Lab I: Internet services. Telnet (SSH), E-mail, FTP

1. Objectives:

- Understand the role of protocols in networking.
- Identify each of the four layers of the OSI model TCP/IP.
- Provide a brief description of the features and operation of well-known TCP/IP applications.

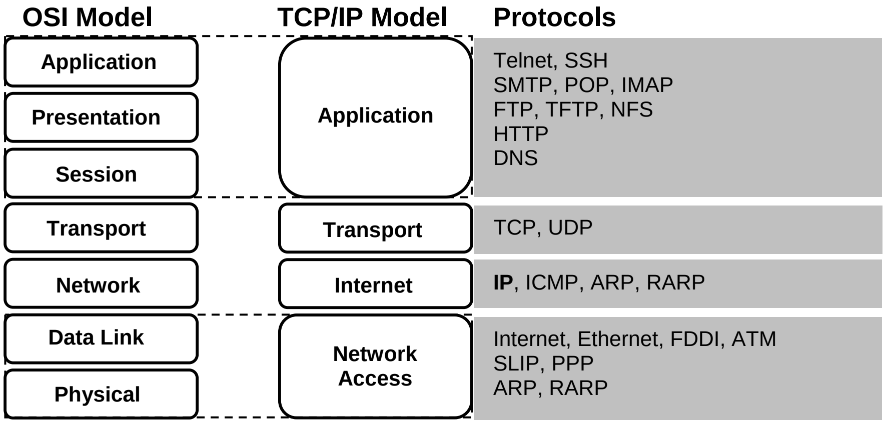
2. Communications protocols. TCP/IP model

In order for data packets to travel from a source to a destination on a network, it is important that all the devices on the network speak the same language or protocol. A data communications protocol is a set of rules or an agreement that determines the format and transmission of data.

To address the problem of network incompatibility, the International Organization for Standardization (ISO) researched networking models like Digital Equipment Corporation net (DECnet), Systems Network Architecture (SNA), and TCP/IP in order to find a generally applicable set of rules for all networks. Using this research, the ISO created a network model that helps vendors create networks that are compatible with other networks.

TCP/IP (Transmission Control Protocol/Internet Protocol)was developed as an open standard. This meant that anyone was free to use TCP/IP. This helped speed up the development of TCP/IP as a standard.

The TCP/IP model has the following four layers:



OSI Model	TCP/IP Model	Protocols
Application Presentation Session	Application	Telnet, SSH SMTP, POP, IMAP FTP, TFTP, NFS HTTP DNS
Transport	Transport	TCP, UDP
Network	Internet	**IP**, ICMP, ARP, RARP
Data Link Physical	Network Access	Internet, Ethernet, FDDI, ATM SLIP, PPP ARP, RARP

Application layer

The application layer includes the OSI (Open System Interconnection) session and presentation layer details. It handles issues of representation, encoding, and dialog control.

Some of the most commonly used application layer protocols include the following: Simple Mail Transfer Protocol (SMTP) and Post Office Protocol (POP), Hyper Text Transfer Protocol (HTTP), File Transfer Protocol (FTP), Domain Name System (DNS) etc.

Transport layer

The transport layer deals with the quality of service issues of reliability, flow control, and error correction. One of its protocols, the transmission control protocol (TCP), provides excellent and flexible ways to create reliable, well-flowing, low-error network communications.

TCP is a connection-oriented protocol. It maintains a dialogue between source and destination while packaging application layer information into units called segments. Connection-oriented does not mean that a circuit exists between the communicating computers. It does mean that Layer 4 segments travel back and forth between two hosts to acknowledge the connection exists logically for some period.

User Datagram Protocol (UDP) is the connectionless transport protocol in the TCP/IP protocol stack. UDP is a simple protocol that exchanges datagrams, without acknowledgments or guaranteed delivery. Error processing and retransmission must be handled by higher layer protocols

Internet layer

The purpose of the Internet layer is to divide TCP segments into packets and send them from any network. The packets arrive at the destination network independent of the path they took to get there. The specific protocol that governs this layer is called the Internet Protocol (IP). Best path determination and packet switching occur at this layer.

The relationship between IP and TCP is an important one. IP can be thought to point the way for the packets, while TCP provides a reliable transport.

Network Access layer

This layer is concerned with all of the components, both physical and logical, that are required to make a physical link. It includes the networking technology details, including all the details in the OSI physical and data link layers.

3. Telnet service (SSH)

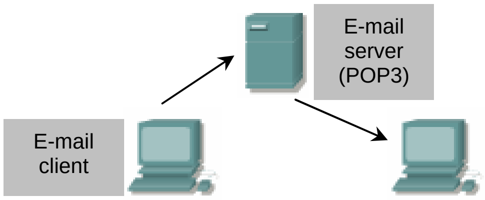
Telnet client software provides the ability to login to a remote Internet host that is running a Telnet server application and then to execute commands from the command line. A Telnet client is referred to as a local host. Telnet server, which uses special software called a daemon, is referred to as a remote host.

To make a connection from a Telnet client, the connection option must be selected. A dialog box typically prompts for a host name and terminal type. The host name is the IP address or DNS name of the remote computer. The terminal type describes the type of terminal emulation that the Telnet client should perform. The Telnet operation uses none of the processing power from the transmitting computer. Instead, it transmits the keystrokes to the remote host and sends the resulting screen output back to the local monitor. All processing and storage take place on the remote computer.

SSH (Secure Shell) is a secured version of Telnet for Linux systems.

4. E-mail systems

Email servers or MTA (Mail Transfer Agent) communicate with each other using the Simple Mail Transfer Protocol (SMTP) to send and receive mail. The SMTP protocol transports email messages in ASCII format using TCP.



When a mail server receives a message destined for a local client, it stores that message and waits for the client to collect the mail. There are several ways for mail clients to collect their mail. They can use programs that access the mail server files directly or collect their mail using one of many network protocols. The most popular mail client protocols are POP3 and IMAP4, which both use TCP to transport data. Even though mail clients use these special protocols to collect mail, they almost always use SMTP to send mail. Since two different protocols, and possibly two different servers, are used to send and receive mail, it is possible that mail clients can perform one task and not the other. Therefore, it is usually a good idea to troubleshoot e-mail sending problems separately from e-mail receiving problems.

When checking the configuration of a mail client, verify that the SMTP and POP or IMAP settings are correctly configured. A good way to test if a mail server is reachable is to Telnet to the SMTP port (25) or to the POP3 port (110). The following command format is used at the Windows command line to test the ability to reach the SMTP service on the mail server at IP address:

```
[student@tc /student]$ telnet 192.168.1.1 25
```

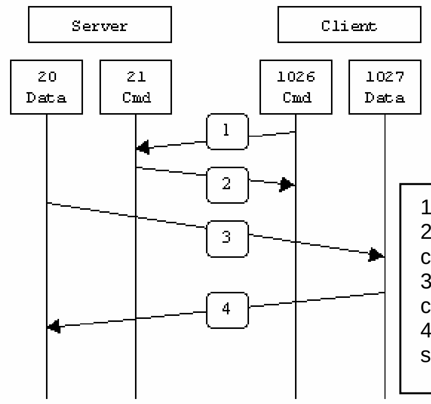
The SMTP protocol does not offer much in the way of security and does not require any authentication. Administrators often do not allow hosts that are not part of their network to use their SMTP server to send or relay mail. This is to prevent unauthorized users from using their servers as mail relays.

5. FTP

FTP is a reliable, connection-oriented service that uses TCP to transfer files between systems that support FTP. The main purpose of FTP is to transfer files from one computer to another by copying and moving files from servers to clients, and from clients to servers. When files are copied from a server, FTP first establishes a control connection (port 21) between the client and the server. Then a second connection is established (port 20), which is a link between the computers through which the data is transferred. Data transfer can occur in ASCII mode or in binary mode. These modes determine the encoding used for data file, which in the OSI model is a presentation layer task. After the file transfer has ended, the data connection terminates automatically. When the entire session of copying and moving files is complete, the command link is closed when the user logs off and ends the session.
There are two types of FTP connections : active si passive.

Active FTP

In active mode FTP the client connects from a random unprivileged port (N > 1023) to the FTP server's command port, port 21. Then, the client starts listening to port N+1 and sends the FTP command PORT N+1 to the FTP server. The server will then connect back to the client's specified data port from its local data port, which is port 20.



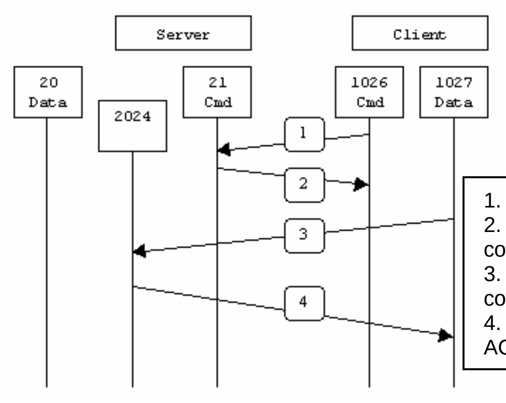
1. FTP server's port 21 from anywhere (Client initiates connection).
2. FTP server's port 21 to ports > 1023 (Server responds to client's control port).
3. FTP server's port 20 to ports > 1023 (Server initiates data connection to client's data port).
4. FTP server's port 20 from ports > 1023 (Client sends ACKs to server's data port).

In step 1, the client's command port contacts the server's command port and sends the command PORT 1027. The server then sends an ACK back to the client's command port in step 2. In step 3 the server initiates a connection on its local data port to the data port the client specified earlier. Finally, the client sends an ACK back as shown in step 4.
The main problem with active mode FTP actually falls on the client side. The FTP client doesn't make the actual connection to the data port of the server--it simply tells the server what port it is listening on and the server connects back to the specified port on the client. From the client side firewall this appears to be an outside system initiating a connection to an internal client--something that is usually blocked.

Passive FTP

In order to resolve the issue of the server initiating the connection to the client a different method for FTP connections was developed. This was known as passive mode, or PASV, after the command used by the client to tell the server it is in passive mode.
In passive mode FTP the client initiates both connections to the server, solving the problem of firewalls filtering the incoming data port connection to the client from the server. When opening an FTP connection, the client opens two random unprivileged ports locally (N > 1023 and N+1). The first port contacts the server on port 21, but instead of then issuing a PORT command and allowing the server to connect back to its data port, the client will issue the PASV command. The result of this is that the server then opens a

random unprivileged port (P > 1023) and sends the PORT P command back to the client. The client then initiates the connection from port N+1 to port P on the server to transfer data.



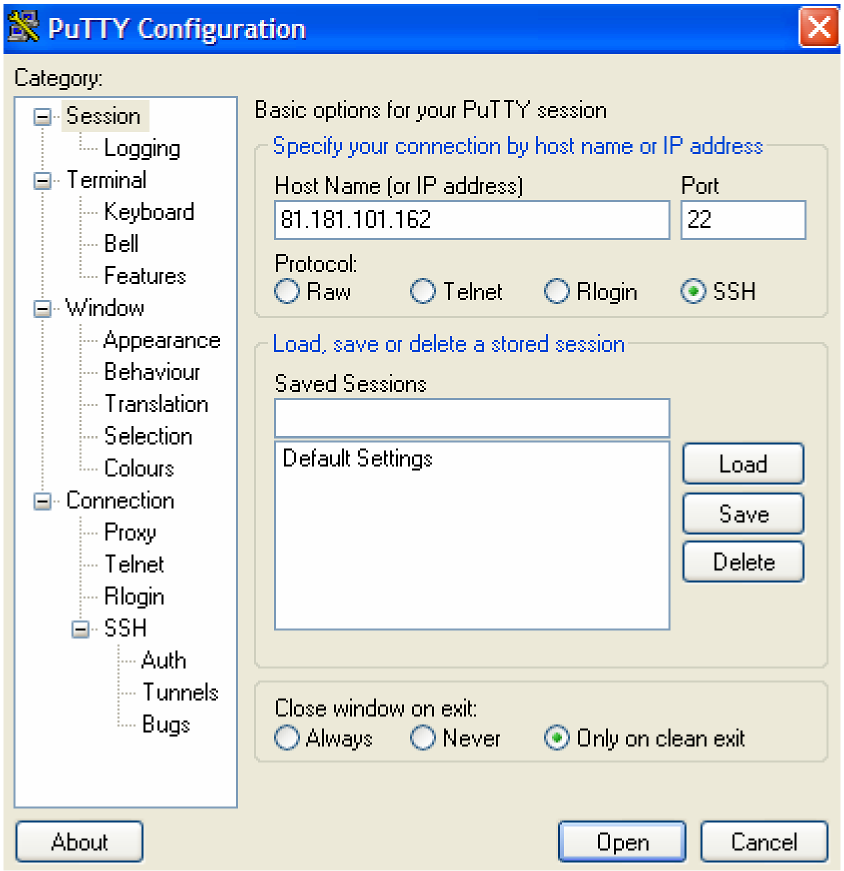
1. FTP server's port 21 from anywhere (Client initiates connection).
2. FTP server's port 21 to ports > 1023 (Server responds to client's control port).
3. FTP server's ports > 1023 from anywhere (Client initiates data connection to random port specified by server).
4. FTP server's ports > 1023 to remote ports > 1023 (Server sends ACKs (and data) to client's data port).

In step 1, the client contacts the server on the command port and issues the PASV command. The server then replies in step 2 with PORT 2024, telling the client which port it is listening to for the data connection. In step 3 the client then initiates the data connection from its data port to the specified server data port. Finally, the server sends back an ACK in step 4 to the client's data port.

While passive mode FTP solves many of the problems from the client side, it opens up a whole range of problems on the server side. The biggest issue is the need to allow any remote connection to high numbered ports on the server. Fortunately, many FTP daemons, including the popular WU-FTPD allow the administrator to specify a range of ports which the FTP server will use.

5. Practical work

1. **Objective :** connect to a distant host usingSSH protocols and test some UNIX commands.
Launch *putty.exe* application (Telnet client - port 23, SSH client – port 22). Select protocol SSH radio button and fill the field Host Name (or IP address) with (*81.181.101.162 – a Linux host*). The connection is made for the account *student.*



PuTTY Configuration

Basic options for your PuTTY session

Specify your connection by host name or IP address

Host Name (or IP address): 81.181.101.162

Port: 22

Protocol: ○ Raw ○ Telnet ○ Rlogin ● SSH

Load, save or delete a stored session

Saved Sessions: Default Settings

Close window on exit: ○ Always ○ Never ● Only on clean exit

After the connection is made a terminal will be open with Linux command line.
Test the following UNIX commands : *finger, uname, man, ls, cd, pwd, cat, mkdir, rmdir, echo, less, more, vi, ps, exit.*
What about the */etc/services* file?

2. **Objective :** Send an e-mail message using SMTP commands and read the content of the e-mail box for *student* account.
The following command format is used at the Linux command line:

```
[student@tc /student]$telnet localhost 25
```

to connect to the remote host *localhost* on port 25 (SMTP port). The server responds with the following message (e-mail application dependent) :

```
Trying 127.0.0.1…
Connected to localhost.
Escape character is '^]'.
220 linux.site ESMTP Postfix
```

Next, to send an e-mail message type the following commands :

```
HELO localhost
250 linux.site
MAIL FROM: student@localhost
250 Ok
RCPT TO: student@localhost
250 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
This is an e-mail message from student@localhost for student@localhost.
.
250 Ok: queued as 873D222485
QUIT
221 Bye
Connection closed by foreign host.
[student@tc /student]$
```

To read the content of the e-mail box for *student* account use the *mail* command on Linux command line. The *mail* is a very simple, text e-mail client for Linux OS.

3. **Objective :** Connect and transfer files using FTP.
The following command format is used at the Linux command line :

```
[student@tc /student]$ ftp –d localhost
```

The server responds with the following message (ftp application dependent) :

```
Trying 127.0.0.1…                          ftp> ls
Connected to localhost.                    ftp: setsockopt (ignored): Permission denied
220 (vsFTPd  2.0.3).                       ---> EPSV
Name (localhost:student): anonymous        229 Entering Extended Passive Mode (|||39446|)
---> USER anonymous                        ---> LIST
331 Please specify the password.           150 Here comes the directory listing.
Password: student@localhost                -rwxr-xr-x 1 0 0 101 Jul 19 2006 fisier.txt
---> PASS XXX                              226 Directory send OK.
230 Login successful.                      ftp> quit
---> SYST                                  ---> QUIT
215 Unix Ttype: L8                         221 Goodbye.
Remote system is UNIX.                     [student@tc /student]$
Using binary mode to transfer files.
```

The FTP commands are outlined with *italic* characters.

Lab II: Monitoring Internet Connections (TCP/IP)

1. Objectives:

- Identify some applications for network configuration.
- Use basic testing procedures to test the Internet connection.
- Understand the physical connection that has to take place for a computer to connect to the Internet.

2. Connecting to Internet

The Internet is the largest data network on earth. The Internet consists of a multitude of interconnected networks both large and small. At the edge of this giant network is the individual consumer computer.

Connection to the Internet can be broken down into the physical connection, the logical connection, and the application.

A physical connection is made by connecting a specialized expansion card such as a modem or a network interface card (NIC) from a computer (PC) to a network. The physical connection is used to transfer signals between PCs within the local network and to remote devices on the Internet.

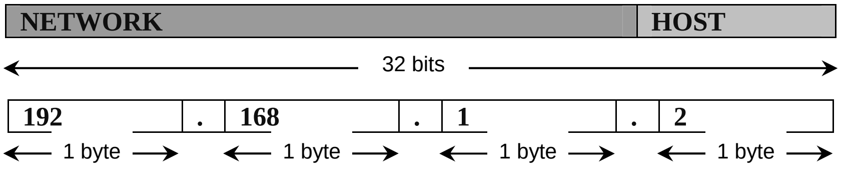
The logical connection uses standards called protocols. A protocol is a formal description of a set of rules and conventions that govern how devices on a network communicate. Connections to the Internet may use multiple protocols. The Transmission Control Protocol/Internet Protocol (TCP/IP) suite is the primary protocol used on the Internet. TCP/IP is a suite of protocols that work together to transmit data.

e application that interprets the data and displays the information in an understandable form is the last part of the connection. Applications work with protocols to send and receive data across the Internet. A web browser displays Hypertext Markup Language (HTML) as a web page. File Transfer Protocol (FTP) is used to download files and programs from the Internet. Web browsers also use proprietary plug-in applications to display special data types such as movies or flash animations.

3. Internet Addresses (IP)

Each computer in a TCP/IP network must be given a unique identifier, or IP address.

An IP address is a 32-bit sequence of 1s and 0s. Figure 3 shows a sample 32-bit number. To make the IP address easier to use, the address is usually written as four decimal numbers separated by periods (ex. 192.168.1.2), Every IP address has two parts. One part identifies the network where the system is connected, and a second part identifies that particular system on the network. Each octet ranges from 0 to 255.



When IP addresses are assigned to computers, some of the bits on the left side of the 32-bit IP number represent a network. The number of bits designated depends on the address class. The bits left over in the 32-bit IP address identify a particular computer on the network. A computer is referred to as the host. The IP address of a computer consists of a network and a host part that represents a particular computer on a particular network.

To inform a computer how the 32-bit IP address has been split, a second 32-bit number called a subnetwork mask is used. This mask is a guide that indicates how the IP address should be interpreted by identifying how many of the bits are used to identify the network of the computer. The subnetwork mask sequentially fills in the 1s from the left side of the mask. A subnet mask will always be all 1s until the network address is identified and then be all 0s from there to the right most bit of the mask. The bits in the subnet mask that are 0 identify the computer or host on that network.

4. TCP/UDP port numbers

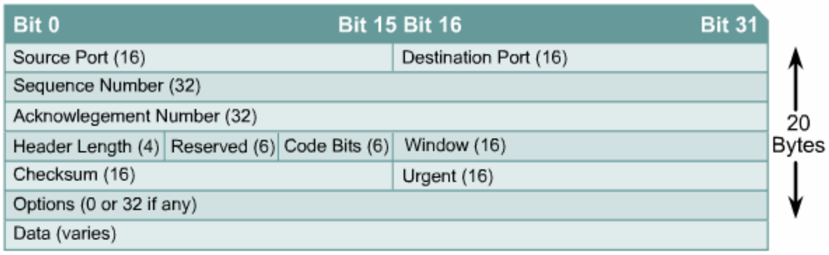
Both TCP and UDP use port (socket) numbers to pass information to the upper layers. Port numbers are used to keep track of different conversations crossing the network at the same time.
Application software developers agree to use well-known port numbers that are issued by the Internet Assigned Numbers Authority (IANA). [1]Any conversation bound for the FTP application uses the standard port numbers 20 and 21. Port 20 is used for the data portion and port 21 is used for control. Conversations that do not involve an application with a well-known port number are assigned port numbers randomly from within a specific range above 1023. Some ports are reserved in both TCP and UDP, but applications might not be written to support them. [2]Port numbers have the following assigned ranges:

- Numbers below 1024 are considered well-known ports numbers
- Numbers above 1024 are dynamically assigned ports numbers
- Registered port numbers are those registered for vendor-specific applications. Most of these are above 1024

5. Transport protocols: TCP and UDP. Connection establishment.

Transmission Control Protocol (TCP) is a connection-oriented Layer 4 protocol that provides reliable full-duplex data transmission. TCP is part of the TCP/IP protocol stack. In a connection-oriented environment, a connection is established between both ends before the transfer of information can begin. TCP is responsible for breaking messages into segments, reassembling them at the destination station, resending anything that is not received, and reassembling messages from the segments. TCP supplies a virtual circuit between end-user applications.

TCP segment format:



Bit 0		Bit 15	Bit 16	Bit 31
Source Port (16)			Destination Port (16)	
Sequence Number (32)				
Acknowlegement Number (32)				
Header Length (4)	Reserved (6)	Code Bits (6)	Window (16)	
Checksum (16)			Urgent (16)	
Options (0 or 32 if any)				
Data (varies)				

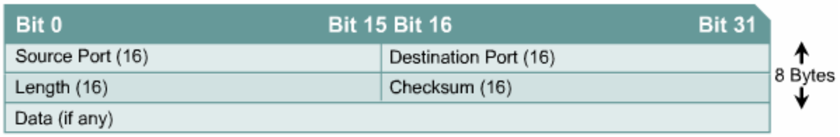
20 Bytes

where:

- **Source port** – Number of the calling port
- **Destination port**– Number of the called port
- **Sequence number** – Number used to ensure correct sequencing of the arriving data
- **Acknowledgment number** – Next expected TCP octet
- **Header length HLEN** – Number of 32-bit words in the header
- **Reserved** – Set to zero
- **Code bits** – Control functions, such as setup and termination of a session
- **Window** – Number of octets that the sender is willing to accept
- **Checksum** – Calculated checksum of the header and data fields
- **Urgent pointer** – Indicates the end of the urgent data
- **Options** One option currently defined, maximum TCP segment size
- **Data** – Upper-layer protocol data.

User Datagram Protocol (UDP) is the connectionless transport protocol in the TCP/IP protocol stack. UDP is a simple protocol that exchanges datagrams, without acknowledgments or guaranteed delivery. Error processing and retransmission must be handled by higher layer protocols.
UDP uses no windowing or acknowledgments so reliability, if needed, is provided by application layer protocols. UDP is designed for applications that do not need to put sequences of segments together

UDP segment format:



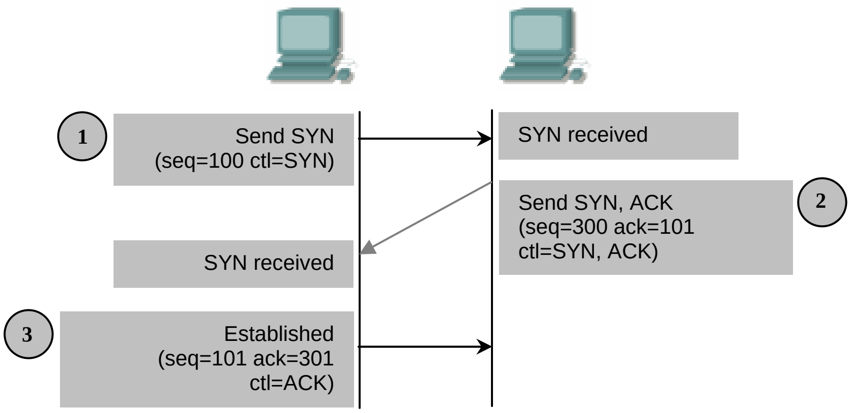
Bit 0	Bit 15 Bit 16 Bit 31	
Source Port (16)	Destination Port (16)	8 Bytes
Length (16)	Checksum (16)	
Data (if any)		

unde :

- **Source port** – Number of the calling port
- **Destination port** – Number of the called port
- **Length** – Number of bytes including header and data
- **Checksum** – Calculated checksum of the header and data fields
- **Data** – Upper-layer protocol data.

TCP is a connection-oriented protocol. TCP requires connection establishment before data transfer begins. For a connection to be established or initialized, the two hosts must synchronize their Initial Sequence Numbers (ISNs). Synchronization is done through an exchange of connection establishing segments that carry a control bit called SYN, for synchronize, and the ISNs. Segments that carry the SYN bit are also called “SYNs". This solution requires a suitable mechanism for picking an initial sequence number and a slightly involved handshake to exchange the ISNs.

The synchronization requires each side to send its own initial sequence number and to receive a confirmation of exchange in an acknowledgment (ACK) from the other side. Each side must also receive the INS from the other side and send a confirming ACK. The sequence is as follows:

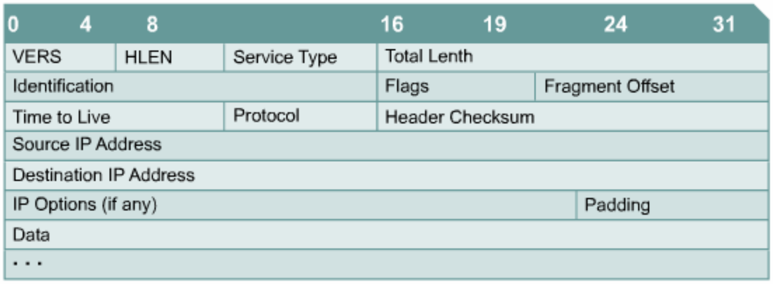


1. A→B SYN—(A) initial sequence number is X, ACK number is 0, SYN bit is set, but ACK bit is not set.
2. B→A ACK—(A) sequence number is X + 1, (B) initial sequence number is Y, and SYN and ACK bit are set.
3. A→B ACK—(B) sequence number is Y + 1, (A) sequence number is X + 1, the ACK bit is set, but the SYN bit is not set.

6. IP Protocol

The purpose of the Internet layer is to select the best path through the network for packets to travel. The main protocol that functions at this layer is the Internet Protocol (IP).

IP header:



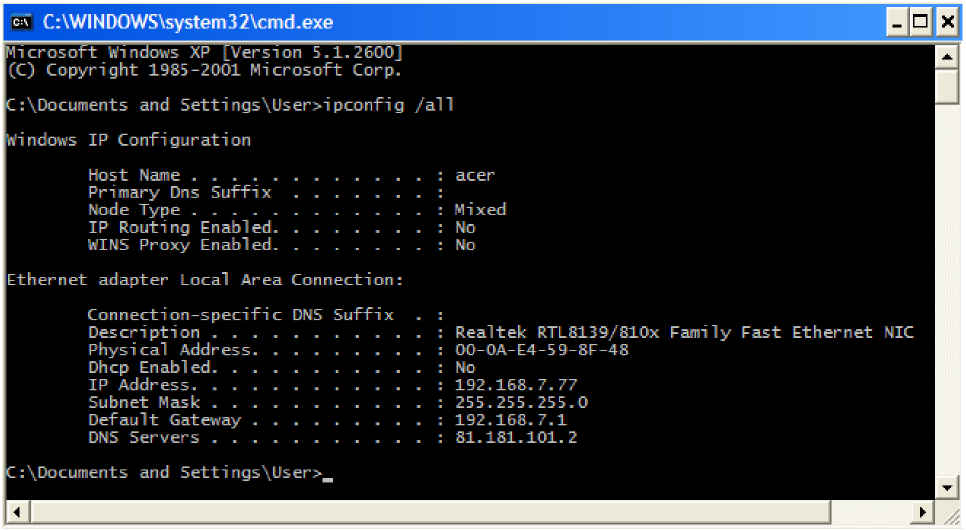
0	4	8	16	19	24	31
VERS	HLEN	Service Type	Total Lenth			
Identification			Flags	Fragment Offset		
Time to Live		Protocol	Header Checksum			
Source IP Address						
Destination IP Address						
IP Options (if any)					Padding	
Data						
. . .						

where:

- **VERS** – Indicates the version of IP currently used; four bits. If the version field is different than the IP version of the receiving device, that device will reject the packets.
- **IP header length HLEN** – Indicates the datagram header length in 32-bit words. This is the total length of all header information, accounting for the two variable-length header fields.
- **Type-of-service TOS** – Specifies the level of importance that has been assigned by a particular upper-layer protocol, eight bits.
- **Total length** – Specifies the length of the entire packet in bytes, including data and header, 16 bits. To get the length of the data payload subtract the HLEN from the total length.
- **Identification** – Contains an integer that identifies the current datagram, 16 bits. This is the sequence number.
- **Flags** – A three-bit field in which the two low-order bits control fragmentation. One bit specifies whether the packet can be fragmented, and the other specifies whether the packet is the last fragment in a series of fragmented packets.
- **Fragment offset** – Used to help piece together datagram fragments, 13 bits. This field allows the previous field to end on a 16-bit boundary.
- **Time-to-live TTL** – A field that specifies the number of hops a packet may travel. This number is decreased by one as the packet travels through a router. When the counter reaches zero the packet is discarded. This prevents packets from looping endlessly.
- **Protocol** – indicates which upper-layer protocol, such as TCP or UDP, receives incoming packets after IP processing has been completed, eight bits.
- **Header checksum** – helps ensure IP header integrity, 16 bits.
- **Source address** – specifies the sending node IP address, 32 bits.
- **Destination address** – specifies the receiving node IP address, 32 bits.
- **Options** – allows IP to support various options, such as security, variable length.
- **Padding** – extra zeros are added to this field to ensure that the IP header is always a multiple of 32 bits.
- **Data** – contains upper-layer information, variable length up to 64 Kb.

7. Practical work

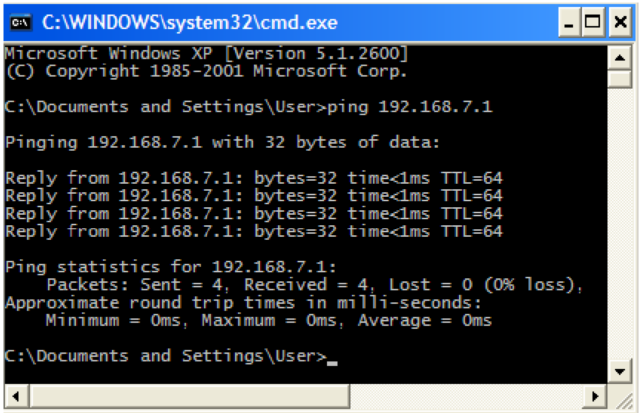
1. **Objective :** Gather information including connection, host name, Layer 2 MAC address and Layer 3 TCP/IP network address information.
Using the taskbar, choose **Start** then **Run**. Type **ipconfig /all** and press the **Enter** key. The figure shows the detailed IP configuration screen:



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\User>ipconfig /all

Windows IP Configuration

        Host Name . . . . . . . . . . . . : acer
        Primary Dns Suffix  . . . . . . . :
        Node Type . . . . . . . . . . . . : Mixed
        IP Routing Enabled. . . . . . . . : No
        WINS Proxy Enabled. . . . . . . . : No

Ethernet adapter Local Area Connection:

        Connection-specific DNS Suffix  . :
        Description . . . . . . . . . . . : Realtek RTL8139/810x Family Fast Ethernet NIC
        Physical Address. . . . . . . . . : 00-0A-E4-59-8F-48
        Dhcp Enabled. . . . . . . . . . . : No
        IP Address. . . . . . . . . . . . : 192.168.7.77
        Subnet Mask . . . . . . . . . . . : 255.255.255.0
        Default Gateway . . . . . . . . . : 192.168.7.1
        DNS Servers . . . . . . . . . . . : 81.181.101.2

C:\Documents and Settings\User>_
```

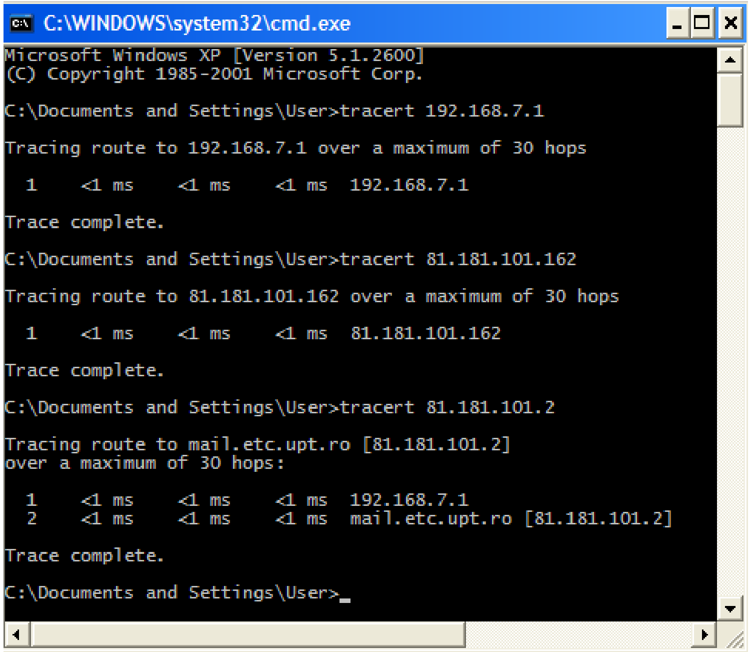
2. **Objective :** Testing connectivity with *ping*.
Ping is a utility used to verify Internet connectivity. The *ping* command works by sending multiple IP packets (ICMP Echo Request) to a specified destination. Each packet sent is a request for a reply (ICMP Echo Reply). The output response for a ping contains the success ratio and round-trip time to the destination. From this information, it is possible to determine if there is connectivity to a destination. The **ping** command is used to test the NIC transmit/receive function, the TCP/IP configuration, and network connectivity.

ping 127.0.0.1 - This ping is unique and is called an internal *loopback* test. It verifies the operation of the TCP/IP stack and NIC transmit/receive function.



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\User>ping 192.168.7.1

Pinging 192.168.7.1 with 32 bytes of data:

Reply from 192.168.7.1: bytes=32 time<1ms TTL=64
Reply from 192.168.7.1: bytes=32 time<1ms TTL=64
Reply from 192.168.7.1: bytes=32 time<1ms TTL=64
Reply from 192.168.7.1: bytes=32 time<1ms TTL=64

Ping statistics for 192.168.7.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 0ms, Average = 0ms

C:\Documents and Settings\User>_
```

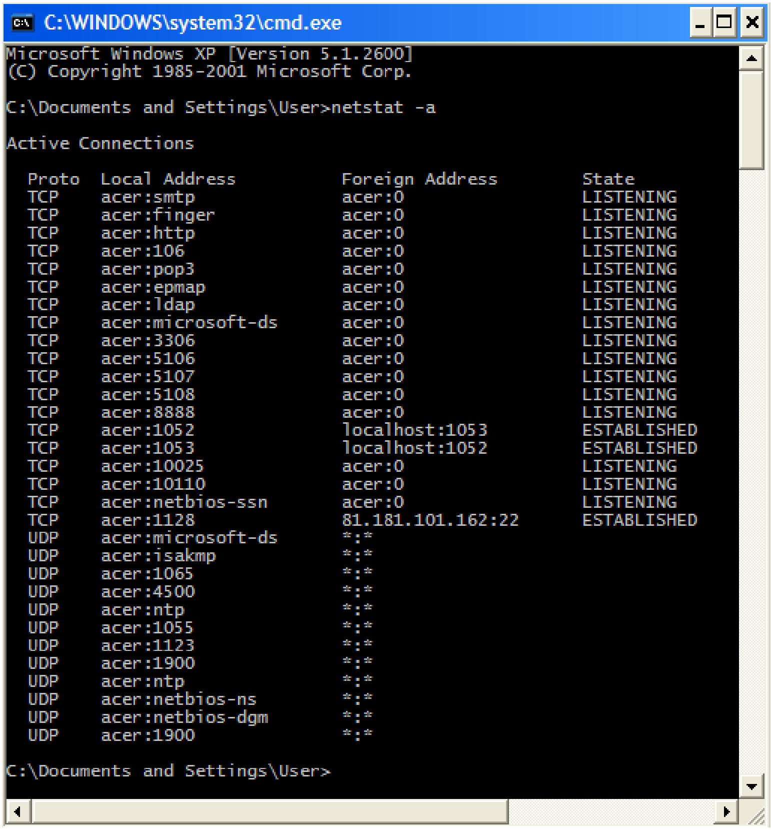
3. **Objective :** Learn to use the Trace Route (*tracert*) command from a workstation.
Tracert uses the same echo requests and replies as the *ping* command but in a slightly different way.



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\User>tracert 192.168.7.1

Tracing route to 192.168.7.1 over a maximum of 30 hops

  1    <1 ms    <1 ms    <1 ms  192.168.7.1

Trace complete.

C:\Documents and Settings\User>tracert 81.181.101.162

Tracing route to 81.181.101.162 over a maximum of 30 hops

  1    <1 ms    <1 ms    <1 ms  81.181.101.162

Trace complete.

C:\Documents and Settings\User>tracert 81.181.101.2

Tracing route to mail.etc.upt.ro [81.181.101.2]
over a maximum of 30 hops:

  1    <1 ms    <1 ms    <1 ms  192.168.7.1
  2    <1 ms    <1 ms    <1 ms  mail.etc.upt.ro [81.181.101.2]

Trace complete.

C:\Documents and Settings\User>_
```

The preceding figure shows the listings of all routers the *tracert* requests had to pass through to get to the destination. Each router represents a point where one network connects to another network and the packet was forwarded through.

4. **Objective :** Gather information including connections, routing tables, NIC statistics.
The application used is *netstat*. Test the following options : *-a*, *-n*,*-b*, *-r*, *-e*.



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\User>netstat -a

Active Connections

  Proto  Local Address          Foreign Address        State
  TCP    acer:smtp              acer:0                 LISTENING
  TCP    acer:finger            acer:0                 LISTENING
  TCP    acer:http              acer:0                 LISTENING
  TCP    acer:106               acer:0                 LISTENING
  TCP    acer:pop3              acer:0                 LISTENING
  TCP    acer:epmap             acer:0                 LISTENING
  TCP    acer:ldap              acer:0                 LISTENING
  TCP    acer:microsoft-ds      acer:0                 LISTENING
  TCP    acer:3306              acer:0                 LISTENING
  TCP    acer:5106              acer:0                 LISTENING
  TCP    acer:5107              acer:0                 LISTENING
  TCP    acer:5108              acer:0                 LISTENING
  TCP    acer:8888              acer:0                 LISTENING
  TCP    acer:1052              localhost:1053         ESTABLISHED
  TCP    acer:1053              localhost:1052         ESTABLISHED
  TCP    acer:10025             acer:0                 LISTENING
  TCP    acer:10110             acer:0                 LISTENING
  TCP    acer:netbios-ssn       acer:0                 LISTENING
  TCP    acer:1128              81.181.101.162:22      ESTABLISHED
  UDP    acer:microsoft-ds      *:*
  UDP    acer:isakmp            *:*
  UDP    acer:1065              *:*
  UDP    acer:4500              *:*
  UDP    acer:ntp               *:*
  UDP    acer:1055              *:*
  UDP    acer:1123              *:*
  UDP    acer:1900              *:*
  UDP    acer:ntp               *:*
  UDP    acer:netbios-ns        *:*
  UDP    acer:netbios-dgm       *:*
  UDP    acer:1900              *:*

C:\Documents and Settings\User>
```

Lab III: Configuring Ethernet Networks

1. Goals

- To understand the layers described by the standard IEEE 802.3 and the way that the protocol CSMA/CD works like;
- To identify the entities composing an Ethernet network
- To gain knowledge about configuring and managing Ethernet networks using Windows operating systems

2. Ethernet technologies

The term Ethernet refers to the family of products associated with Local Area Networks described by the standard 802.3, which defines the physical layer and part of the data link layer from the OSI-ISO reference model. Several types of Ethernet networks are currently implemented, using twisted pair and optical fiber as transmission medium.

Ethernet LANs consist of network nodes and interconnecting media. The network nodes fall into two major classes:

• **Data terminal equipment (DTE)—**Devices that are either the source or the destination of data frames. DTEs are typically devices such as PCs, workstations, file servers, or print servers that, as a group, are all often referred to as end stations.

• **Data communication equipment (DCE)**—Intermediate network devices that receive and forward frames across the network. DCEs may be either standalone devices such as repeaters, network switches, and routers, or communications interface units such as interface cards and modems.

2.1 Ethernet network topologies

LANs take on many topological configurations, but regardless of their size or complexity, all will be a combination of only three basic interconnection structures or network building blocks.

The simplest structure is the point-to-point interconnection, shown in figure 1. Only two network units are involved, and the connection may be DTE-to-DTE, DTE-to-DCE, or DCE-to-DCE. The cable in point-to-point interconnections is known as a network link. The maximum allowable length of the link depends on the type of cable and the transmission method that is used.

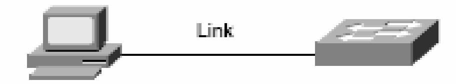


Fig. 1: Point-to-point interconnection

Coaxial bus structure

The original Ethernet networks were implemented with a coaxial bus structure, as shown in figure 2. Segment lengths were limited to 500 meters, and up to 100 stations could be connected to a single segment. Individual segments could be interconnected with repeaters, as long as multiple paths did not exist between any two stations on the network and the number of DTEs did not exceed 1024. The total path distance between the most-distant pair of stations was also not allowed to exceed a maximum prescribed value.

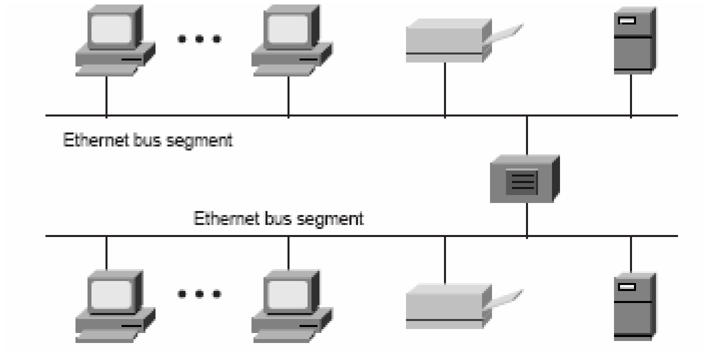


Fig.2: Bus topology on coaxial cable

Although new networks are no longer connected in a bus configuration, some older bus-connected networks still exist and are still useful.

Star topology

Since the early 1990s, the network configuration of choice has been the star-connected topology, shown in figure 3. The central network unit is either a multiport repeater (also known as a hub) or a network switch. All connections in a star network are point-to-point links implemented with either twisted-pair or optical fiber cable.

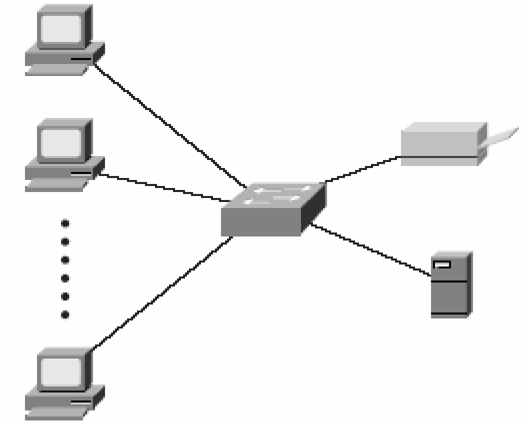


Figure 3: Principles of star topology

2.2 IEEE 802.3 and its relation with the ISO reference model

Figure 4 shows the IEEE 802.3 logical layers and their relationship to the OSI reference model. As with all IEEE 802 protocols, the ISO data link layer is divided into two IEEE 802 sublayers, the Media Access Control (MAC) sublayer and the MAC-client sublayer. The IEEE 802.3 physical layer corresponds to the ISO physical layer.

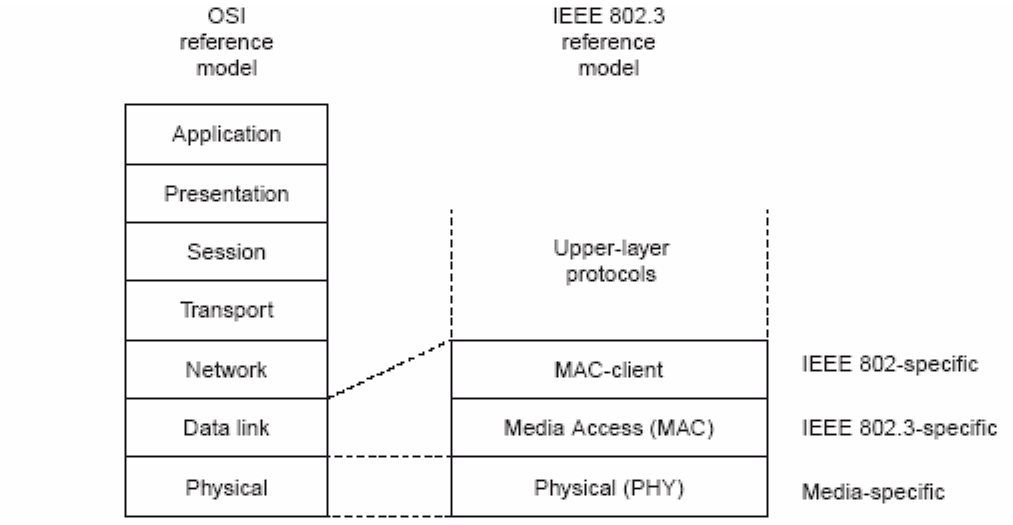


Figure 4: Etherenet's logical relationship to ISO reference model

The MAC-client sublayer may be one of the following:

- Logical Link Control (LLC), if the unit is a DTE. This sublayer provides the interface between the Ethernet MAC and the upper layers in the protocol stack of the end station. The LLC sublayer is defined by IEEE 802.2 standards.
- Bridge entity, if the unit is a DCE. Bridge entities provide LAN-to-LAN interfaces between LANs that use the same protocol (for example, Ethernet to Ethernet) and also between different protocols (for example, Ethernet to Token Ring). Bridge entities are defined by IEEE 802.1 standards.

The MAC layer controls the node’s access to the network media and is specific to the individual protocol. All IEEE 802.3 MACs must meet the same basic set of logical requirements, regardless of whether they include one or more of the defined optional protocol extensions. The only requirement for basiccommunication (communication that does not require optional protocol extensions) between two network nodes is that both MACs must support the same transmission rate.

The 802.3 physical layer is specific to the transmission data rate, the signal encoding, and the type of media interconnecting the two nodes. Gigabit Ethernet, for example, is defined to operate over either twisted-pair or optical fiber cable, but each specific type of cable or signal-encoding procedure requires a different physical layer implementation.

3. Ethernet frame structure

The IEEE 802.3 standard defines a basic data frame format that is required for all MAC implementations and several additional optional formats that are used to extend the protocol’s basic capability. The basic data frame format contains the seven fields shown in figure 5.

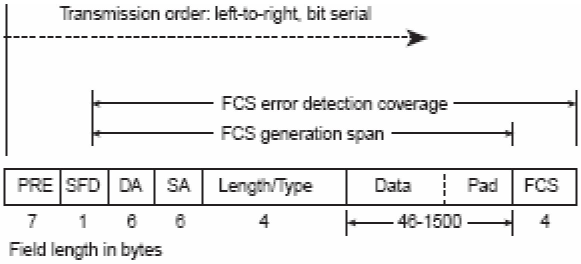


Figure 5: Ethernet frame structure

- **Preamble (PRE)—**Consists of 7 bytes. The PRE is an alternating pattern of ones and zeros that tells receiving stations that a frame is coming, and that provides a means to synchronize the frame-reception portions of receiving physical layers with the incoming bit stream.
- **Start-of-frame delimiter (SOF)—**Consists of 1 byte. The SOF is an alternating pattern of ones and zeros, ending with two consecutive 1-bits indicating that the next bit is the left-most bit in the left-most byte of the destination address.
- **Destination address (DA)—**Consists of 6 bytes. The DA field identifies which station(s) should receive the frame. The left-most bit in the DA field indicates whether the address is an individual address (indicated by a 0) or a group address (indicated by a 1). The second bit from the left indicates whether the DA is globally administered (indicated by a 0) or locally administered (indicated by a 1). The remaining 46 bits are a uniquely assigned value that identifies a single station, a defined group of stations, or all stations on the network.
- **Source addresses (SA)—**Consists of 6 bytes. The SA field identifies the sending station. The SA is always an individual address and the left-most bit in the SA field is always 0.

Individual addresses are also known as unicast addresses because they refer to a single MAC and are assigned by the NIC manufacturer from a block of addresses allocated by the IEEE. Group addresses (a.k.a. multicast addresses) identify the end stations in a workgroup and are assigned by the network manager. A special group address (all 1s—the broadcast address) indicates all stations on the network.

- **Length/Type—**Consists of 2 bytes. This field indicates either the number of MAC-client data bytes that are contained in the data field of the frame, or the frame type ID if the frame is assembled using an optional format. If the Length/Type field value is less than or equal to 1500, the number of LLC bytes in the Data field is equal to the Length/Type field value. If the Length/Type field value is greater than 1536, the frame is an optional type frame, and the Length/Type field value identifies the particular type of frame being sent or received.
- **Data—**Is a sequence of *n* bytes of any value, where *n* is less than or equal to 1500. If the length of the Data field is less than 46, the Data field must be extended by adding a filler (a pad) sufficient to bring the Data field length to 46 bytes.
- **Frame check sequence (FCS)—**Consists of 4 bytes. This sequence contains a 32-bit cyclic redundancy check (CRC) value, which is created by the sending MAC and is recalculated by the receiving MAC to check for damaged frames. The FCS is generated over the DA, SA, Length/Type, and Data fields.

4. CSMA/CD and the multiple access to the communication medium

The CSMA/CD protocol was originally developed as a means by which two or more stations could share a common media in a switch-less environment when the protocol does not require central arbitration, access tokens, or assigned time slots to indicate when a station will be allowed to transmit. Each Ethernet MAC determines for itself when it will be allowed to send a frame.

The CSMA/CD access rules are summarized by the protocol's acronym:

- **Carrier sense—**Each station continuously listens for traffic on the medium to determine when gaps between frame transmissions occur.
- **Multiple access—**Stations may begin transmitting any time they detect that the network is quiet (there is no traffic).
- **Collision detect—**If two or more stations in the same CSMA/CD network (collision domain) begin transmitting at approximately the same time, the bit streams from the transmitting stations will interfere (collide) with each other, and both transmissions will be unreadable. If that happens, each transmitting station must be capable of detecting that a collision has occurred before it has finished sending its frame.

Each must stop transmitting as soon as it has detected the collision and then must wait a quasirandom length of time (determined by a back-off algorithm) before attempting to retransmit the frame.

The worst-case situation occurs when the two most-distant stations on the network both need to send a frame and when the second station does not begin transmitting until just before the frame from the first station arrives. The collision will be detected almost immediately by the second station, but it will not be detected by the first station until the corrupted signal has propagated all the way back to that station.

The maximum time that is required to detect a collision (the collision window, or "slot time") is approximately equal to twice the signal propagation time between the two most-distant stations on the network.

This means that both the minimum frame length and the maximum collision diameter are directly related to the slot time. Longer minimum frame lengths translate to longer slot times and larger collision diameters; shorter minimum frame lengths correspond to shorter slot times and smaller collision diameters.

The trade-off was between the need to reduce the impact of collision recovery and the need for network diameters to be large enough to accommodate reasonable network sizes.

5. Network devices: repeaters, hubs and switches

In the early Ethernet, the electrical signals transmitted within the network were regenerated using repeaters and, a little bit later hubs. Such a device was receives the signal on one of its ports (a hub is oftentimes referred to as a multi-port repeater) and broadcasts it to all the other ports (excepting the one used for reception). This is the reason why all DTEs could compete with each other (the signals transmitted from any two stations could collide).

The big step forward (which brought the success of Ethernet versus Token Ring competitors) was done when relatively low-priced network switches became available on the market shortly after the mid-1990s and essentially made network repeaters obsolete for large networks. Although repeaters can accept only one frame at a time and then send it to all active ports (except the port on which it is being received), switches are equipped with the following:

- MAC-based ports with I/O frame buffers that effectively isolate the port from traffic being sent at the same time to or from other ports on the switch
- Multiple internal data paths that allow several frames to be transferred between different ports at the same time.

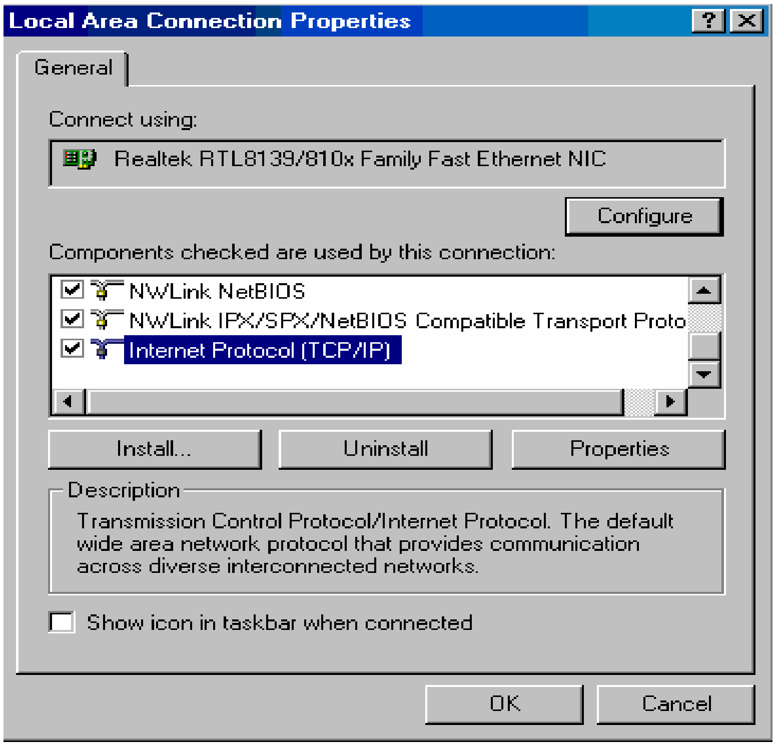
These may seem like small differences, but they produce a major effect in network operation. Because each port provides access to a high-speed network bridge (the switch), the collision domain in the network is reduced to a series of small domains in which the number of participants is reduced to two—the switch port and the connected NICs. Furthermore, because each participant is now in a private collision domain, his or her available bandwidth has not only been markedly increased, it was also done without having to change the link speed.

Consider, for example, a 48-station workgroup with a couple of large file servers and several network printers on a 100-Mbps CSMA/CD network. The average available bandwidth, not counting interframe gaps and collision recovery, would be 100 ¸ 50 = 2 Mbps (network print servers do not generate network traffic). On the other hand, if the same workgroup were still on a 10Base-T network in which the repeaters had been replaced with network switches, the bandwidth available to each user would be 10 Mbps.

6. Practical work

1. Goal: to configure the network settings of a PC

In order to prepare the PC you are working with for a connection to an Ethernet LAN (as a particular example our lab's LAN) the required steps are the following: right click on the icon My Network Places, choose Properties-Local Area Network, right clic again and select Properties. The execution of this sequence will open the window below:



Connect using:

Realtek RTL8139/810x Family Fast Ethernet NIC

Components checked are used by this connection:

- ☑ NWLink NetBIOS
- ☑ NWLink IPX/SPX/NetBIOS Compatible Transport Proto
- ☑ Internet Protocol (TCP/IP)

Description

Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.

☐ Show icon in taskbar when connected

Figure 6: Local network settings configuration

A double-click on the TCP/IP line will open a new window, allowing you to establish the IP settings for your work station. The IP addressing scheme used in our LAN will be the following:

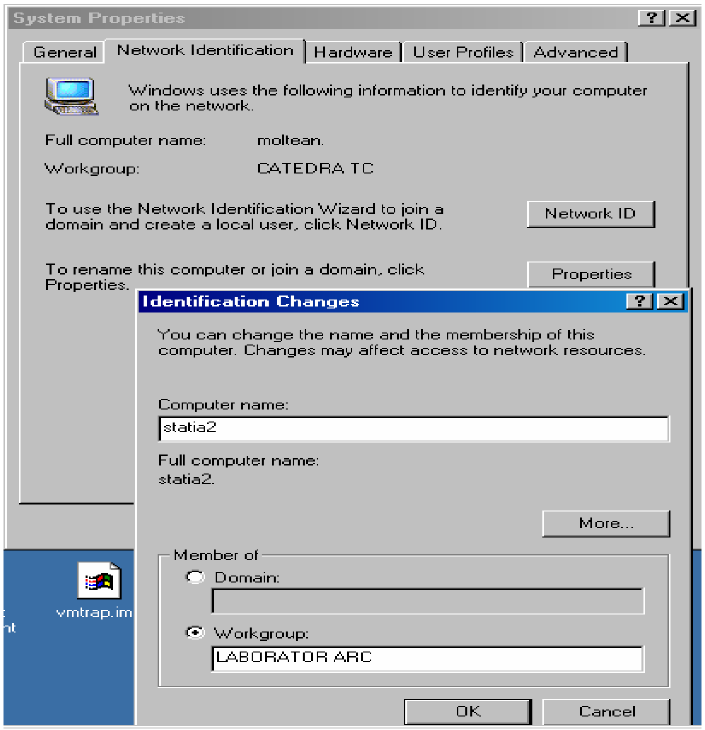
IP address: 192.168.7.X (where the last byte X will be different from station to station).
Warning: the first available address in this range (192.168.7.1) cannot be allocated, because this address correspond to our lab's gateway (see below).
Netmask: 255.255.255.0
Default Gateway: 192.168.7.1
DNS: 81.181.101.2

2. Goal: testing the inter-connectivity in the lab's LAN and the external connectivity with Internet

Once the required settings introduced, you can check their form by using *ipconfig/all* from the command line (DOS mode). In order to check if the gateway router is reachable, you can use *ping 192.168.7.1* and wait for an answer. The same command is used to check the external connectivity (you may, for example, ping www.yahoo.com).

3. Goal: Setting our LAN's working domain

In order for the computers to communicate with each other, they must be configured appropriately. Hence, the working group must be configured with the same name. In order to set this group, you must right click on My Computer, choose Properties – Network Identification – Properties. This will open the following window:



System Properties

Windows uses the following information to identify your computer on the network.

Full computer name: moltean.

Workgroup: CATEDRA TC

To use the Network Identification Wizard to join a domain and create a local user, click Network ID.

To rename this computer or join a domain, click Properties.

Identification Changes

You can change the name and the membership of this computer. Changes may affect access to network resources.

Computer name: statia2

Full computer name: statia2.

Member of

○ Domain:

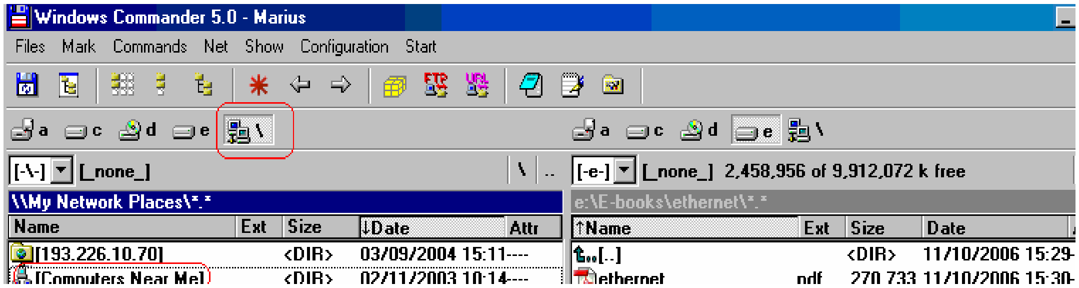
● Workgroup: LABORATOR ARC

Figure 7: Workgroup configuration

In the Workgroup tab, you must introduce ARC LAB, as the figure above shows. The computer's name must be set to StationX, where X is the decimal value of the last byte from the IP addressed, as chosen in the first step of your work.

4. Goal : seeking for your neighbors

In order to check the correctitude of your previous work, you can try to look for the other computers from your LAN. This can be done by double-clicking on Network Neighborhood (eventually using an auxiliary tool such as Total Commander), and then Computers Near Me, as indicated in the figure below.


Windows Commander 5.0 - Marius
Files Mark Commands Net Show Configuration Start
a c d e \
a c d e \

[-\-] [_none_]

\\My Network Places*.*

Name	Ext	Size	↓Date	Attr
[193.226.10.70]		<DIR>	03/09/2004 15:11	----
[Computers Near Me]		<DIR>	02/11/2003 10:14	----

[-e-] [_none_] 2,458,956 of 9,912,072 k free

e:\E-books\ethernet\*.*

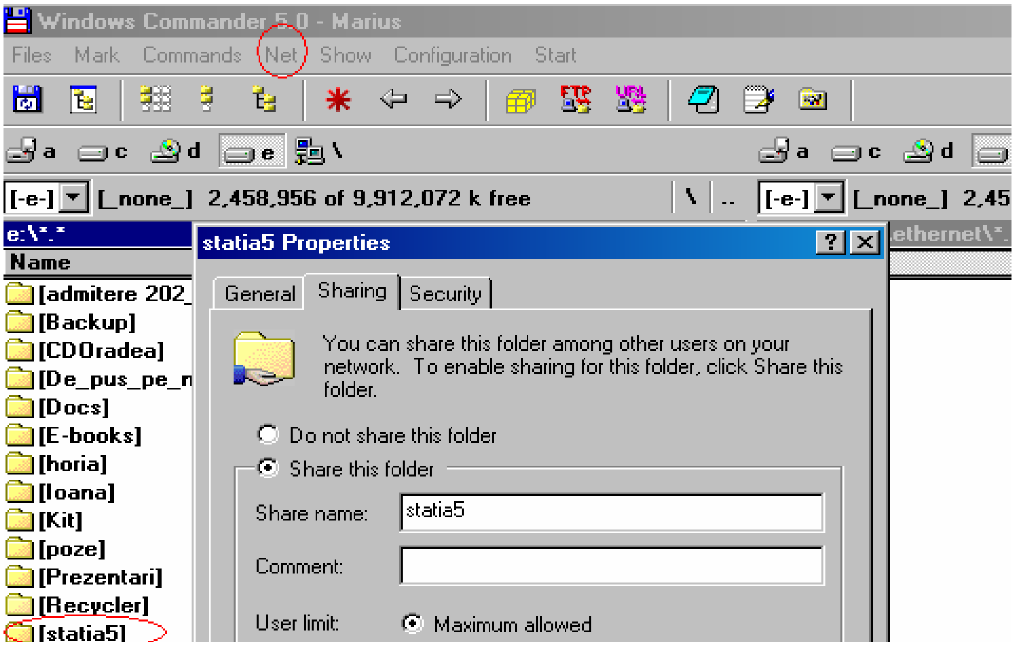
↑Name	Ext	Size	Date
[..]		<DIR>	11/10/2006 15:29
ethernet	pdf	270,733	11/10/2006 15:30

Figure 8: Looking for your neighbors

The effect that you are looking for is to be able to see al the other workstation listed in the right tab from Windows Commander.

5. Goal: Sharing your resources

On every workstation you shall create on the D drive a folder having the same name as your computer (e.g: the students from Station5 will create the folder Station5). This folder can be shared either using Windows Commander or using Windows Explorer. In the first case you must follow the steps: Share Current Directory – Sharing – Share this Folder (see figure 9). When all the students will complete this operation, they should be able to access the shared file from any other computer, bi right clicking on the desired computer name from Computers Near me.


Windows Commander 5.0 - Marius
Files Mark Commands Net Show Configuration Start
a c d e \
a c d
? ✕
General Sharing Security

[-e-] [_none_] 2,458,956 of 9,912,072 k free

[-e-] [_none_] 2,45

e:\*.*

Name

[admitere 202_
[Backup]
[CDOradea]
[De_pus_pe_n
[Docs]
[E-books]
[horia]
[Ioana]
[Kit]
[poze]
[Prezentari]
[Recycler]
[statia5]

ethernet*.

statia5 Properties

You can share this folder among other users on your network. To enable sharing for this folder, click Share this folder.

○ Do not share this folder

● Share this folder

Share name: statia5

Comment:

User limit: ● Maximum allowed

Figure 9: sharing your work

Lab IV: MAC and IP Addresses

1. Goals

- To understand the concepts of IP and MAC address
- To understand the differences between different classes of IP addresses
- To notice the difference between the fixed and dynamic IP address allocation
- To understand the difference between private and public IP addresses
- To introduce the sub-networking concepts

2. Introduction

The Internet can be defined as a virtual network which relies on the interconnection of physical systems using some dedicated devices, usually called gateways. The protocol stack that governs the Internet is TCP/IP. This work will be dedicated to addressing issues, because the addresses are the main mean which helps the physical network details to be transparent to the TCP/IP software. This makes this huge network which is Internet to work as a single, uniform entity.

3. IP addressing

As with any other network-layer protocol, the IP addressing scheme is integral to the process of routing IP datagrams through an internetwork. Each IP address has specific components and follows a basic format. These IP addresses can be subdivided and used to create addresses for subnetworks, as discussed in more detail later in this laboratory work.

Each host on a TCP/IP network is assigned a unique 32-bit logical address that is divided into two main parts: the network number and the host number. The network number identifies a network and must be assigned by the Internet Network Information Center (InterNIC) if the network is to be part of the Internet. An Internet Service Provider (ISP) can obtain blocks of network addresses from the InterNIC and can itself assign address space as necessary. The host number identifies a host on a network and is assigned by the local network administrator.

The 32-bit IP address is grouped eight bits at a time, separated by dots, and represented in decimal format (known as *dotted decimal notation*). Each bit in the octet has a binary weight (128, 64, 32,16, 8, 4, 2, 1). The minimum value for an octet is 0, and the maximum value for an octet is 255. Figure 1 illustrates the basic format of an IP address.

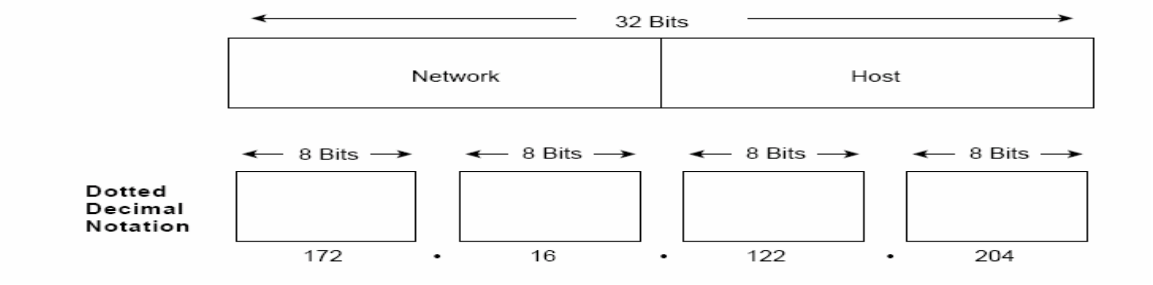


Figure 1: Basic format of IP addresses

This format is characteristic to IPv4, which is currently the most widely-used IP protocol version.

3.1 IP address classes

IP addressing supports five different address classes: A, B,C, D, and E. Only classes A, B, and C are available for commercial use. The way that these classes can be identified is illustrated in figure 2.

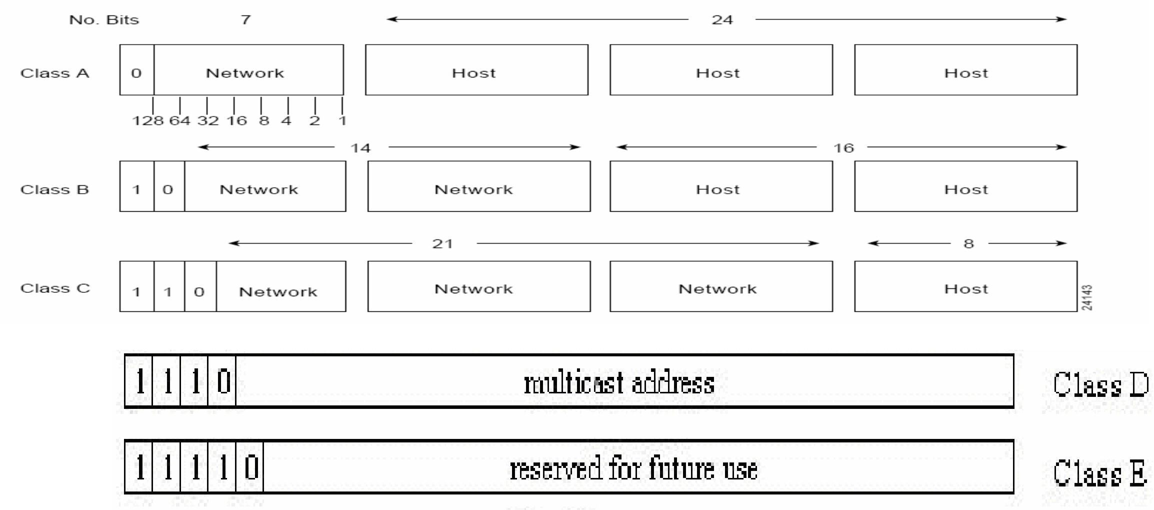
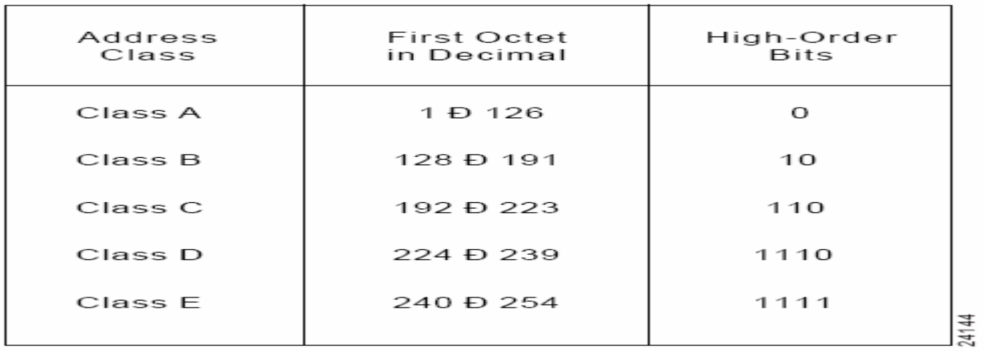


Figure 2: The five address classes

The class of address can be determined easily by examining the first octet of the address and mapping that value to a class range in the following table. In an IP address of 172.31.1.2 (with reference to figure 1), for example, the first octet is 172. Because 172 falls between 128 and 191, 172.31.1.2 is a Class B address. Table 1 summarizes the range of possible values for the first octet of each address class.



Address Class	First Octet in Decimal	High-Order Bits
Class A	1 Ð 126	0
Class B	128 Ð 191	10
Class C	192 Ð 223	110
Class D	224 Ð 239	1110
Class E	240 Ð 254	1111

Table 1: The range of every IP address class

3.2 Special use IP addresses

Several address ranges are reserved for "Special Use". These addresses all have restrictions of some sort placed on their use, and in general should not appear in normal use on the public Internet. The following briefly documents these addresses – in general they are used in specialized technical contexts. They are described in more detail in RFC 3330.

Private Use IP addresses:

- 10.0.0.0 - 10.255.255.255
- 172.16.0.0 - 172.31.255.255
- 192.168.0.0 - 192.168.255.255

The above address blocks are reserved for use on private networks, and should never appear in the public Internet. There are hundreds of thousands of such private networks (for example home firewalls sometimes make use of them). The IANA has no record of who uses these address blocks. Anyone may use these address blocks within their own network without any prior notification to IANA.

The point of private address space is to allow many organizations in different places to use the same addresses, and as long as these disconnected or self-contained islands of IP-speaking computers (private intranets) are not connected, there is no problem. If you see an apparent attack, or spam, coming from one of these address ranges, then either it is coming from your local environment, or the address has been "spoofed".

Autoconfiguration IP Addresses:

- 169.254.0.0 - 169.254.255.255

Addresses in the range 169.254.0.0 to 169.254.255.255 are used automatically by some PCs and Macs when they are configured to use IP, do not have a static IP Address assigned, and are unable to obtain an IP address using DHCP.

This traffic is intended to be confined to the local network, so the administrator of the local network should look for misconfigured hosts. Some ISPs inadvertently also permit this traffic, so you may also want to contact your ISP. This is documented in RFC 3330.

"Loopback" IP addresses:

- 127.0.0.0 - 127.255.255.255

Each computer on the Internet uses 127.0.0.0/8 to identify itself, to itself. 127.0.0.0 to 127.255.255.255 is earmarked for what is called "loopback". This construct allows a computer to establish/validate its IP stack. Most software only uses 127.0.0.1 for loopback purposes (the other addresses in this range are seldom used). All of the addresses within the loopback address are treated with the same levels of restriction in Internet routing, so it is difficult to use any other addresses within this block for anything other than node specific applications, generally bootstraping. This is documented in RFC 3330.

3.3 The link towards MAC addresses

ARP - Address Resolution Protocol

Note that it's not enough to have a destination IP address in order to send and receive data; the transmitting device must have a destination MAC address as well. If the sender doesn't know the MAC address of the destination, it has to get that address before data can be sent. To obtain the unknown Layer Two (MAC) address when the Layer Three (IP) address is known, the sender transmits an ARP Request. This is a Layer Two broadcast, which has a destination address of ff-ff-ff-ff-ff-ff. Since Ethernet is a broadcast media, every other device on the segment will see it. However, the only device that will answer it is the device with the matching Layer Three address. That device will send an ARP Reply, unicast back to the device that sent the original ARP Request. The sender will then have a MAC address to go with the IP address and can then transmit. Repeaters and Hubs are Layer One (Physical Layer) devices, and they have no impact on ARP. A repeater's job is simply to regenerate a signal to make it stronger, and a hub is simply a multiport repeater. Therefore, neither a repeater, nor a hub have impact on ARP. Switches are Layer Two devices, so you might think they impact ARP's operation; after all, ARP deals with getting an unknown MAC address to correspond with a known IP address. While that's certainly true, switches don't impact ARP for one simple reason: switches forward broadcasts out every port except the one it was originally received on. The ARP Reply will be unicast to the device requesting it, as with the previous example. Now here's the exception: a router. Routers accept broadcasts, but routers will not forward them. For example, consider a PC with the address 20.1.1.1 /16. That host assumes it's on the same physical segment as the device 20.1.2.200 /16, since their IP addresses are both on the same subnet (20.1.0.0 /16). The problem here is that a router separates the two devices, and the router will not forward the ARP broadcast. A router will answer the ARP Request, however, with the MAC address of the router interface the ARP Request was received on. In this case, the router will respond to the ARP Request with its own E1 interface's MAC address.

When the device at 20.1.1.1 receives this ARP Response, it thinks the MAC address of 20.1.2.200 is 11-11-11-11-11-11. Therefore, the destination IP for traffic destined for the remote host will be 20.1.2.200, but the MAC destination will actually be that of the router's E1 interface.

RARP: Reversed ARP

RARP obtains a device's IP address when it already knows its own MAC address. A separate device, a RARP Server, tells the device what its MAC address is in response to the RARP Request.

3.4 Basics of subnetting

IP networks can be divided into smaller networks called subnetworks (or subnets). Subnetting provides the network administrator with several benefits, including extra flexibility, more efficient use of network addresses, and the capability to contain broadcast traffic (a broadcast will not cross a router).

Subnets are under local administration. As such, the outside world sees an organization as a single network and has no detailed knowledge of the organization’s internal structure.

A given standard (according to IP classes) network address can be broken up into many subnetworks. A subnet address is created by “borrowing” bits from the host field and designating them as the subnet field. The number of borrowed bits varies and is specified by the subnet mask. Figure 3 shows how bits are borrowed from the host address field to create the subnet address field, for a B class IP address.

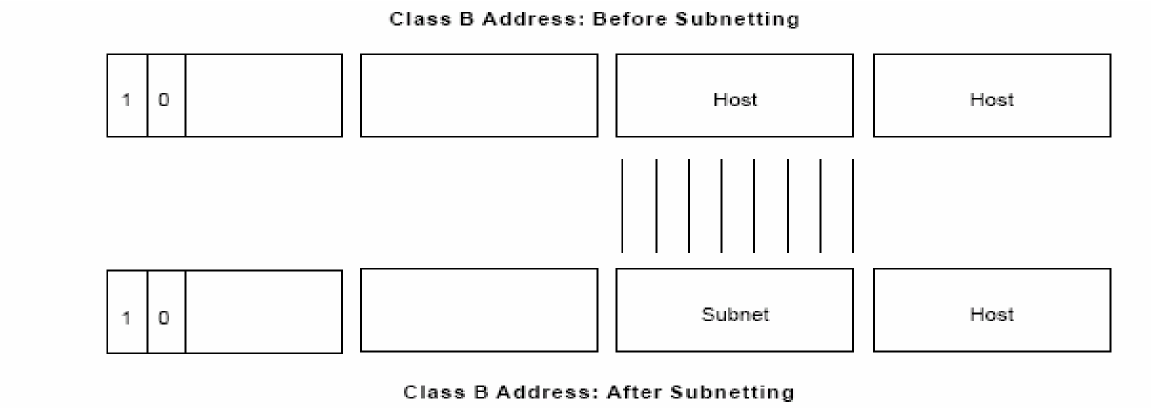

Class B Address: Before Subnetting
1
0
Host
Host
1
0
Subnet
Host
Class B Address: After Subnetting

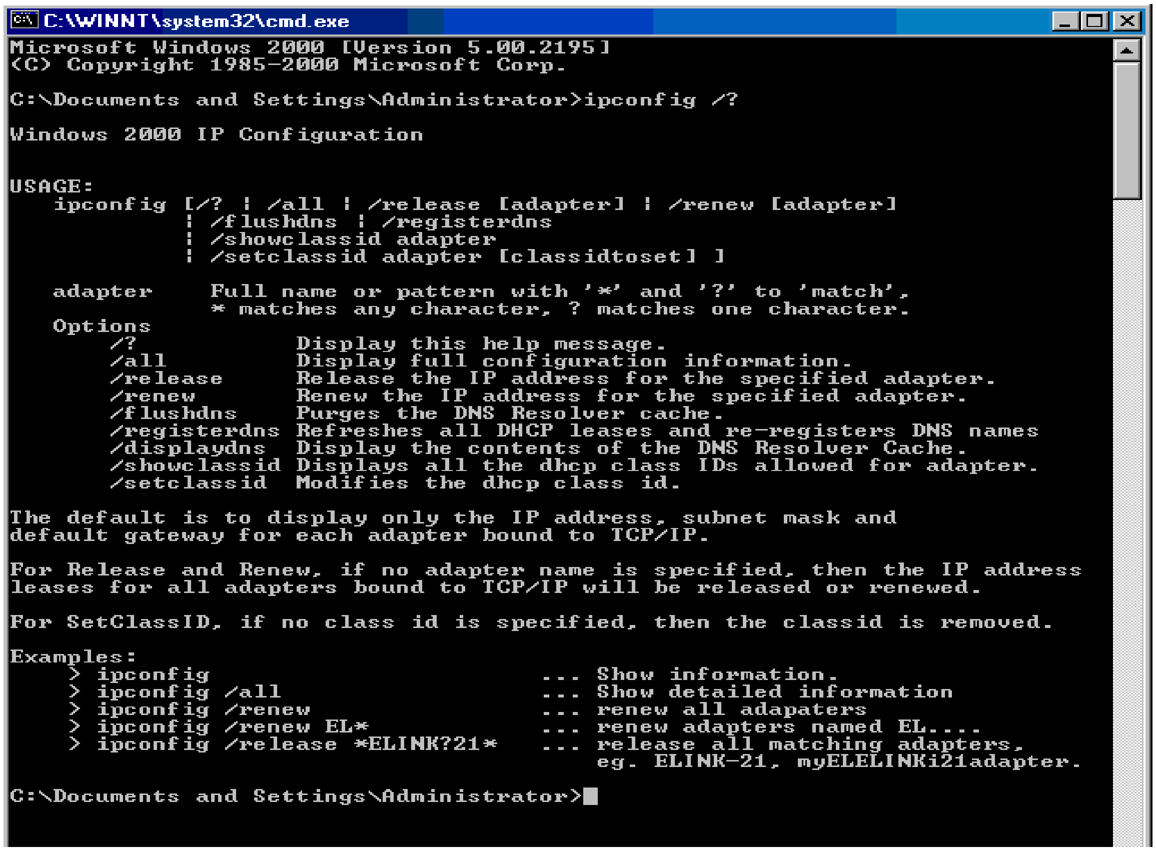
Figure 3: Subnetting principle

For example in the case of B class IP address 172.16.1.0, 172.16.2.0, 172.16.3.0, and 172.16.4.0 are all subnets within network 171.16.0.0. With respect to subnetting principles, it is important to understand the principles of a subnet mask. Such a mask use the same format and representation technique as IP addresses. The subnet mask, however, has binary 1s in all bits specifying the network and subnetwork fields, and binary 0s in all bits specifying the host field.

4. Practical work

1. Goal**:** to check the IP and MAC configurations for the PCs in our laboratory

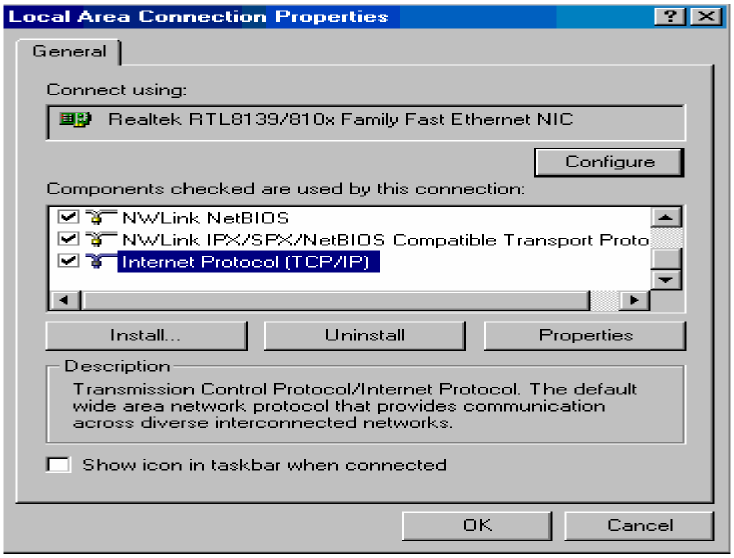
In order to check the network configuration you can use the command *ipconfig*. For this purpose, you must select the option *Run* from the *Start* menu, and then *Cmd*. In the DOS window opened, you must check what are the parameters of ipconfig command, using the syntax *ipconfig ?/* . This operation is illustrated in figure 4.
In order to check your network settings, you must launch the command *ipconfig /all*. The students shall check what are the IP and MAC address of their PCs, and they should observe what are the differences between the two addresses. You must read carefully all information displayed and try to understand their meaning.



```
Microsoft Windows 2000 [Version 5.00.2195]
(C) Copyright 1985-2000 Microsoft Corp.

C:\Documents and Settings\Administrator>ipconfig /?

Windows 2000 IP Configuration


USAGE:
    ipconfig [/? | /all | /release [adapter] | /renew [adapter]
                | /flushdns | /registerdns
                | /showclassid adapter
                | /setclassid adapter [classidtoset] ]

    adapter     Full name or pattern with '*' and '?' to 'match',
                * matches any character, ? matches one character.
    Options
        /?           Display this help message.
        /all         Display full configuration information.
        /release     Release the IP address for the specified adapter.
        /renew       Renew the IP address for the specified adapter.
        /flushdns    Purges the DNS Resolver cache.
        /registerdns Refreshes all DHCP leases and re-registers DNS names
        /displaydns  Display the contents of the DNS Resolver Cache.
        /showclassid Displays all the dhcp class IDs allowed for adapter.
        /setclassid  Modifies the dhcp class id.

The default is to display only the IP address, subnet mask and
default gateway for each adapter bound to TCP/IP.

For Release and Renew, if no adapter name is specified, then the IP address
leases for all adapters bound to TCP/IP will be released or renewed.

For SetClassID, if no class id is specified, then the classid is removed.

Examples:
     > ipconfig                         ... Show information.
     > ipconfig /all                    ... Show detailed information
     > ipconfig /renew                  ... renew all adapaters
     > ipconfig /renew EL*              ... renew adapters named EL....
     > ipconfig /release *ELINK?21*     ... release all matching adapters,
                                            eg. ELINK-21, myELELINKi21adapter.

C:\Documents and Settings\Administrator>
```

Figure 4: Checking IPCONFIG options

A second option which allows you to check and, furthermore, to configure your network settings using Win2000 operating system is the following: right click on *My Network Places*, choose *Properties-Local Area Network*, right click again and *Properties*. These steps will open the window shown in figure 5.



Connect using:

Realtek RTL8139/810x Family Fast Ethernet NIC

Components checked are used by this connection:

☑ NWLink NetBIOS
☑ NWLink IPX/SPX/NetBIOS Compatible Transport Proto
☑ Internet Protocol (TCP/IP)

Description

Transmission Control Protocol/Internet Protocol. The default wide area network protocol that provides communication across diverse interconnected networks.

□ Show icon in taskbar when connected

Figure 5: Setting network properties

By double clicking on the TCP/IP option (highlighted in fig. 5), a new window will open, allowing you to view and to modify the network related information: the IP address, the netmask, the DNS server, whether or not a DHCP is used in the network. Finally, you must check the option advanced, carefully observing what are the information and the configuration options available.

2. Goal: to understand the basics of IP subnetting

The students will have to spilt the local network from the ARC laboratory into three subnetworks. Note that the network address of this LAN is 192.168.7.0. The second subnetwork must have the address 192.168.7.16. The second one will have the last byte .32 and the last one .64. At the completion of this exercise, the students must be able to answer the following questions:
How many computers can be addressed this way in every subnetwork?
What will be the broadcast addresses for these computers?