram carrying this HTTP 200 OK message? Which of these fields are the same, and which are different than your answer to question 4 above?

The time is 6.117.570 sec.
The source IP is 64.233.169.104, the port 80.
The destination IP is 71.192.34.104, the port 4335.

The same fields are: the source IP and its port, and the destination port.
The different fields are: the time and the destination IP.

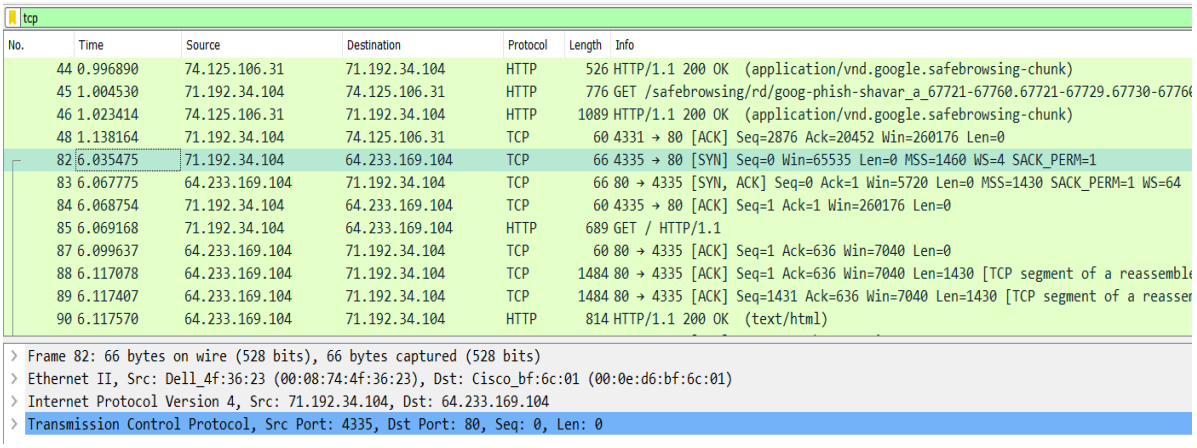
9. In the NAT_ISP_side trace file, at what time were the client-to-server TCP SYN segment and the server-to-client TCP ACK segment corresponding to the segments in question 5 above captured? What are the source and destination IP addresses and source and destination ports for these two segments? Which of these fields are the same, and which are different than your answer to question 5 above?

From client-to-server:

The time is 6.035475 sec.

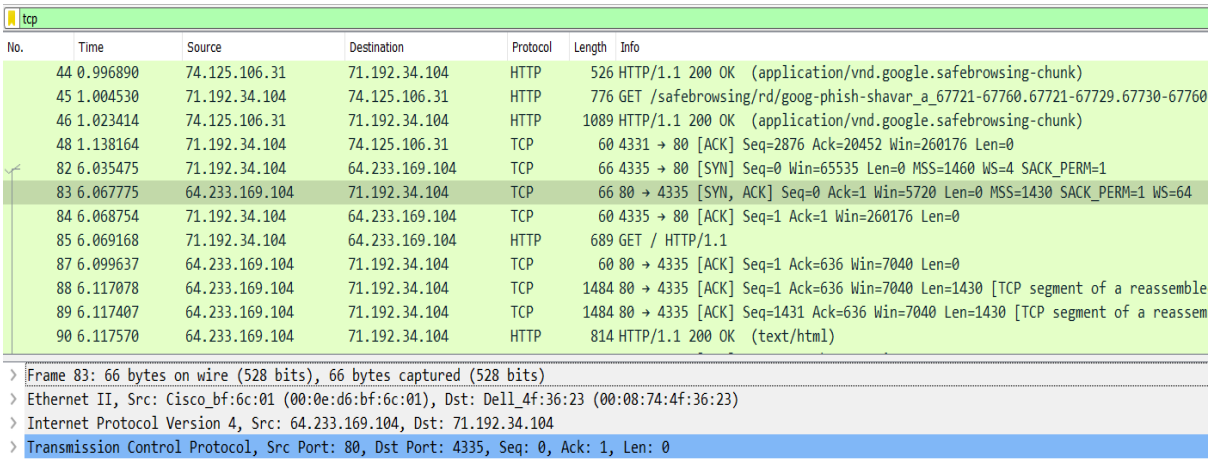
The source IP is 71.192.34.104, the port 4335.

The destination IP is 64.233.169.104, the port 80.



From server-to-client:

The time is 6.067775 sec.
The source IP is 64.233.169.104, the port 80.
The destination IP is 71.192.34.104, the port 4335.



There are not the same fields, all fields are different from each other.

UGE 8

Chapter 4

R32. How does generalized forwarding differ from destination-based forwarding?

Forwarding has two main operations: match and action. With destination-based forwarding, the match operation of a router looks up only the destination IP address of the to-be-forwarded datagram, and the action operation of the router involves sending the packet into the switching fabric to a specified output port. With generalized forwarding, the match can be made over multiple header fields associated with different protocols at different layers in the protocol stack, and the action can include forwarding the packet to one or more output ports, load-balancing packets across multiple outgoing interfaces, rewriting header values (as in NAT), purposefully blocking/dropping a packet (as in a firewall), sending a packet to a special server for further processing and action, and more

R34. What is meant by the "match plus action" operation of a router or switch? In the case of destination-based forwarding packet switch, what is matched and what is the action taken? In the case of an SDN, name three fields that can be matched, and three actions that can be taken.

"Match plus action" means that a router or a switch tries to find a match between some of the header values of a packet with some entry in a flow table, and then based on that match, the router decides to which interface(s) the packet will be forwarded and even some more operations on the packet. In the case of destination-based forwarding packet switch, a router only tries to find a match between a flow table entry with the destination IP address of an arriving packet, and the action is to decide to which interface(s) the packet will be forwarded. In the case of an SDN, there are many fields can be matched, for example, IP source address, TCP source port, and source MAC address; there are also many actions can be taken, for example, forwarding, dropping, and modifying a field value.

Chapter 5

R19. Names four different types of ICMP messages.

Echo reply (to ping), type 0, code 0

Destination network unreachable, type 3 code 0

Destination host unreachable, type 3, code 1.

Source quench (congestion control), type 4 code 0.

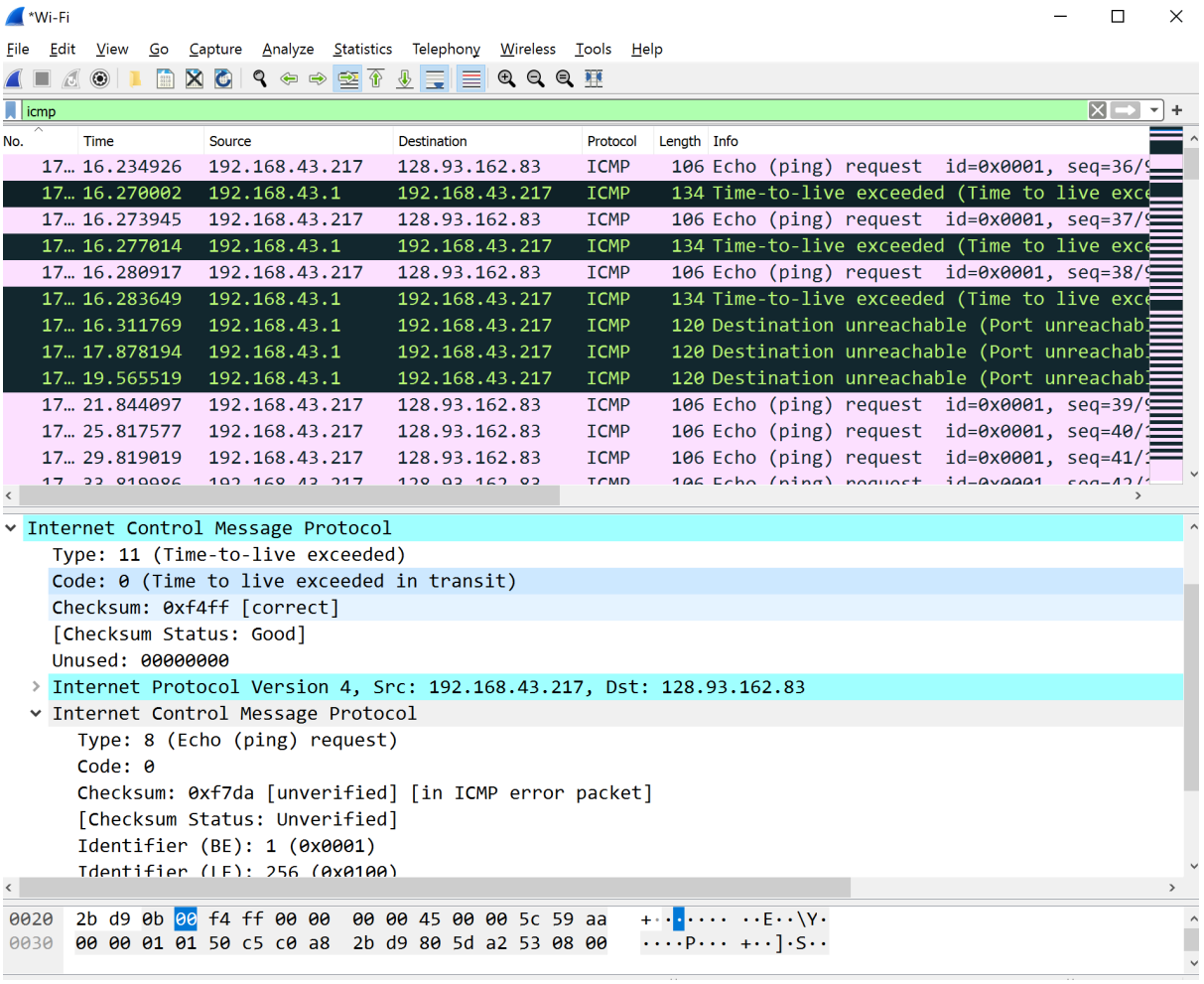
Wireshark ICMP 7.0

5.What is the IP address of your host? What is the IP address of the target destination host?

The IP of the host is 192.168.43.1

The IP of the target is 128.93.162.217

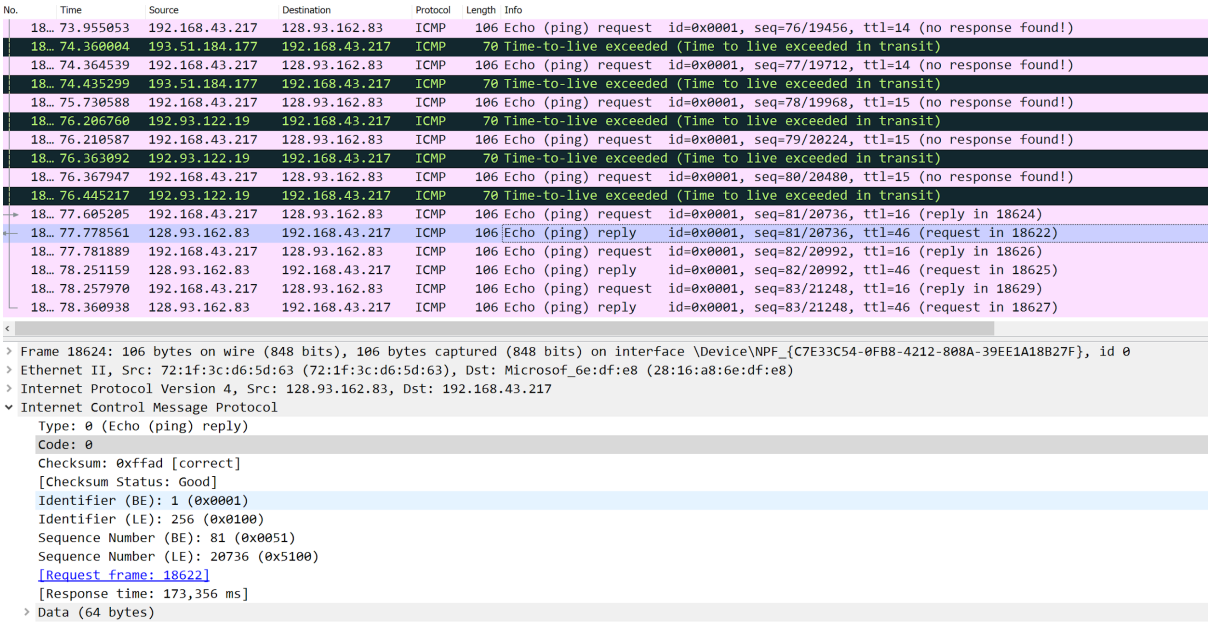
8.Examine the ICMP error packet in your screenshot. It has more fields than the ICMP echo packet. What is included in those fields?



The ICMP error packet is not the same as the ping query packets. It contains both the IP header and the first 8 bytes of the original ICMP packet that the error is for.

9.Examine the last three ICMP packets received by the source host. How are these packets different from the ICMP error packets? Why are they different?

The last three ICMP packets are message type 0 (echo reply) rather than 11(TTL expired). They are different because the datagrams have made it all the way to the destination host before the TTL expired.



Uge 9

R2. If all the links in the Internet were to provide reliable delivery service, would the TCP reliable delivery service be redundant? Why or why not?

TCP Reliable Delivery Service is not redundant. Although each link guarantees that an IP datagram sent over the link will be received at the other end of the link without errors, it is not guaranteed that the IP datagrams will arrive at the ultimate destination in the proper order.

R3. Name three error-detection strategies employed by link layer.

Parity Check, Checksum and Cyclic Redundancy Check (CRC)

R4. Suppose two nodes start to transmit at the same time a packet of length L over a broadcast channel of rate R. Denote the propagation delay between the two nodes as dprop. Will there be a collision if dprop 6 L/R? Why or why not?

There will be a collision in the sense that while a node is transmitting it will start to receive a packet from the other node.

P1. Suppose the information content of a packet is the bit pattern 1010 0111 0101 1001 and an even parity scheme is being used. What would the value of the field containing the parity bits be for the case of a two-dimensional parity scheme? Your answer should be such that a minimum-length checksum field is used.

1 1 1 0 1
0 1 1 0 0
1 0 0 1 0
1 1 0 1 1
1 1 0 0 0

P2. Show (give an example other than the one in Figure 6.5) that two-dimensional parity checks can correct and detect a single bit error. Show (give an example of) a double-bit error that can be detected but not corrected.

Suppose we begin with the initial two-dimensional parity matrix:

0 0 0 0
1 1 1 1
0 1 0 1
1 0 1 0

With a bit error in row 2, column 3, the parity of row 2 and column 3 is now wrong in the matrix below:

0 0 0 0
1 1 0 1
0 1 0 1
1 0 1 0

Now suppose there is a bit error in row 2, column 2 and column 3. The parity of row 2 is now correct! The parity of columns 2 and 3 is wrong, but we can't detect in which rows the error occurred!

0 0 0 0
1 0 0 1
0 1 0 1

1 0 1 0

The above example shows that a double bit error can be detected (if not corrected).

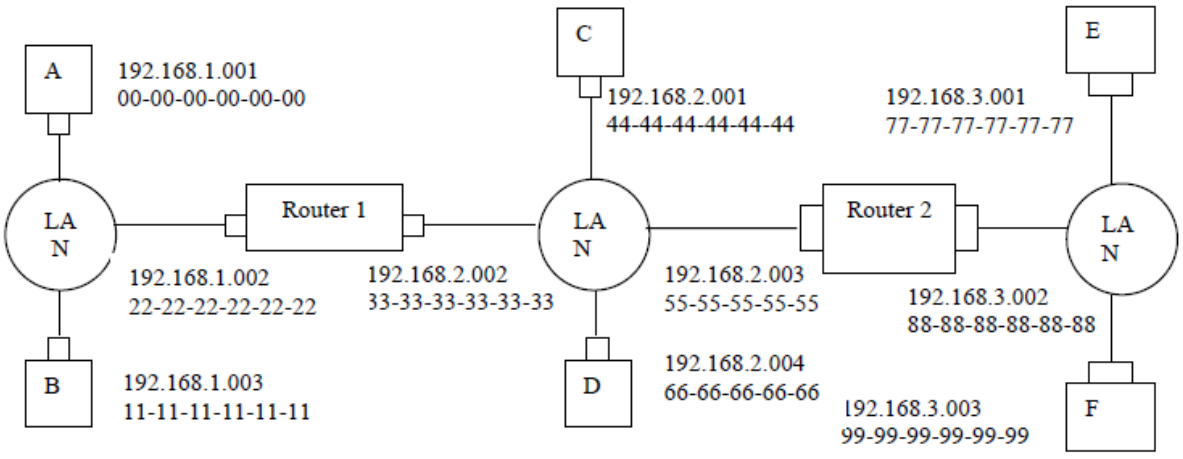
Uge 10

R9. How big is the MAC address space? The IPv4 address space? The IPv6 address space?

2^{48} MAC addresses; 2^{32} IPv4 addresses; 2^{128} IPv6 addresses.

P14. Consider three LANs interconnected by two routers, as shown in Figure 6.33.
a. Assign IP addresses to all of the interfaces. For Subnet 1 use addresses of the form 192.168.1.xxx; for Subnet 2 uses addresses of the form 192.168.2.xxx; and for Subnet 3 use addresses of the form 192.168.3.xxx.
b. Assign MAC addresses to all of the adapters.
c. Consider sending an IP datagram from Host E to Host B. Suppose all of the ARP tables are up to date. Enumerate all the steps, as done for the single-router example in Section 6.4.1.
d. Repeat (c), now assuming that the ARP table in the sending host is empty (and the other tables are up to date).

a), b) See figure below.



c)

1. Forwarding table in E determines that the datagram should be routed to interface 192.168.3.002.
2. The adapter in E creates and Ethernet packet with Ethernet destination address 88-88-88-88-88-88.
3. Router 2 receives the packet and extracts the datagram. The forwarding table in this router indicates that the datagram is to be routed to 198.162.2.002.
4. Router 2 then sends the Ethernet packet with the destination address of 33-33-33-33-33-33 and source address of 55-55-55-55-55-55 via its interface with IP address of 198.162.2.003.
5. The process continues until the packet has reached Host B.

d) ARP in E must now determine the MAC address of 198.162.3.002. Host E sends out an ARP query packet within a broadcast Ethernet frame. Router 2 receives the query packet and sends to Host E an ARP response packet. This ARP response packet is carried by an Ethernet frame with Ethernet destination address 77-77-77-77-77-77.

P15. Consider Figure 6.33. Now we replace the router between subnets 1 and 2 with a switch S1, and label the router between subnets 2 and 3 as R1.

a) No. E can check the subnet prefix of Host F's IP address, and then learn that F is on the same LAN. Thus, E will not send the packet to the default router R1.

Ethernet frame from E to F:

Source IP = E's IP address

Destination IP = F's IP address

Source MAC = E's MAC address

Destination MAC = F's MAC address

b) No, because they are not on the same LAN. E can find this out by checking B's IP address.

Ethernet frame from E to R1:

Source IP = E's IP address

Destination IP = B's IP address

Source MAC = E's MAC address

Destination MAC = The MAC address of R1's interface connecting to Subnet 3.

c) Switch S1 will broadcast the Ethernet frame via both its interfaces as the received ARP frame's destination address is a broadcast address. And it learns that A resides on Subnet 1 which is connected to S1 at the interface connecting to Subnet 1. And, S1 will update its forwarding table to include an entry for Host A.

Yes, router R1 also receives this ARP request message, but R1 won't forward the message to Subnet 3.

B won't send ARP query message asking for A's MAC address, as this address can be obtained from A's query message.

Once switch S1 receives B's response message, it will add an entry for host B in its forwarding table, and then drop the received frame as destination host A is on the same interface as host B (i.e., A and B are on the same LAN segment).

Wireshark Ethernet and Arp 7.0

5.What is the value of the Ethernet source address? Is this the address of your computer, or of gaia.cs.umass.edu (Hint: the answer is no). What device has this as its Ethernet address?

The source address 00:0c:41:45:90:a8 is neither the Ethernet address of gaia.cs.umass.edu nor the address of my computer. It is the address of my Linksys router, which is the link used to get onto my subnet.

6.What is the destination address in the Ethernet frame? Is this the Ethernet address of your computer?

The destination address 00:09:5b:61:8e:6d is the address of
my computer.

7.Give the hexadecimal value for the two -byte Frame type field. What upper layer protocol does this correspond to?

The hex value for the Frame type field is 0x0800. This value corresponds to the IP protocol

8.How many bytes from the very start of the Ethernet frame does the ASCII "O" in "OK" (i.e., the HTTP response code) appear in theEthernet frame?

The ASCII "O" appears 52 bytes from the start of the Ethernet frame. Again, there are 14 bytes of Ethernet frame, and then 20 bytes of IP header followed by 20 bytes of TCP header before the HTTP data is encountered.

Uge 11

R7. Why are acknowledgments used in 802.11 but not in wired Ethernet?

Acknowledgments are used in an 802.11 network due to the high rate of errors in wireless networks due to signal loss, signal to noise ratio, fading, collisions, multi-path propagation and interference. If there were no acknowledgements sent back to the sender the sender would have no way of knowing that the receiver received the data intact and without error.

P11. In Section 7.5, one proposed solution that allowed mobile users to maintain their IP addresses as they moved among foreign networks was to have a foreign network advertise a highly specific route to the mobile user and use the existing routing infrastructure to propagate this information throughout the network. We identified scalability as one concern. Suppose that when a mobile user moves from one network to another, the new foreign network advertises a specific route to the mobile user, and the old foreign network withdraws its route. Consider how routing information propagates in a distance-vector algorithm (particularly for the case of interdomain routing among networks that span the globe).

a. Will other routers be able to route datagrams immediately to the new foreign network as soon as the foreign network begins advertising its route?

a) No. All the routers might not be able to route the datagram immediately. This is because the Distance Vector algorithm (as well as the inter-AS routing protocols like BGP) is decentralized and takes some time to terminate. So, during the time when the algorithm is still running as a result of advertisements from the new foreign network, some of the routers may not be able to route datagrams destined to the mobile node.

b. Is it possible for different routers to believe that different foreign networks contain the mobile user?

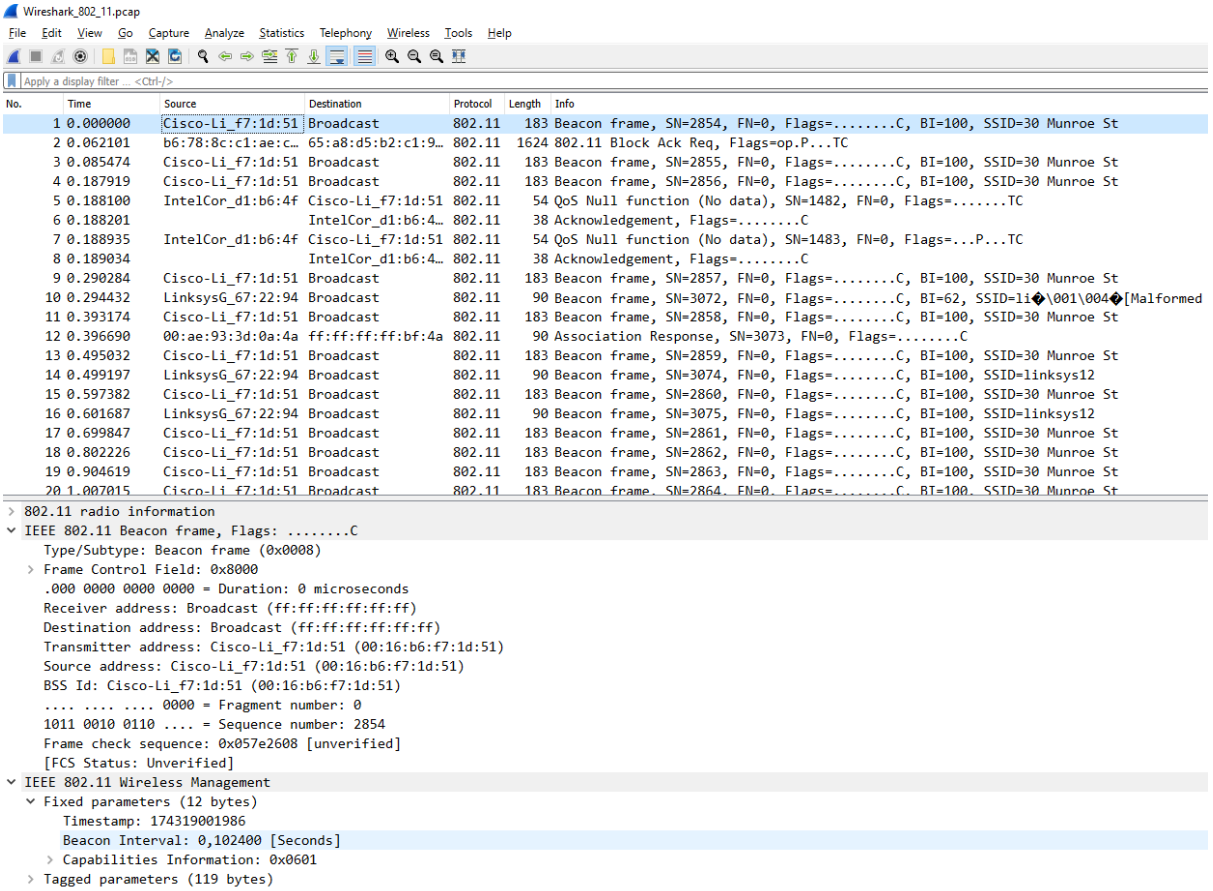
b) Yes. This might happen when one of the nodes has just left a foreign network and joined a new foreign network. In this situation, the routing entries from the old foreign network might not have been completely withdrawn when the entries from the new network are being propagated.

c. Discuss the timescale over which other routers in the network will eventually learn the path to the mobile users.

c) The time it takes for a router to learn a path to the mobile node depends on the number of hops between the router and the edge router of the foreign network for the node.

Wireshark 802.11 v7.0

2.What are the intervals of time between the transmissions of the beacon frames the linksys_ses_24086 access point? From the 30 Munroe St . access point?(Hint: this interval of time is contained in the beacon frame itself)



The beacon interval for both access points in reported in the Beacon Interval of the 802.11 wireless LAN Management frame as .1024 second s (i.e., just over 100 milliseconds). Note that the 30 Munroe St AP beacon frames show up in the trace at this regularity, but the beacons from the linsys_SES_24086 AP do not

Uge 12

R1. Operational devices such as firewalls and intrusion detection systems are used to counter attacks against an organization's network. What is the basic difference between a firewall and an intrusion detection system?

A firewall is a hardware and/or software which functions in a networked environment to block unauthorized access while permitting authorized communications. Firewall is a device and/or a software that stands between a local network and the Internet, and filters traffic that might be harmful. An Intrusion Detection System (IDS) is a software or hardware device installed on the network (NIDS) or host (HIDS) to detect and report intrusion attempts to the network.

We can think a firewall as security personnel at the gate and an IDS device is a security camera after the gate. A firewall can block connection, while a Intrusion Detection System (IDS) cannot block connection. An Intrusion Detection System (IDS) alert any intrusion attempts to the security administrator.

R3. The encryption technique itself is known—published, standardized, and available to everyone, even a potential intruder. Then where does the security of an encryption technique come from?

Encryption is foundational for a variety of technologies, but it is especially important for keeping HTTP requests and responses secure, and for authenticating website origin servers. The protocol responsible for this is called HTTPS(Hypertext Transfer Protocol Secure). A website served over HTTPS instead of HTTP will have a URL that begins with https:// instead of http://, usually represented by a secured lock in the address bar.

HTTPS uses the encryption protocol called Transport Layer Security (TLS). In the past, an earlier encryption protocol called Secure Sockets Layer (SSL) was the standard, but TLS has replaced SSL. A website that implements HTTPS will have a TLS certificate installed on its origin server.

P1. Using the monoalphabetic cipher in Figure 8.3, encode the message "This is a secret message." Decode the message "fsgg ash."

It becomes "kill him" in plaintext after decoding.