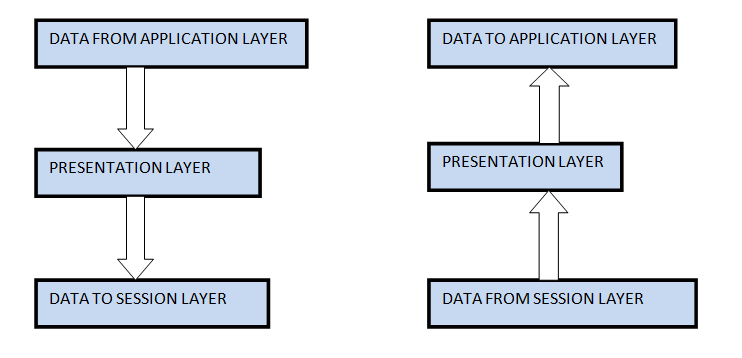
**UNIT-IV**

Presentation layer Abstract syntax notation, Data compression techniques, Cryptography Application such as file transfer, Electronic mail and virtual terminals, X 400 protocol for electrical messaging overview of ARPANET, MAP, TOP, Novell Netware, PC/NOS, Unix support for networking.

### 1. Presentation layer Abstract syntax notation

The primary goal of this layer is to take care of the syntax and semantics of the information exchanged between two communicating systems. Presentation layer takes care that the data is sent in such a way that the receiver will understand the information (data) and will be able to use the data. Languages (syntax) can be different of the two communicating systems. Under this condition presentation layer plays a role translator.



#### FUNCTIONS OF PRESENTATION LAYER:

1. **Translation:**Before being transmitted, information in the form of characters and numbers should be changed to bit streams. The presentation layer is responsible for interoperability between encoding methods as different computers use different encoding methods. It translates data between the formats the network requires and the format the computer.
2. **Encryption:**It carries out encryption at the transmitter and decryption at the receiver.
3. **Compression:**It carries out data compression to reduce the bandwidth of the data to be transmitted. The primary role of Data compression is to reduce the number of bits to be transmitted. It is important in transmitting multimedia such as audio, video, text etc.

**2. Data compression techniques**

The process of reducing the volume of data by applying a compression technique is called compression. The resulting data is called compressed data. The reverse process of reproducing the original data from compressed data is called decompression. The resulting data is called decompressed data.

**Reasons to Compress**

* Reduce file size
* Save disk space
* Increase transfer speed at a given data rate
* Allow real-time transfer at a given data rate.

**Types of Compression Techniques:**

Compression techniques can be categorized based on following consideration:

* Lossless or loss
* Symmetrical or asymmetrical
* Software or hardware

**1. Lossless or loss**

If the decompressed data is the same as the original data, it is referred to as lossless compression, otherwise the compression is loss.

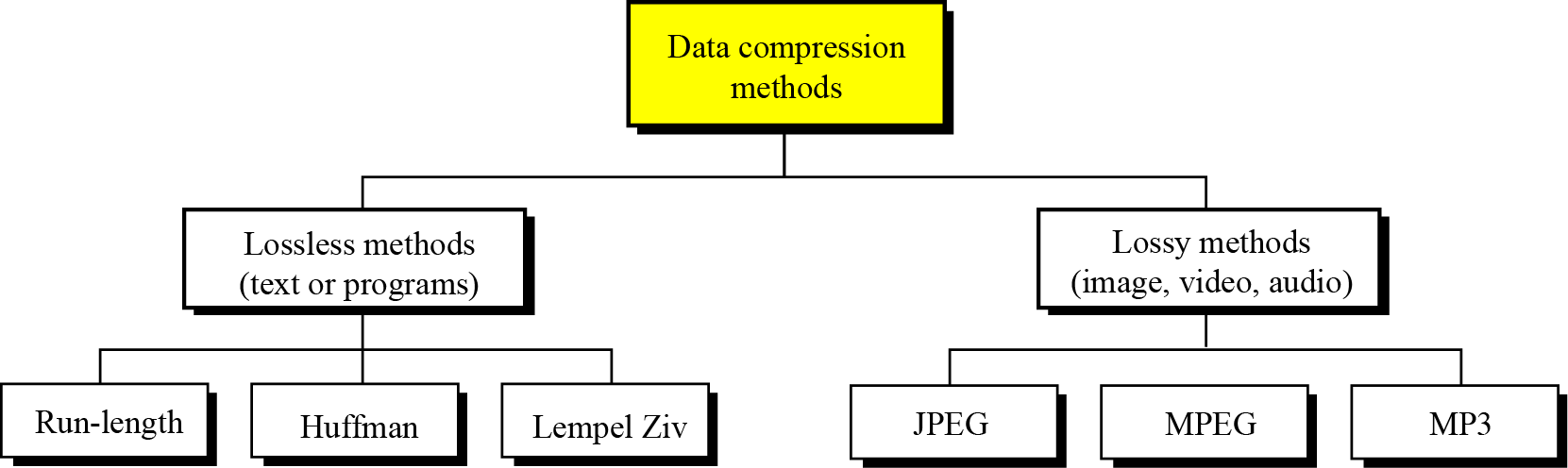
**2. Symmetrical or asymmetrical**

In symmetrical compression, the time required to compress and to decompress are roughly the same.

In asymmetrical compression, the time taken for compression is usually much longer than decompression.

**3. Software or hardware**

A compression technique may be implemented either in hardware or software. As compared to software codes (coder or decoder), hardware codes offer better quality and performance.

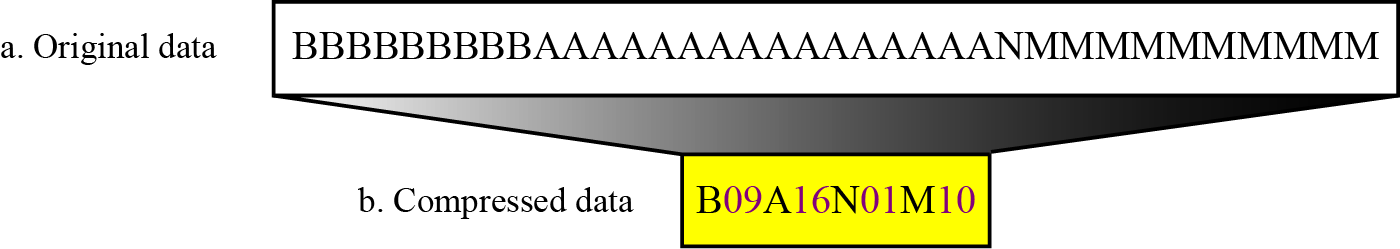


**1. Lossless methods**

* The original data and the data after compression and decompression are exactly the same because, in these methods, the compression and decompression algorithms are exact inverses of each other: no part of the data is lost in the process.
* Redundant data is removed in compression and added during decompression.

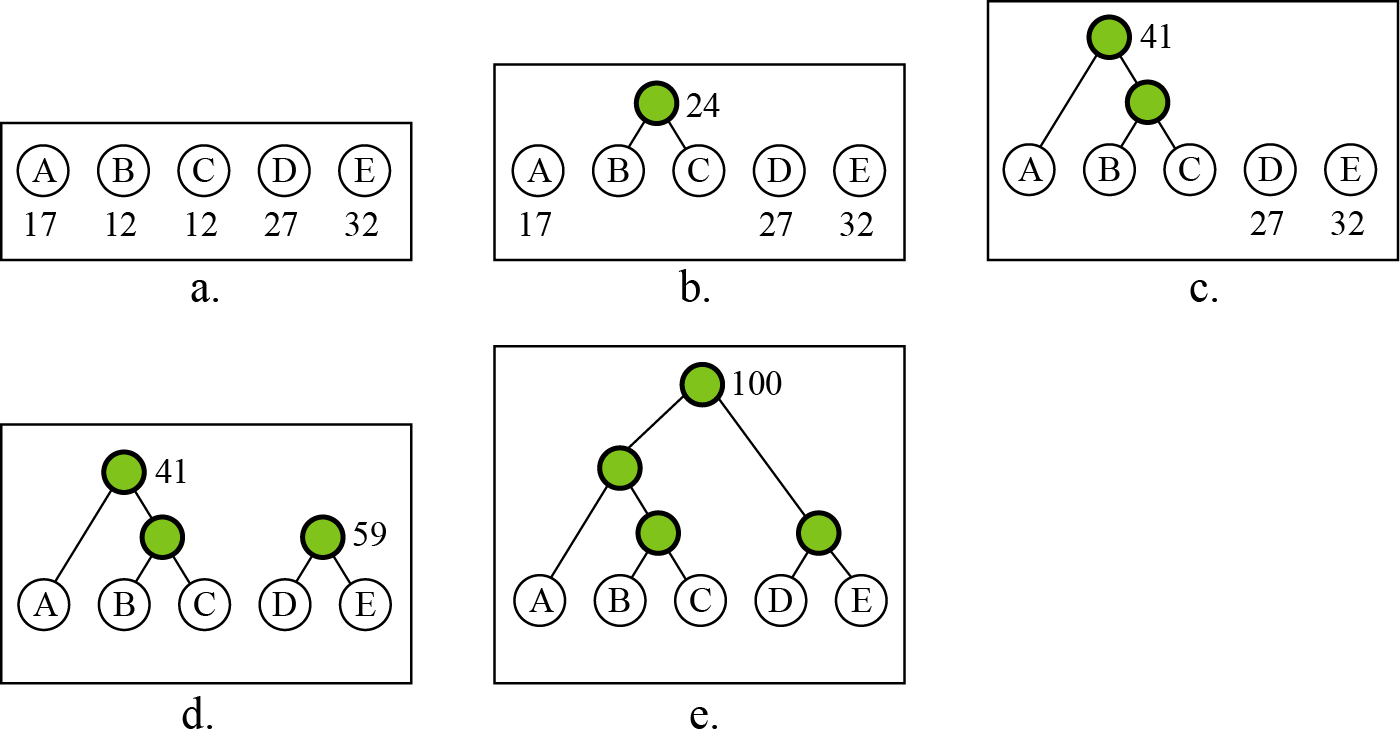
**Run-length encoding**

Run-length encoding is probably the simplest method of compression.It can be used to compress data made of any combination of symbols.It does not need to know the frequency of occurrence of symbols and can be very efficient if data is represented as 0s and 1s.

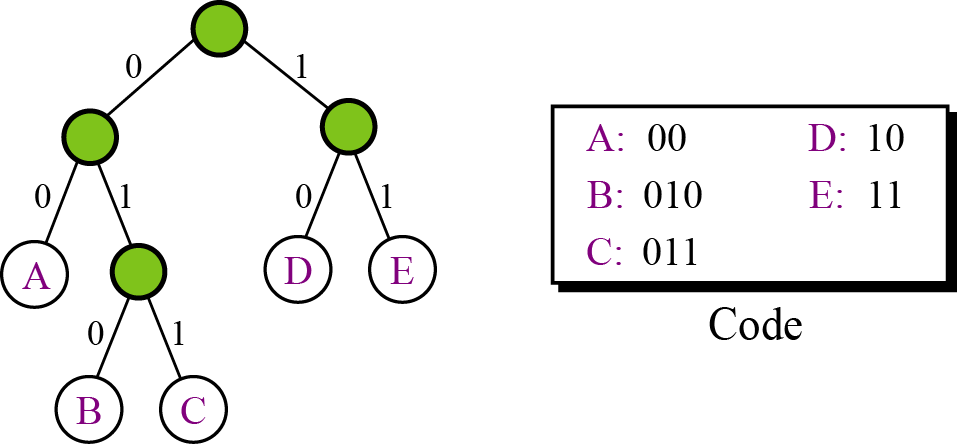
****

**Huffman coding**

Huffman coding assigns shorter codes to symbols that occur more frequently and longer codes to those that occur less frequently.For example, imagine we have a text file that uses only five characters (A, B, C, D, E). Before we can assign bit patterns to each character, we assign each character a weight based on its frequency of use.

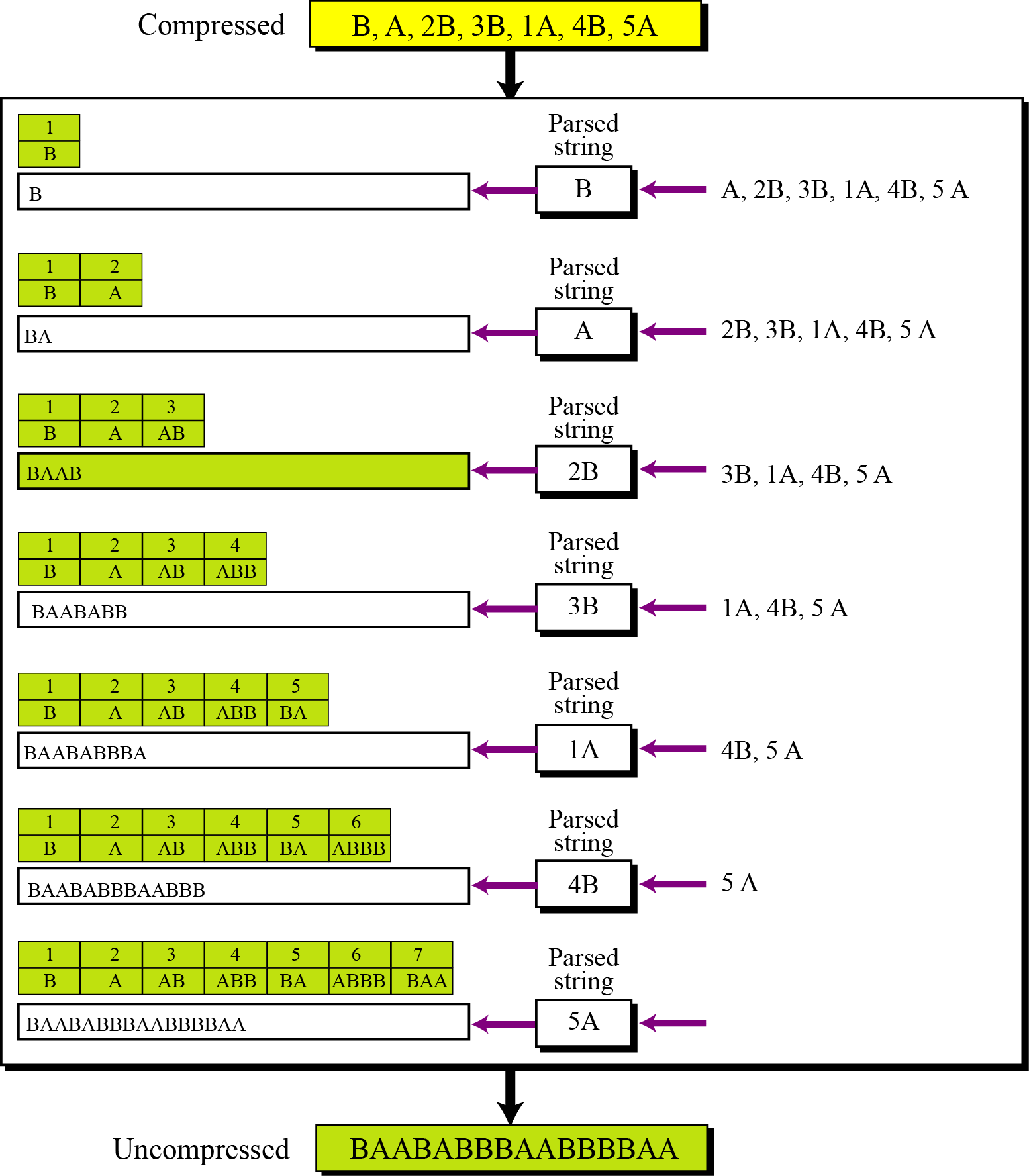


A character’s code is found by starting at the root and following the branches that lead to that character. The code itself is the bit value of each branch on the path, taken in sequence.



**Lempel Ziv encoding**

Lempel Ziv (LZ) encoding is an example of a category of algorithms called dictionary-based encoding.The idea is to create a dictionary (a table) of strings used during the communication session.If both the sender and the receiver have a copy of the dictionary, then previously-encountered strings can be substituted by their index in the dictionary to reduce the amount of information transmitted.

****

**2. Lossy methods**

Our eyes and ears cannot distinguish subtle changes. In such cases, we can use a lossy data compression method. These methods are cheaper—they take less time and space when it comes to sending millions of bits per second for images and video.

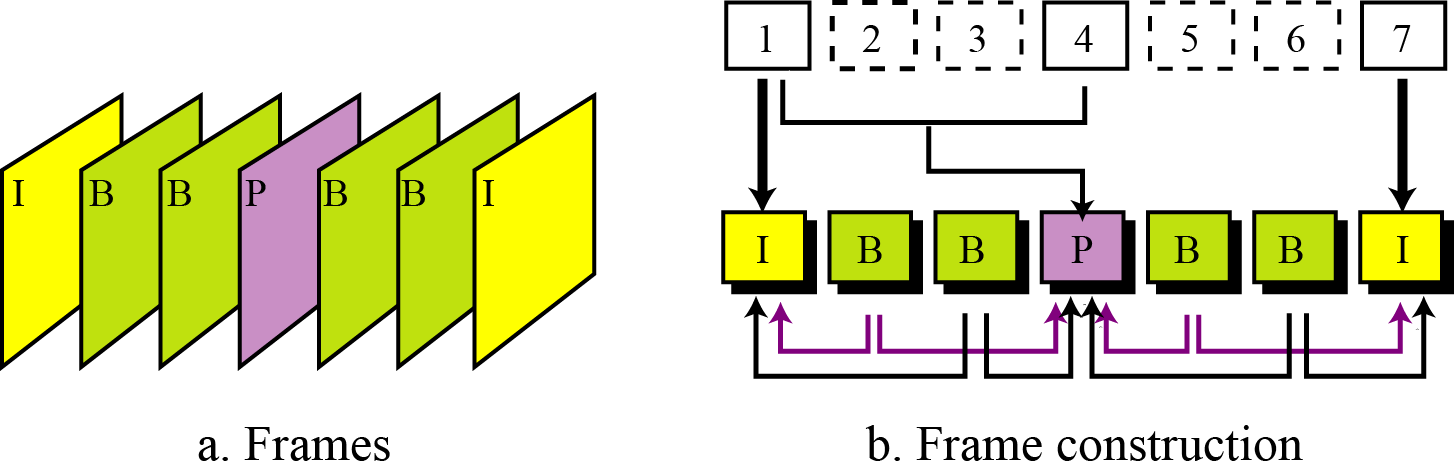
Several methods have been developed using lossy compression techniques. JPEG (Joint Photographic Experts Group) encoding is used to compress pictures and graphics, MPEG (Moving Picture Experts Group) encoding is used to compress video, and MP3 (MPEG audio layer 3) for audio compression.

**Image compression – JPEG encoding**

An image can be represented by a two-dimensional array (table) of picture elements (pixels). A grayscale picture of 307,200 pixels is represented by 2,457,600 bits, and a color picture is represented by 7,372,800 bits.In JPEG, a grayscale picture is divided into blocks of 8 × 8 pixel blocks to decrease the number of calculations because, as we will see shortly, the number of mathematical operations for each picture is the square of the number of units

**Video compression – MPEG encoding**

The Moving Picture Experts Group (MPEG) method is used to compress video. In principle, a motion picture is a rapid sequence of a set of frames in which each frame is a picture. In other words, a frame is a spatial combination of pixels, and a video is a temporal combination of frames that are sent one after another. Compressing video, then, means spatially compressing each frame and temporally compressing a set of frames.



**MPEG frames**

**Audio compression**

Audio compression can be used for speech or music. For speech we need to compress a 64 kHz digitized signal, while for music we need to compress a 1.411 MHz signal.Two categories of techniques are used for audio compression: predictive encoding and perceptual encoding.

**3.Cryptography**

* A word with Greekorigins, means “secret writing”.
* The term to refer to the science and art of transforming messages to make them secure and immune to attacks.
* Applications of cryptography includeATM cards, computer passwords, and electronic commerce.
* Cryptography is the science and art of transforming messages to make them secure and immune to attack.

**Cryptography Issues**

* **Confidentiality:**Only sender, intended receiver should “understand” message contents.
* **End-Point Authentication:** Sender and receiver want to confirm identity of each other.
* **Message Integrity:** Sender and receiver want to ensure message not altered (in transit, or afterwards) without detection.
* **Message Nonrepudiation:** Message nonrepudiation means that a sender must not be able to deny sending a message that he or she, in fact, did send.
* **Entity Authentication:** In entity authentication (or user identification) the entity or user is verified prior to access to the system resources

**Plaintext and Ciphertext**

* The original message, before being transformed, is called plaintext.
* After the message is transformed, it is called ciphertext.
* An encryption algorithm transforms the plaintext into ciphertext; a decryption algorithm transforms the ciphertext back into plaintext.
* Example:
  + Plaintext: HELLO
  + Ciphertext: KHOOR

**Cipher**

* Encryption and Decryption algorithms are referred as ciphers.
* Also used to refer to different categories of algorithms in cryptography.
* Example (Traditional Substitution Ciphers):
  + Monoalphabetic cipher
  + Polyalphabetic cipher
    - Plaintext: HELLO
    - Ciphertext: ABNZF

**Key**

* A key is a number (or a set of numbers) that the cipher operates on, as an algorithm.
* To encrypt a message, we need an encryption algorithm, an encryption key, and the plaintext.
* To decrypt a message, we need a decryption algorithm, a decryption key, and the ciphertext. These reveal the original plaintext.
* Types:
  + Shared key, Public key and Private key.

**What are the Types of Cryptography**

**1.Symmetric Key Cryptography (Secret Key Cryptography)**

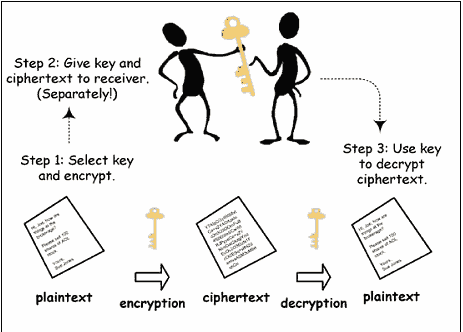
* + Same Key is used by both parties

**Advantages**

* 1. Simpler and Faster

**Disadvantages**

* 1. Less Secured



**2. Asymmetric Key Cryptography (Public Key Cryptography)**

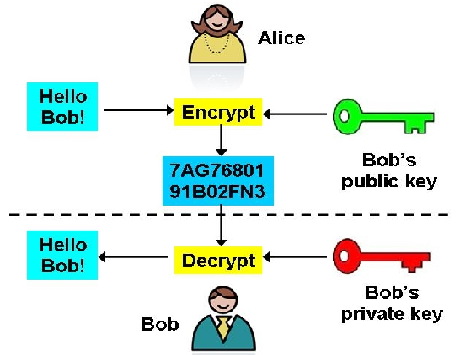
* + **2 different keys are used**
  + **Users get the Key from an Certificate Authority**

**Advantages**

* 1. **More Secured**
  2. **Authentication**

**Disadvantages**

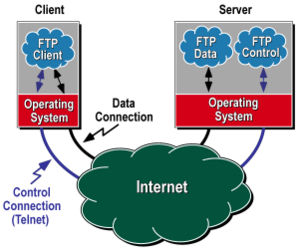
* 1. **Relatively Complex**



|  |  |
| --- | --- |
| **SYMMETRIC KEY CRYPTOGRAPHY** | **ASYMMETRIC KEY CRYPTOGRAPHY** |
| 1)The same algorithm with the same key is used for encryption and decryption.  2) The key must be kept secret.    3) It may be impossible or at least impractical to decipher a message if no other information is available. | 1) One algorithm is used for encryption and decryption with a pair of keys, one for encryption and one for decryption.    2) One of the two keys must be kept secret.    3) It may be impossible or at least impractical to decipher a message if no other information is available. |

**Applications**

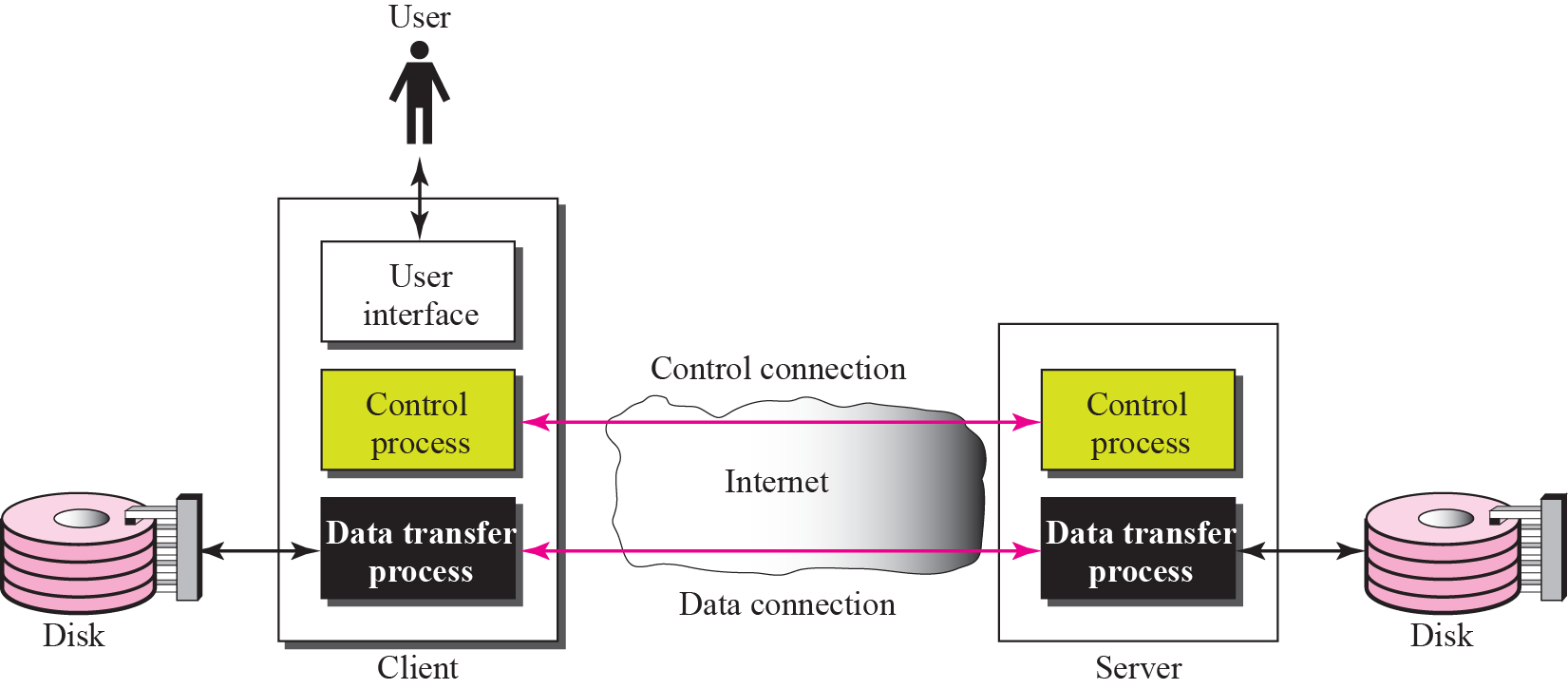
**1.FTP(File Transfer Protocol)**



File Transfer Protocol (FTP) is a standard network protocol used to copy a file from one host to another over a TCP/IP-based network, such as the Internet.FTP is built on client-server architecture. Simplest and most secure way to exchange files over the Internet or intranet.

Transferring files from a client computer to a server computer is called **"uploading"** and transferring from a server to a client is **"downloading".**To access an FTP server, users must be able to connect to the Internet or an intranet (via a modem or local area network) with an FTP client program.FTP uses the services of TCP. It needs two TCP connections. The well-known **port 21** is used for the **control connection** and the well-known **port 20 for the data connection.**

**Communication Structure of FTP**

****

**Control Connection**

The communication path between the USER-PI (Process Interpreter) and SERVER-PI for the exchange of commands and replies. This connection follows the Telnet Protocol.

**Data Connection**

A full duplex connection over which data is transferred, in a specified mode and type. The data transferred may be a part of a file, an entire file or a number of files. The path may be between a server-DTP (Data Transfer Process) and a user-DTP, or between two server-DTPs.

**Types of Connections**

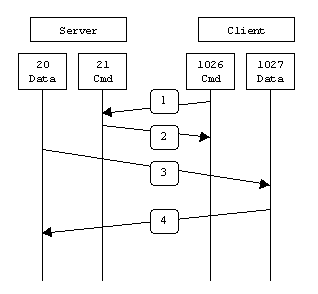
**Active FTP**

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.

From the server-side firewall's standpoint, to support active mode FTP the following communication channels need to be opened:

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

When drawn out, the connection appears as follows:



**Passive FTP**

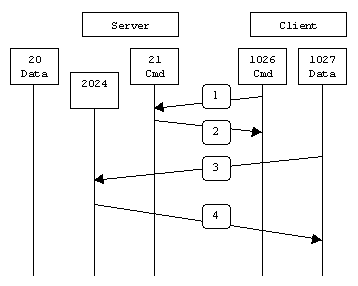
In order to resolve the issue of the server initiating the connection to the client a different method for FTP connections was developed. This was known as passive mode, or PASV, after the command used by the client to tell the server it is in passive mode.

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 P back to the client in response to the PASV command. The client then initiates the connection from port N+1 to port P on the server to transfer data.

From the server-side firewall's standpoint, to support passive mode FTP the following communication channels need to be opened:

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

When drawn, a passive mode FTP connection looks like this:



**Port numbers**

21: File Transfer Protocol (FTP)

22: Secure Shell (SSH)

23: Telnet remote login service.(Telnet)

25: Simple Mail Transfer Protocol (SMTP)

53: Domain Name System (DNS) service.

80: Hypertext Transfer Protocol (HTTP) used in the World Wide Web.

110: Post Office Protocol (POP3)

443: HTTP with Secure Sockets Layer (SSL)

161,162: Simple Network Management Protocol (SNMP)

**FTP Commands**

### 1. Connect to a FTP site

Connect to a particular FTP server using **ftp** command as shown below.  
Syntax:

$ ftp IP/hostname

Or

$ ftp

ftp> open IP/hostname

You can directly open connection with a remote host using it’s IP or host name from the command line. You can also go to ftp prompt and use open command to connect with remote host.

It will ask you for the user name and password to login. On some public domain FTP server, you can use “anonymous” username with any email address as the password to connect.

### 2. Download a file using ftp

Use the get command to download file from a remote ftp server as shown below.

ftp> get Filename

You have to be in the right mode to download files. i.e binary or ascii mode. Use ascii mode for transferring text files, and binary mode for all other type of files.Download the file and save it with another name. In the following example, index.html file will be downloaded and saved as my.html on the local server.

ftp> get index.html my.html

### 3. Changing FTP Mode to binary or ascii

**Go to ftp Ascii mode**

ftp>ascii

**Go to ftp Binary mode**

ftp> binary

### 4. Uploading a file to FTP server

Use put command to upload a file to a remote ftp server as shown below.

ftp> put filename

### 5. Changing the remote and local directory

Apart from downloading or uploading a file, you may want to change either the remote or local directory, which you can do using **cd** and **lcd** respectively.

**Change the remote server current directory using cd command**

ftp>pwd

257 "/myftpserver" is current directory.

ftp> cd dir1

250 CWD command successful. "/myftpserver/dir1" is current directory.

ftp>pwd

257 "/myftpserver/dir1" is current directory.

**Change the local machine current directory using lcd command**

ftp> !

$ pwd

/home/sathiya/FTP

$ exit

exit

ftp>lcd /tmp

Local directory now /tmp

ftp> !

$ pwd

/tmp

Note:

* executing **!** takes you to the shell.
* prompt starts with **ftp>** is ftp prompt.
* prompt starts with **$**is shell command line.

### 6. Listing the contents of remote directory from FTP

You can view the content of a remote directory using the **ls** / **dir** command.

ftp>ls

### 7. FTP Help

Type **help** or **?** to view list of all available ftp commands.

For a detailed help on a particular ftp command use:

ftp> help COMMAND

### 8. Downloading multiple files with mget command

mget is for fetching multiple files from ftp server. You can use globs to download multiple files. For example, \*.html will download all html files. The glob expansion are done on the remote server. So, it depends on the operating system of the remote server.

ftp>mget \*.html

Fetching /ftptest/features.html to features.html

/ftptest/features.html 100% 2256 2.2KB/s 00:01

Fetching /ftptest/index.html to index.html

/ftptest/index.html 100% 2886 2.8KB/s 00:01

Fetching /ftptest/othertools.html to othertools.html

/ftptest/othertools.html 100% 2282 2.2KB/s 00:01

Fetching /ftptest/samplereport.html to samplereport.html

/ftptest/samplereport.html 100% 15KB 7.3KB/s 00:02

Fetching /ftptest/usage.html to usage.html

/ftptest/usage.html 100% 2340 2.3KB/s 00:01

**To view the file names before downloading, you can also use mls command as shown below.**

ftp>mls \*.html -

/ftptest/features.html

/ftptest/index.html

/ftptest/othertools.html

/ftptest/samplereport.html

/ftptest/usage.html

### 9. Uploading multiple files with mput command

Use mput to upload multiple files together. This works similar to the mget command. The following example uploads all the \*.html file from local server to remote server.

ftp>mput \*.html

### 10. Close a FTP connection

Without exiting the ftp prompt you may want to open a connection to another server. In that case, execute **close** command.

ftp> open ftp.your\_server.com

Already connected to NNN.com, use close first.

ftp> close

221 Goodbye.

ftp> open ftp.your\_server.com

## 2.Electronic mail

Email is a service which allows us to send the message in electronic mode over the internet. It offers an efficient, inexpensive and real time mean of distributing information among people.

### E-Mail Address

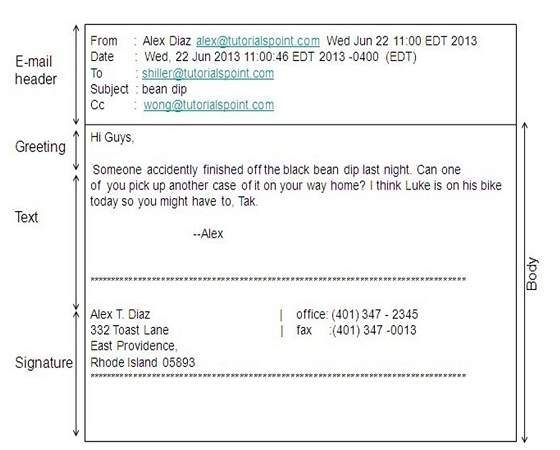
Each user of email is assigned a unique name for his email account. This name is known as E-mail address. Different users can send and receive messages according to the e-mail address.

E-mail is generally of the form username@domainname. For example, webmaster@tutorialspoint.com is an e-mail address where webmaster is username and tutorialspoint.com is domain name.

* The username and the domain name are separated by **& (at)** symbol.
* E-mail addresses are not case sensitive.
* Spaces are not allowed in e-mail address.

## E-mail Message Components

E-mail message comprises of different components: E-mail Header, Greeting, Text, and Signature. These components are described in the following diagram:



### E-mail Header

The first five lines of an E-mail message is called E-mail header. The header part comprises of following fields:

* From
* Date
* To
* Subject
* CC
* BCC

#### FROM

The **From** field indicates the sender’s address i.e. who sent the e-mail.

#### DATE

The **Date** field indicates the date when the e-mail was sent.

#### TO

The **To** field indicates the recipient’s address i.e. to whom the e-mail is sent.

#### SUBJECT

The **Subject** field indicates the purpose of e-mail. It should be precise and to the point.

#### CC

**CC** stands for Carbon copy. It includes those recipient addresses whom we want to keep informed but not exactly the intended recipient.

#### BCC

**BCC** stands for Black Carbon Copy. It is used when we do not want one or more of the recipients to know that someone else was copied on the message.

#### GREETING

Greeting is the opening of the actual message. Eg. Hi Sir or Hi Guys etc.

#### TEXT

It represents the actual content of the message.

#### SIGNATURE

This is the final part of an e-mail message. It includes Name of Sender, Address, and Contact Number.

## Advantages

E-mail has proved to be powerful and reliable medium of communication. Here are the benefits of **E-mail:**

* Reliable
* Convenience
* Speed
* Inexpensive
* Printable
* Global
* Generality

### Reliable

Many of the mail systems notify the sender if e-mail message was undeliverable.

### Convenience

There is no requirement of stationary and stamps. One does not have to go to post office. But all these things are not required for sending or receiving an mail.

### Speed

E-mail is very fast. However, the speed also depends upon the underlying network.

### Inexpensive

The cost of sending e-mail is very low.

### Printable

It is easy to obtain a hardcopy of an e-mail. Also an electronic copy of an e-mail can also be saved for records.

### Global

E-mail can be sent and received by a person sitting across the globe.

### Generality

It is also possible to send graphics, programs and sounds with an e-mail.

## Disadvantages

Apart from several benefits of E-mail, there also exists some disadvantages as discussed below:

* Forgery
* Overload
* Misdirection
* Junk
* No response

### Forgery

E-mail doesn’t prevent from forgery, that is, someone impersonating the sender, since sender is usually not authenticated in any way.

### Overload

Convenience of E-mail may result in a flood of mail.

### Misdirection

It is possible that you may send e-mail to an unintended recipient.

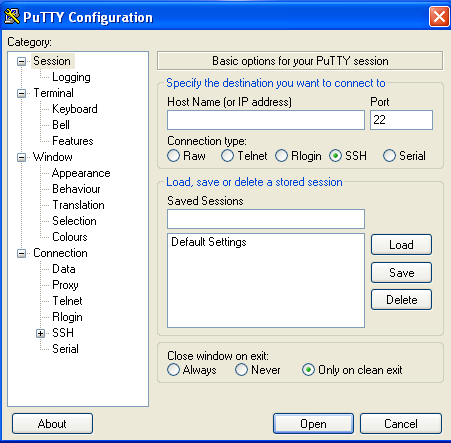
### Junk

Junk emails are undesirable and inappropriate emails. Junk emails are sometimes referred to as spam.

### No Response

It may be frustrating when the recipient does not read the e-mail and respond on a regular basis.

**3. Virtual Terminal**

****

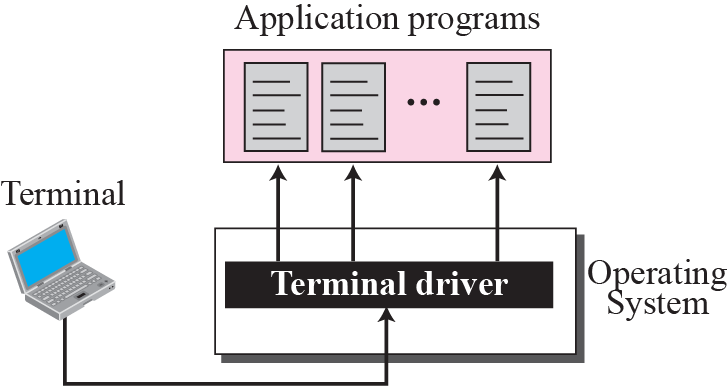
**TELNET**

* TELNET is an abbreviation for terminal network.
* It is a standard TCP/IP protocol for virtual terminal service.
* It is a general purpose client/server application program.
* TELNET enables the establishment of a connection to a remote system in such a way that the local terminal appears to be a terminal at the remote system.
* In other words, it allows the user to log on to a remote computer. After logging on, user can use the services available on the remote computer and transfer the results back to the local computer.

**LOGGING**

* To access the system, user logs into the system with user-id. The system also includes password checking to prevent an unauthorized user accessing the resources.
* Logging process can be local logging or remote logging.
* When the user logs into a local timesharing system, it is called local log-in.
* When the user wants to access an application program or utility located on a remote machine, it is called remote log-in.

**LOCAL LOG-IN**

****

* When the user types at the terminal, the keystrokes are accepted by the terminal driver.
* The terminal driver passes the characters to the operating system.
* The operating system interprets the combination of characters and invokes the desired application program.

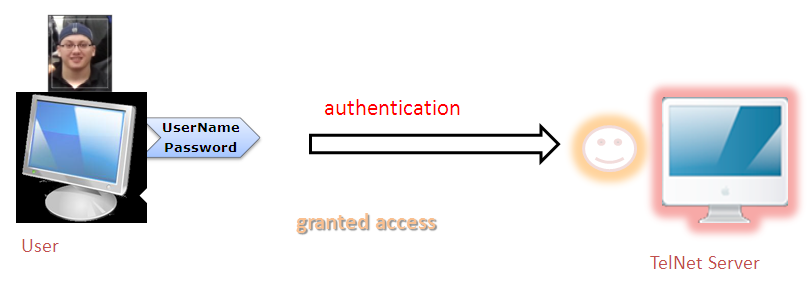
**Uses/Advantages**

* Can be used to send and receive information
* Administration of network elements
* Supports user authentication
* Collaboration of multiple users
* Can be used to send and receive information
* Most OS include an Application layer Telnet client
* All Telnet clients and servers implement a network virtual terminal.

**Disadvantages**

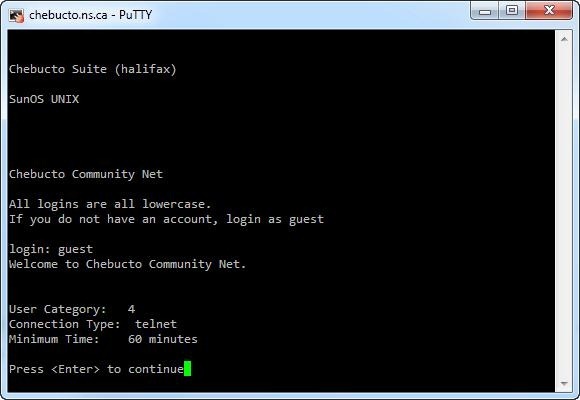
* Display only text and numbers
* No graphics
* No color
* No mouse (no menus, check boxes, *etc*)
* Do not support the transport of encrypted data

**Communication**



**User Interface**

This is a sample telnet session of a network host running virtually on a terminal in Win7 wherein the communication is bidirectional (2-way) the host displays text only, and then awaits an Enter key press to continue

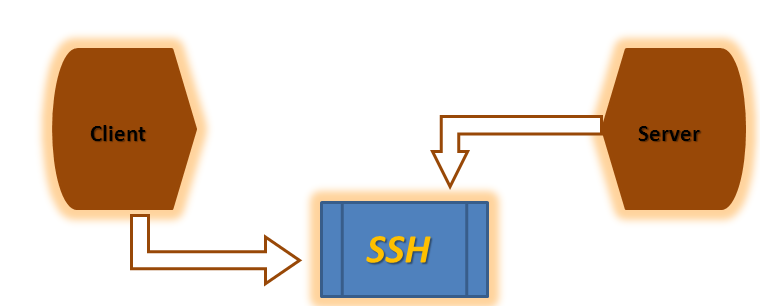


**SECURE SHELL**

SSH is a cryptographic network protocol for secure data communication, via a secure channel over an unsecure network of a server and a client.

**Secure Shell is an alternative protocol to TelNet and Rlogin which connects to Unix servers originally created in 1995.**

**Structure**



Normally a data is transmitted between client and server but not in a secure line, like internet. SSH is important in cloud computing to solve connectivity problems, avoiding the security issues of exposing a cloud-based virtual machine directly on the Internet. An SSH tunnel can provide a secure path over the Internet, through a firewall to a virtual machine

**Key Benefits**

1. Confidentiality - nobody can read the message content

2. Authentication (of both the client and server) - protection against IP spoofing, IP source routing, DNS spoofing, password interception and eavesdropping

3. Integrity - guarantee that data is unaltered on transit

**Advantages**

* Strong integrity checking via message authentication codes
* Better security through key exchange
* Browsing the web through an encrypted proxy connection, using the SSH server as a proxy
* Both ends authenticate themselves to the other end or all traffic encrypted

**Disadvantages**

* Can be considered a security risk by companies or governments who do not trust their users
* SSH2 has inherent design flaws which make it vulnerable to man-in-the-middle-attacks

|  |  |  |
| --- | --- | --- |
| **Command** | **Example** | **Description** |
| **1.     ls** | ls ls -alF | Lists files in current directory List in long format |
| **2.     cd** | cdtempdir  cd ..  cd ~dhyatt/web-docs | Change directory to tempdir  Move back one directory  Move into dhyatt's web-docs directory |
| **3.     mkdir** | mkdir graphics | Make a directory called graphics |
| **4.     rmdir** | rmdiremptydir | Remove directory (must be empty) |
| **5.     cp** | cp file1 web-docs  cp file1 file1.bak | Copy file into directory Make backup of file1 |
| **6.     rm** | rm file1.bak  rm \*.tmp | Remove or delete file Remove all file |
| **7.     mv** | mv old.html new.html | Move or rename files |
| **8.     more** | more index.html | Look at file, one page at a time |
| **9.     lpr** | lpr index.html | Send file to printer |
| **10.   man** | man ls | Online manual (help) about command |
| **Command** | Example | Description |
| **11.     passwd** | passwd | Change passwd |
| **12.     gcc (g++) <source>** | gccfile.c -o file  g++ fil2.cpp -o fil2 | Compile a program written in C Compile a program written in C++ |
| **13.     mail         (pine)** | mail me@tjhsst.edu < file1  pine | Send file1 by email to someone  Read mail using pine |
| **14.     who** | who | Lists who is logged on your machine |
| **15.     date** | date | Print out current date |
| **16.     cal<mo><yr>** | cal 9 2000 | Print calendar for September 2000 |
| **17.   logout (exit)** | logout or exit | How to quit a UNIX shell. |
| **18.Vi** | Vi filename.c | Editor |

**4. OVER VIEW OF ARPANET**

**ARPANET** was the **network** that became the basis for the Internet. Based on a concept first published in 1967, **ARPANET** was developed under the direction of the U.S. Advanced Research Projects Agency (ARPA). In 1969, the idea became a modest reality with the interconnection of four university **computers**.

Short for **Advanced Research Projects Agency Network**, **ARPANET** or **ARPAnet**began development in [1966](http://www.computerhope.com/history/1966.htm) by the United States [ARPA](http://www.computerhope.com/jargon/a/arpa.htm). ARPANET was a [Wide Area Network](http://www.computerhope.com/jargon/w/wan.htm) linking many Universities and research centers, was first to use [packet switching](http://www.computerhope.com/jargon/p/packetsw.htm), and was the beginning of what we consider the [Internet](http://www.computerhope.com/jargon/i/internet.htm) today. ARPANET was created to make it easier for people to access computers, improve computer equipment, and to have a more effective communication method for the military.

ARPANET first came into existence when the first two nodes were established between UCLA and Stanford Research Institute (SRI) in [1969](http://www.computerhope.com/history/1969.htm) followed shortly thereafter by UCSB and the University of Utah. In the picture below is an example of what ARPANET looked like in March [1977](http://www.computerhope.com/history/1977.htm), click the image to see a larger view of the image.

[](http://www.computerhope.com/jargon/a/arpanet.jpg)

ARPANET completed its transition to TCP/IP on January 2, 1983, was later replaced by[NSFNET](http://www.computerhope.com/jargon/n/nsfnet.htm) in [1990](http://www.computerhope.com/history/1990.htm), and then decommissioned on February 28, [1990](http://www.computerhope.com/history/1990.htm).

This may be considered as the breakthrough for many of current ideas, algorithms and Internet technologies. It started in 1960s funded by Advanced Research Projects Agency (ARPA), an organization of the US Defense Department and, therefore, named as ARPANET.

ARPANET was built to accommodate research equipment on packet switching technology and to allow resource sharing for the Department of Defense's contractors. The network interconnected research centers, some military bases and government locations. It soon became popular with researchers for collaboration through electronic mail and other services.

• It is basically a WAN. It was developed by the ARPA (Advanced Research Project Agency) in 1968 which is the research arm of 000.

• ARPANET was designed to service even a nuclear attack.

• Before ARPANET, the networks were basically the telephone networks which operated on the circuit switching principle.

• But this network was too vulnerable, because the loss of even one line or switch would terminate all the conversations.

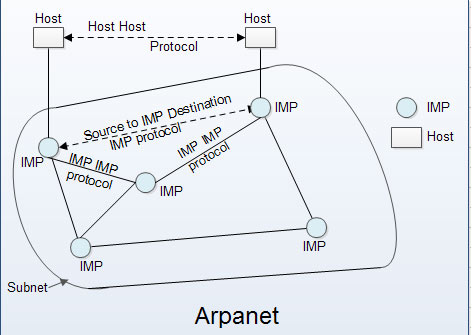
• ARPANET used the concept of packet switching network consisting of subnet and host[computers](http://ecomputernotes.com/fundamental/introduction-to-computer/what-is-computer" \o "computers" \t "_blank).

• The subnet was a datagram subnet and each subnet consists of minicomputers called IMPs (Interface Message Processors).

• Each node of the network used to have an IMP and a host connected by a short wire.

• The host could send messages of upto 8063 bits to its IMP which would break them into packets and forward them independently toward the destination.

• The subnet was the first electronic store-and-forward type packet switched network. So each packet was stored before it was forwarded.

[](http://ecomputernotes.com/images/Arpanet.jpg)

• The software for ARPANET was split into two parts namely subnet and host.

• In 1974 the TCP/IP model and [protocol](http://ecomputernotes.com/computernetworkingnotes/computer-network/protocol" \o "" \t "_self) were invented specifically to handle communication over internetwork because more and more networks were getting connected to ARPANET.

• The *TCP/IP*made the connection of LANs to ARPANET easy.

• During 1980s so many LANs were connected to ARPANET that finding hosts became increasingly difficult and expensive.

• So DNS (Domain Naming System) was created for organizing machines into domains and map host names onto **IP** address.

• .In 1983 the management of ARPANET was handed over to the Defense Communications Agency (DCA) which separated the military portion into a separate MILNET.

• By 1990 the ARPANET was shut down and dismantled, however MILNET continues to operate.