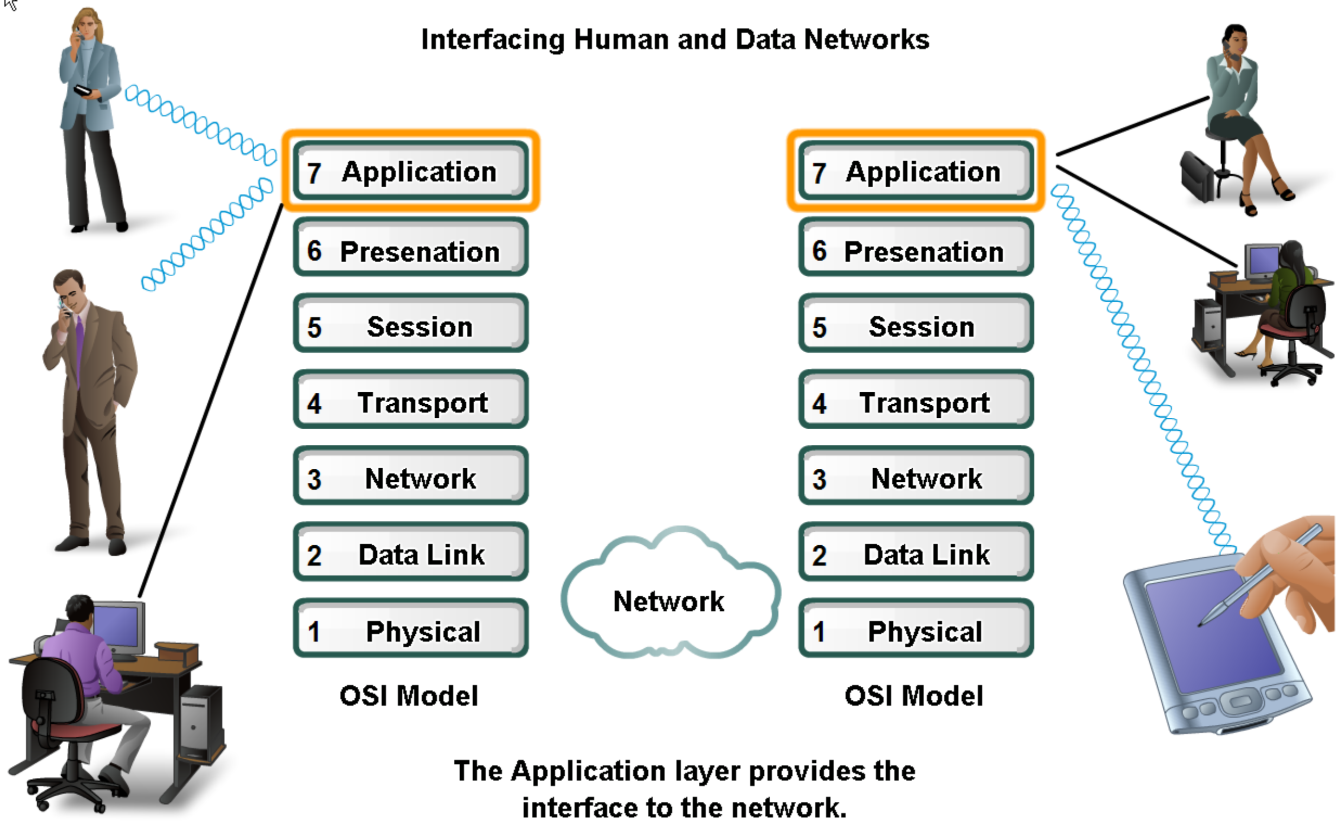
# Unit 5 Application Layer Functionality and Protocols

Version 4.0

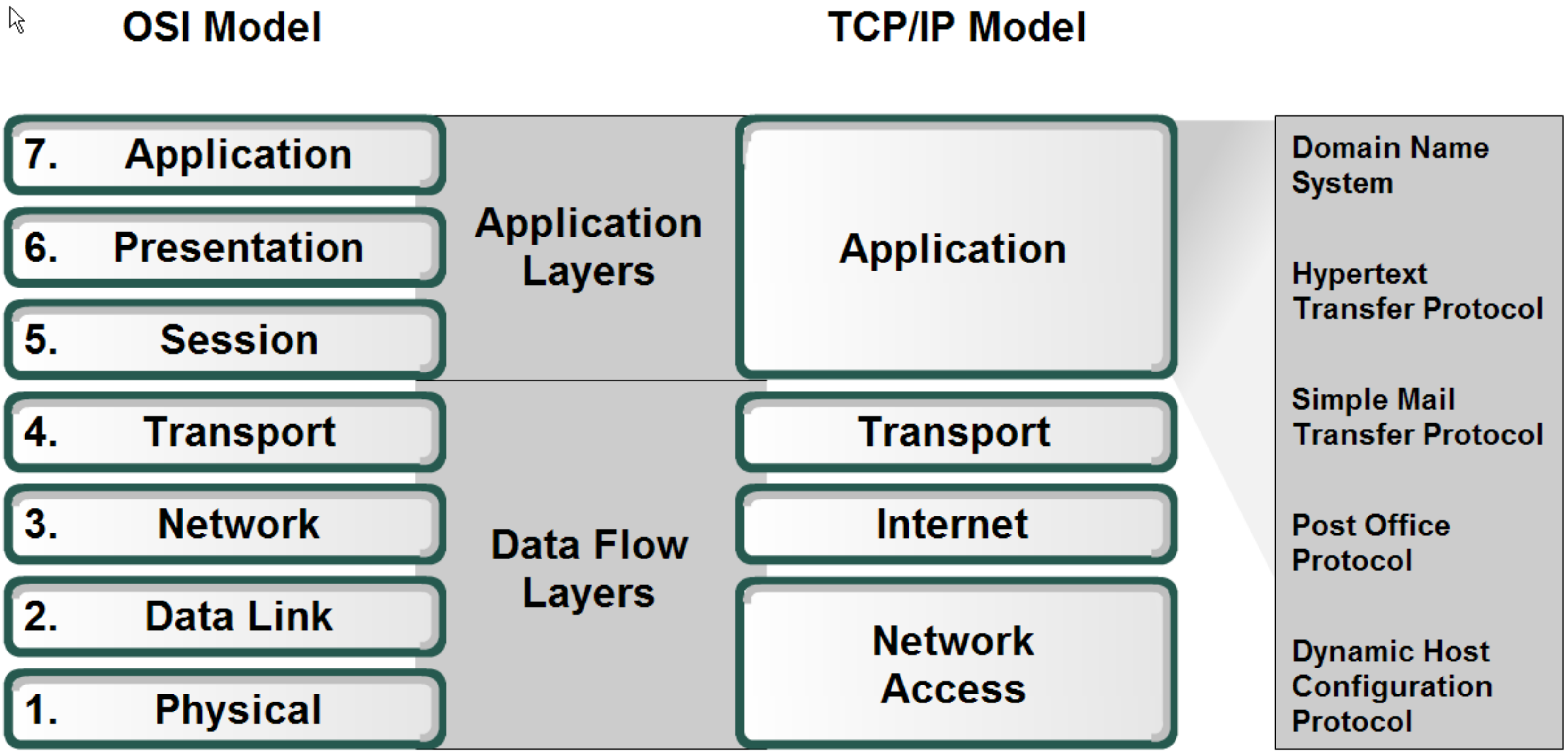
# Applications Layer – allows user to interface with the network!



Interfacing Human and Data Networks

The Application layer provides the interface to the network.

# Application Layer – Provides the interface between the applications on either end of the network.



OSI Model

TCP/IP Model

7. Application

6. Presentation

5. Session

4. Transport

3. Network

2. Data Link

1. Physical

Application Layers

Data Flow Layers

Application

Transport

Internet

Network Access

Domain Name System

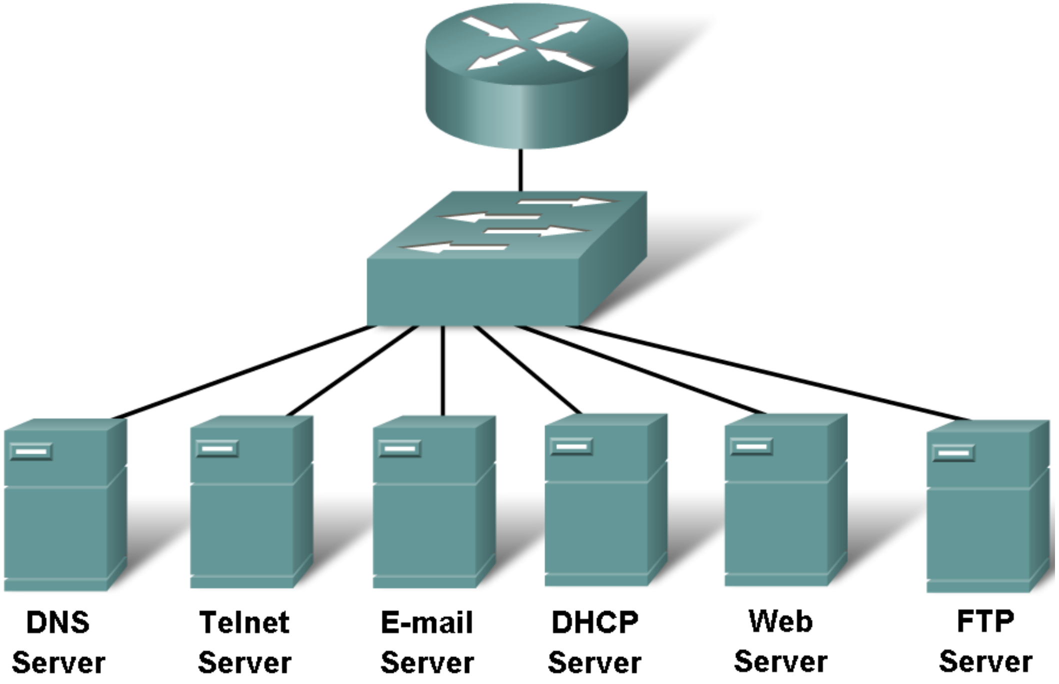
Hypertext Transfer Protocol

Simple Mail Transfer Protocol

Post Office Protocol

Dynamic Host Configuration Protocol

# Protocols and networks



Server Farm

# Protocols

DNS – Matches domain names with IP addresses

HTTP – Used to transfer data between clients/servers using a web browser

SMTP  & POP3 – used to send email messages from clients to servers over the internet

FTP – allows the download/upload of files between a client/server

Telnet – allows users to login to a host from a remote location and take control as if they were sitting at the machine (virtual connection)

DHCP – assigns IP addresses, subnet masks, default gateways, DNS servers, etcs. To users as they login the network

# Application layer software

2 types

- Applications – Provide the human (user) interface. Relies on lower layers to complete the communication process.
- Services – establish an interface to the network where protocols provide the rules and formats that govern how data is treated..

# How data requests occur & are filled

Client/server model

- Advantages:
- Centralized administration
- Security is easier to enforce

Application layer services and protocols

Peer-to-peer networking and applications

# Client/Server Model

Client –

- device requesting information (initiates the data exchange)
- Can also UPLOAD data to the servers

Server – device responding to the request

- How does it handle multiple request from multiple users and keep everything in order?
- Relies on support from the lower layer functions to distinguish between services and conversations.
- Server relies on a service called a server daemon – runs in the background and 'listens' for requests for that service. It can then exchange messages as appropriate & send requested data.

Examples:

- E-mail Client on an employee computer issues a request to the e-mail server for any unread e-mail. The server responds by sending the e-mail to the client.
- Conversations can originate with either party.

# Peer-to-Peer (P2P) Network Model

Two or more computers are connected and are able to share resources **without having a dedicated server**

Every end device can function as a client or server on a ‘per request’ basis

Resources are **decentralized** (information can be located anywhere)

Difficult to enforce security and policies

User accounts and access rights have to be set individually on each peer device

# Common Port Numbers

TCP

- FTP – 20-21
- Telnet – 23
- SMTP – 25
- DNS – 53 (Both TCP & UDP)
- HTTP – 80

UDP

- DHCP – 67 & 68
- POP – 110

# DNS(Domain Name System) Services

An Internet service that translates domain names into IP addresses.

**DNS resolver** – supports name resolution for other network applications and services that need it.

Devices are usually given 1 or more DNS Server addresses they can use for name resolution.

Uses different types of **resource records** to actually resolve the name/IP address issues

# DNS

The **Domain Name System (DNS)** is a central part of the Internet, providing a way to match names (a website you’re seeking) to numbers (the address for the website).

Anything connected to the Internet - laptops, tablets, mobile phones, websites - has an Internet Protocol (IP) address made up of numbers.

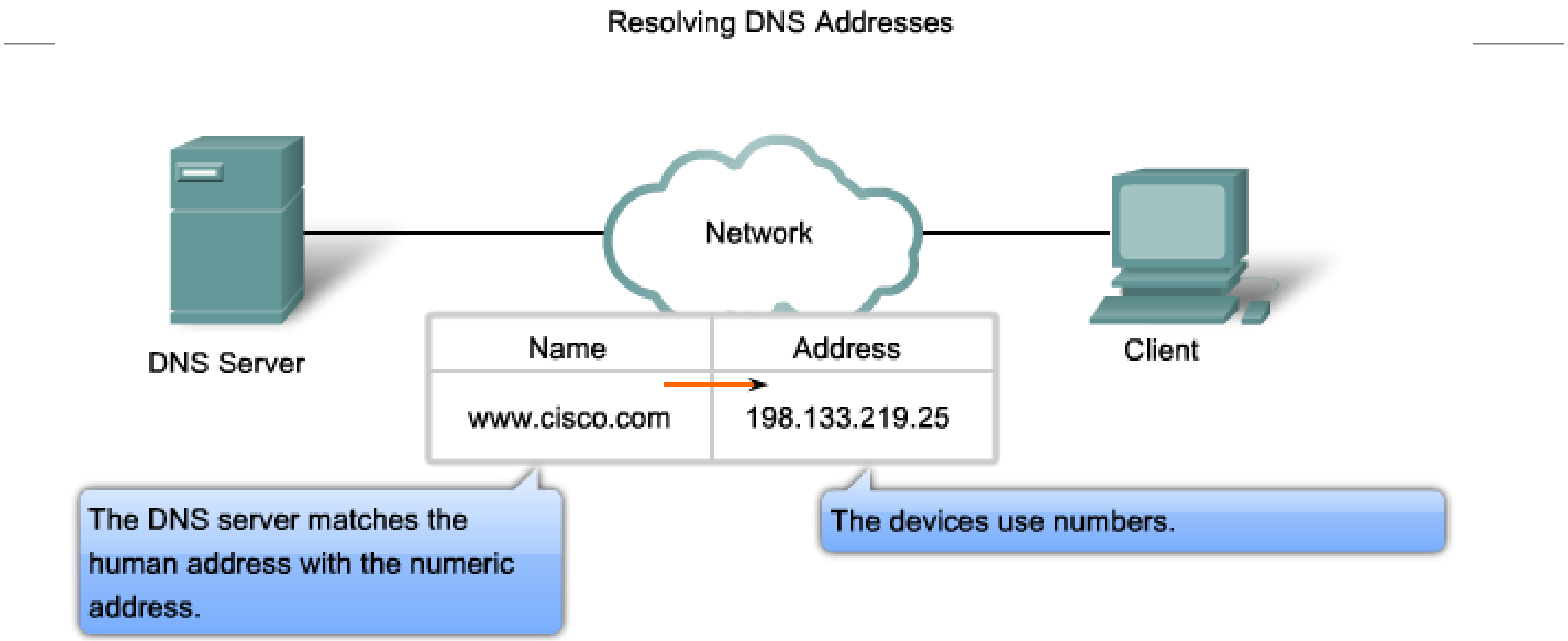
Your favourite website might have an IP address like 64.202.189.170, but this is obviously not easy to remember.

However a domain name such as bestdomainnameever.com is something people can recognize and remember.

DNS syncs up domain names with IP addresses enabling humans to use memorable domain names while computers on the Internet can use IP addresses.

# DNS Services and Protocol

Resolving DNS Addresses



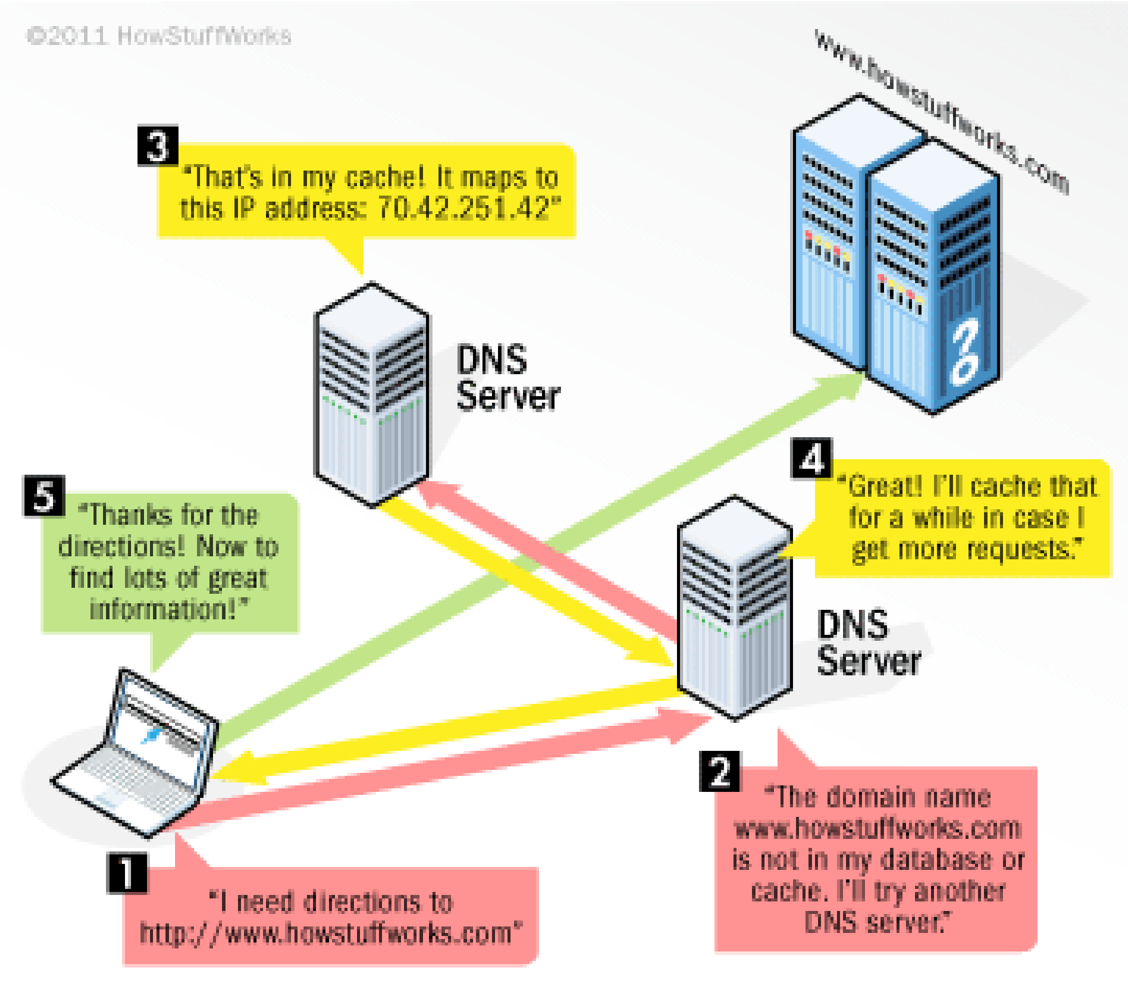
| Name | Address |
|---|---|
| www.cisco.com | 198.133.219.25 |

The DNS server matches the human address with the numeric address.

The devices use numbers.

DNS Servers resolve names to IP addresses. It would be difficult to remember the IP address of every website we like to visit, but we can remember names.

When you enter a URL into your Web browser, your DNS server uses its resources to resolve the name into the IP address for the appropriate Web server.


©2011 HowStuffWorks
www.howstuffworks.com
3
"That's in my cache! It maps to this IP address: 70.42.251.42"
DNS Server
4
"Great! I'll cache that for a while in case I get more requests."
5
"Thanks for the directions! Now to find lots of great information!"
DNS Server
2
"The domain name www.howstuffworks.com is not in my database or cache. I'll try another DNS server."
1
"I need directions to http://www.howstuffworks.com"

# Domain Name System Architecture

The Domain name system comprises of **Domain Names, Domain Name Space, Name Server** that have been described below:

## Domain Names

Domain Name is a symbolic string associated with an IP address. There are several domain names available;
some of them are generic such as **com, edu, gov, net** etc, while some country level domain names such as **au, in, za, us** etc.

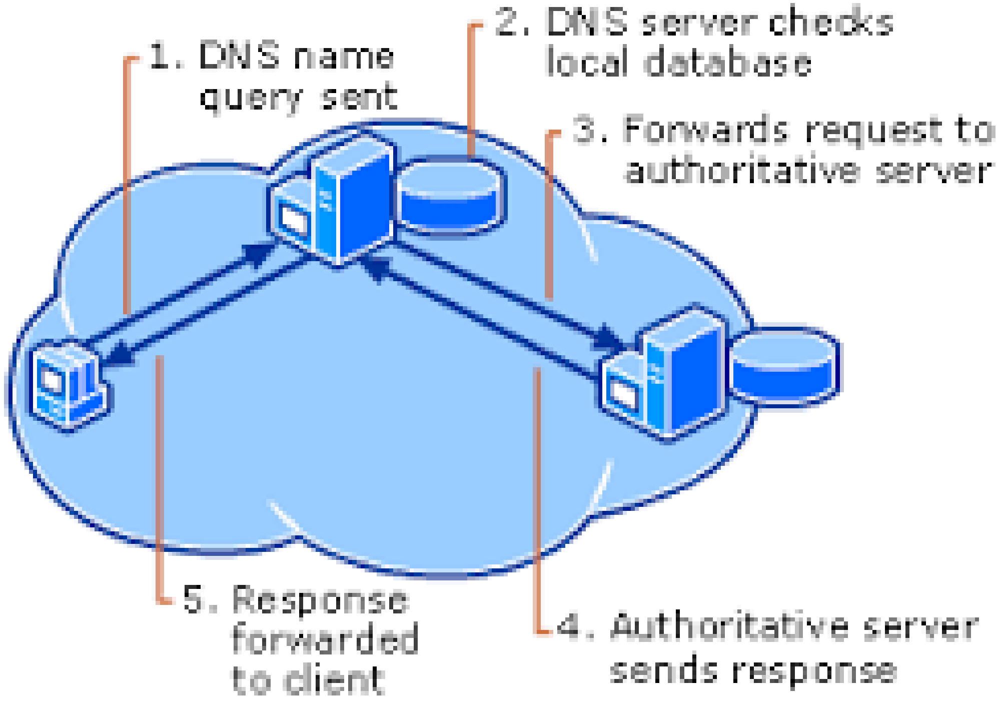
The following table shows the **Generic** Top-Level Domain names:

| | |
|---|---|
| Domain Name | Meaning |
| Com | Commercial business |
| Edu | Education |
| Gov | U.S. government agency |
| Int | International entity |
| Mil | U.S. military |
| Net | Networking organization |
| Org | Non profit organization |

# The following table shows the **Country top-level** domain names:

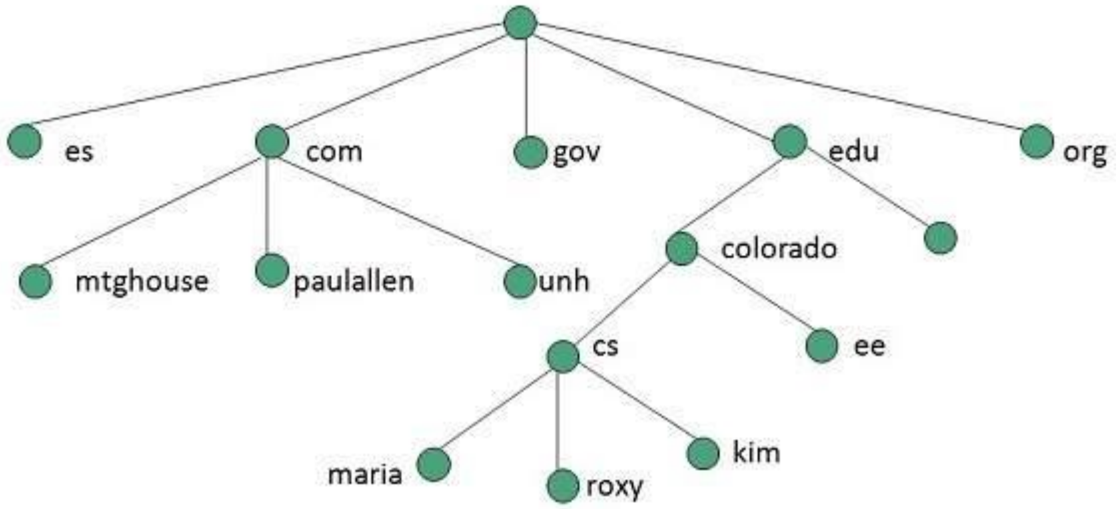
| | |
|---|---|
| Domain Name | Meaning |
| au | Australia |
| in | India |
| cl | Chile |
| fr | France |
| us | United States |
| za | South Africa |
| uk | United Kingdom |
| jp | Japan |
| es | Spain |
| de | Germany |
| ca | Canada |
| ee | Estonia |
| hk | Hong Kong |

## DNS Resolution


1. DNS name query sent
2. DNS server checks local database
3. Forwards request to authoritative server
4. Authoritative server sends response
5. Response forwarded to client

# Domain Name Space

The domain name space refers a hierarchy in the internet naming structure. This hierarchy has multiple levels (from 0 to 127), with a root at the top. The following diagram shows the domain name space hierarchy:



In the above diagram each subtree represents a domain. Each domain can be partitioned into sub domains and these can be further partitioned and so on.

# Name Server

Name server contains the DNS database. This database comprises of various names and their corresponding
IP addresses.
Since it is not possible for a single server to maintain entire DNS database, therefore, the information
is distributed among many DNS servers.

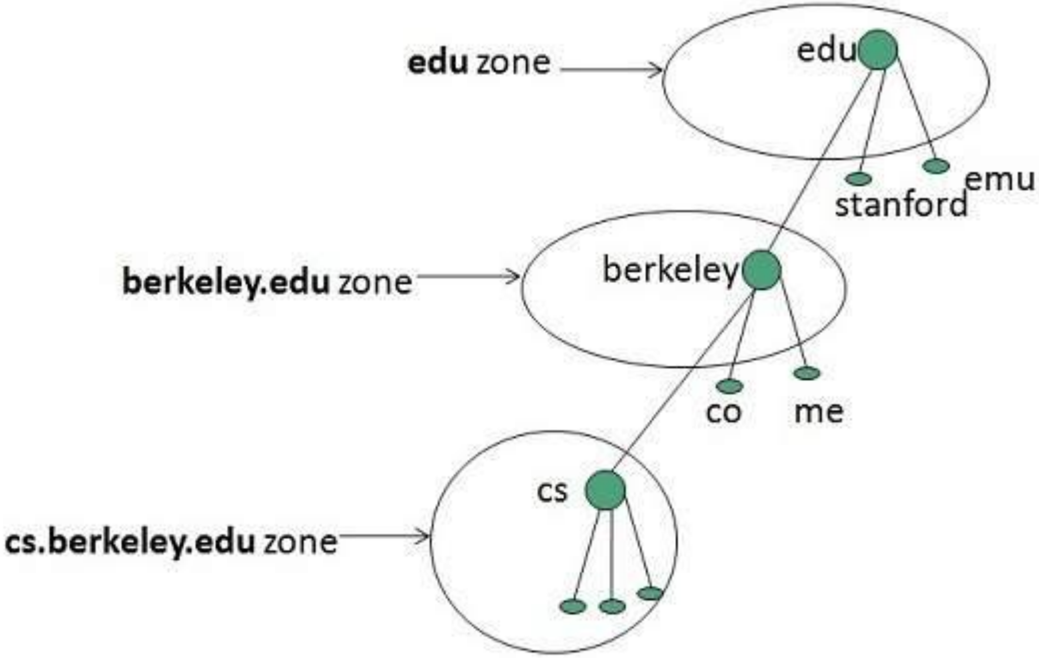
- Hierarchy of server is same as hierarchy of names.
- The entire name space is divided into the zones

# Zones

Zone is collection of nodes (sub domains) under the main domain. The server maintains a database called zone file for every zone.

**If the domain is not further divided into sub domains then domain and zone refers to the same thing.**

**The information about the nodes in the sub domain is stored in the servers at the lower levels however; the original server keeps reference to these lower levels of servers.**



# TYPES OF NAME SERVERS

**ROOT SERVER**

Root Server is the top level server which consists of the entire DNS tree. It does not contain the information about domains but delegates the authority to the other server

**PRIMARY SERVERS**

Primary Server stores a file about its zone. It has authority to create, maintain, and update the zone file.

**SECONDARY SERVER**

Secondary Server transfers complete information about a zone from another server which may be primary or secondary server. The secondary server does not have authority to create or update a zone file.

# DDNS

*The information contained in the domain name space must be stored. However, it is very inefficient and also unreliable to have just one computer store such a huge amount of information.*

A primary server loads all information from the disk file; the secondary server

loads all information from
the primary server.

When the secondary downloads

information from the primary, it is called zone transfer.

# DDNS

*The DNS master file must be updated dynamically. The Dynamic Domain Name System (DDNS) therefore was devised to respond to this need. In DDNS, when a binding between a name and an address is determined, the information is sent, usually by DHCP to a primary DNS server. The primary server updates the zone. The secondary servers are notified either actively or passively.*

# Basic Terminology(1)

## About Internet

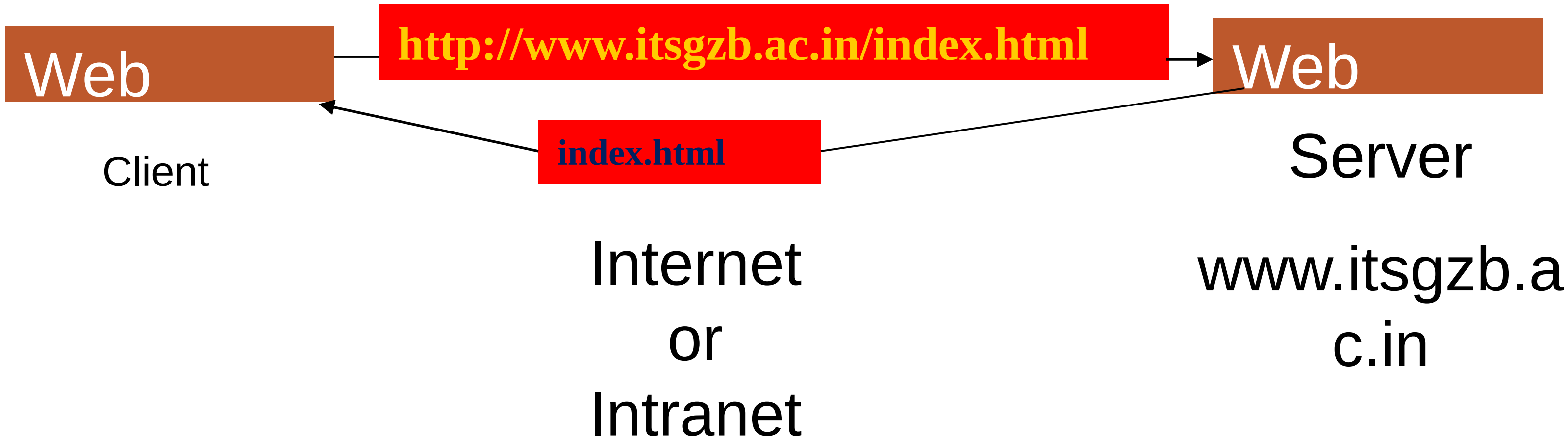
The **Internet** is a massive network of networks, a networking infrastructure.

It connects millions of computers together globally, forming a network in which any computer can communicate with any other computer as long as they are both connected to the Internet.

Information that travels over the Internet does so via a variety of languages known as protocols.

# Basic Terminology(1)

## About Internet



Web

http://www.itsgzb.ac.in/index.html

Web

index.html

Client

Server

Internet
or
Intranet

www.itsgzb.ac.in

Fig. 1: Internet

# About WWW

The **World Wide Web**, or simply *Web*, is a way of accessing information over the medium of the Internet.

It is an information-sharing model that is built on top of the Internet.

The Web uses the HTTP protocol, only one of the languages spoken over the Internet, to transmit data.

Web Services, which use HTTP to allow applications to communicate in order to exchange business logic, use the the Web to share information.

The Web also utilizes browsers, such as Internet Explorer or Firefox, to access Web documents called Web pages that are linked to each other via hyperlinks.

# About WWW (Cont’d..)

Web documents also contain graphics, sounds, text and video.

The Web is just one of the ways that information can be disseminated over the Internet.

The Internet, not the Web, is also used for e-mail, which relies on SMTP, Usenet news groups, instant messaging and FTP.

So the Web is just a portion of the Internet, albeit a large portion, but the two terms are not synonymous and should not be confused

# Remember it!!

The Web is not the Internet

The **Internet** is everything that happens using a *packet-switched* network of computers.

The **World Wide Web** is one particular family of *protocols* and *applications* which use the Internet.

## How many people use the Internet

| | |
|---|---|
| World Total | 407.1 million |

# Storing, locating, and transmitting information on the Web

## How information is located: the URL

- To move from one page of a document to another page, or to another document on the same or another Web site, the user clicks a **hyperlink** (usually just called a link) in the document shown in their Web client.

- Documents and locations within documents are identified by an address, defined as a Uniform Resource Locator, or **URL**. The following URL illustrates the general form:

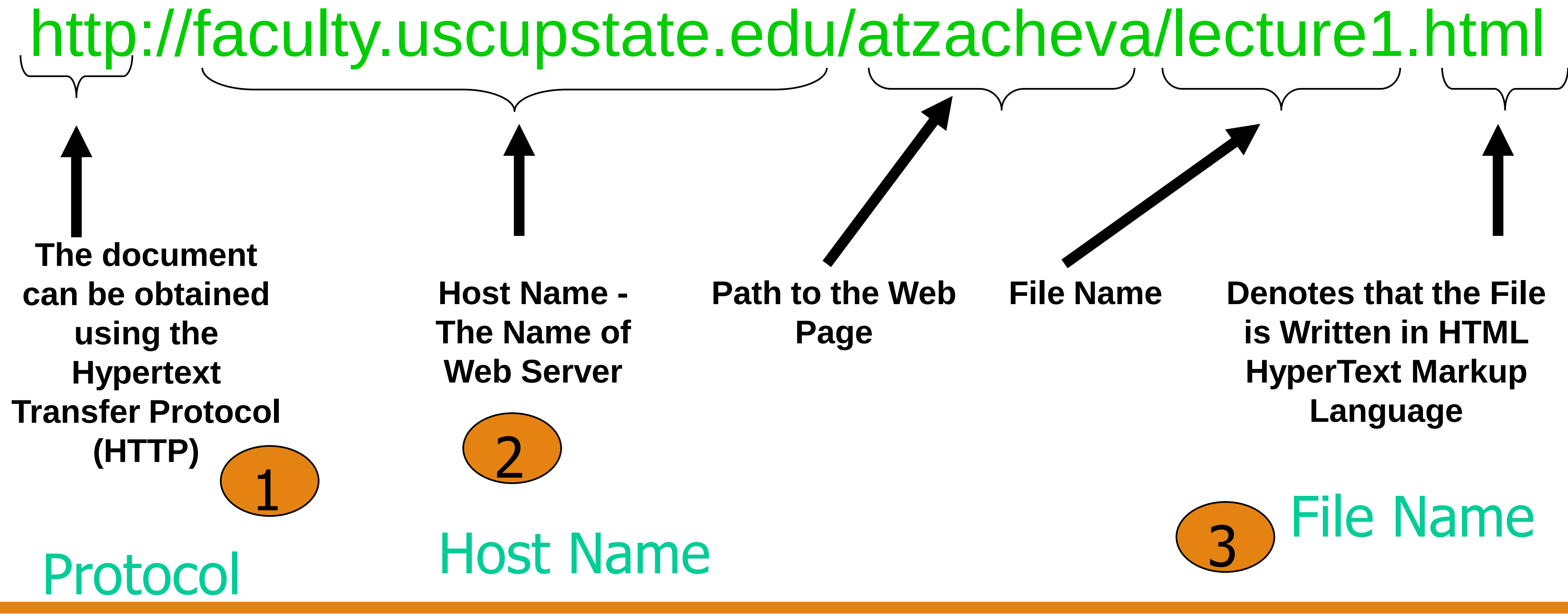
http://www.sybase.com/productsl

or

http://www.sybase.com/inc/corpinfo/mkcreate.html

# The Universal Resource Locator (URL)

**Each page of information on the web has a unique address called the URL at which it can be found.**



http://faculty.uscupstate.edu/atzacheva/lecture1.html

**The document can be obtained using the Hypertext Transfer Protocol (HTTP)**

**Host Name - The Name of Web Server**

**Path to the Web Page**

**File Name**

**Denotes that the File is Written in HTML HyperText Markup Language**

# Storing, locating, and transmitting information on the Web (Cont'd..)

URLs contain information about which server the document is on, and may also specify a particular document available to that server, and even a position within the document.

In addition, a URL may carry other information from a Web client to a Web server, including the values entered into fields in an HTML form.

For more information about URLs and addresses on the Web, see the material available at the following address:

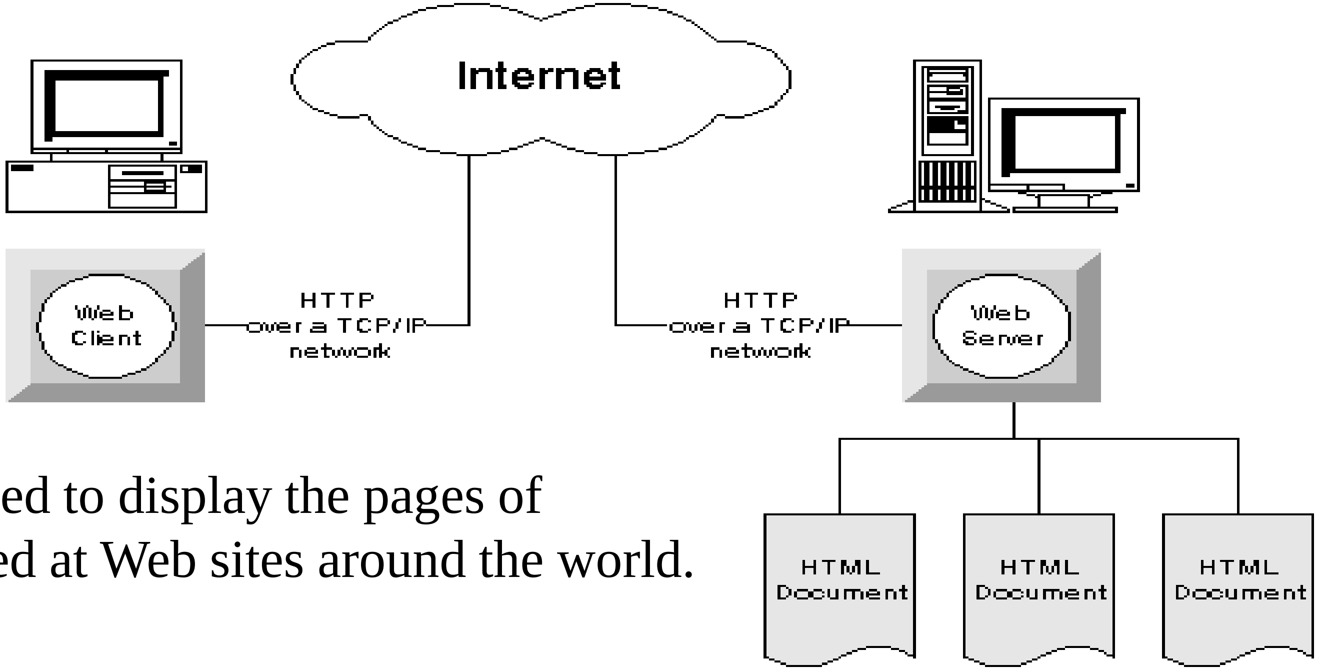
**http://www.w3.org/pub/WWW/Addressing/**

When a user clicks a link on a document on their Web client, the URL is sent to the server of the indicated Web site. The Web server locates the document, and sends the HTML to the Web client across the network.

# Accessing information on the web

As Figure 1 illustrates, information is stored at **Web sites**. Access to the information is managed by a **Web server** for the site. Users access the information using **Web clients**, which are also called **browsers**.

**Figure 1: Accessing information on the Web**



Browser is designed to display the pages of information located at Web sites around the world.

# Accessing information on the web

Information on the Web is stored in documents, using a language called **HTML** (HyperText Markup Language).

Web clients must interpret HTML to be able to display the documents to a user.

The protocol that governs the exchange of information between the Web server and Web client is named **HTTP** (HyperText Transfer Protocol).

# About HTTP

The standard Web transfer protocol is HTTP (HyperText Transfer Protocols).

It is used for document exchange between servers and clients (typically browsers) in the WWW.

To retrieve a document, the client first sends a request to the web server and waits for a reply

An HTTP service (a program that waits for http requests) on the server then handles the request and the document is sent to the client (over a connection established using TCP/IP – Transmission Control Protocol / Internet Protocol)

# About HTML

Programming Language

Used to create web pages

Use Hypertext documents for use on WWW.

# External data in HTML documents

**HTML documents** can include graphics or other types of data by referencing an external file (for example, a GIF or JPEG file for a graphic).

Not all these external formats are supported by all Web clients.

When the document contains such data, the Web client can send a request to the Web server to provide the relevant graphic.

If the Web client does not support the format, it does not request the information from the server.

# About Webpage

A **webpage** or web page is a document or resource of information that is suitable for the World Wide Web and can be accessed through a web browser and displayed on a computer screen

# Technicalities like Firewalls, CGI, Database & Networking

A Firewall is a system or group of systems that enforces as access control policy between two or more networks.

CGI is a standard for interfacing external applications with information servers, such as HTTP or Web Servers.

Database

Photoshop is a widely used image editing application from Adobe, often used as a benchmark for other imaging applications.

Graphics, animation, audio and video present on the web.

# Remember it!!!

Modem : “electronic device that converts computer signals into an analog signal in order to transmit data over a telephone line”.

Router: “It determines the nest network point to which a data packet should be forwarded enroute toward its destination.”

Server: “A server is a computer that handles requests for data, email, file transfers, and other network services from other computers (clients)”.

Web Server : A computer that is permanently connected to the Internet and allows people to read web pages located on that computer is called a web server.

# Remember it!!!

~~Favicon : it is a custom icon that appears next to website’s~~ URL in the address bar of a web browser.

Java: It is pre OOP which is used on web pages for interactivity.

Java Script: It allows things like popups, popunders and interactive elements on a webpage.

Cookie: It keeps track of user preferences, limit exposure of intrusive ads or keep track of login and password information for certain websites.

Hyperlink or link: It is an embedded html code that allows an Internet user go from webpage to webpage and website to website.

# Web Protocols

TCP: Transmission of data from an application to the network. It breaks data down into IP packets before they are sent, and for assembling the packets when they arrive.

**IP : Internet Protocol: C**ommunication with other computers. IP is responsible for the sending and receiving data packets over the Internet.

**HTTP:** takes care of the communication between a web server and a web browser. HTTP is used for sending requests from a web client (a browser) to a web server, returning web content (web pages) from the server back to the client.

HTTPS: takes care of secure communication between a web server and a web browser. HTTPS typically handles credit card transactions and other sensitive data.

# Web Protocols

**FTP (File Transfer Protocol):** a standard Internet protocol, is the simplest way to exchange files between computers on the Internet.

**SMTP:** SMTP is generally used to send messages from a mail client to a mail server.

**POP:** Protocol used to retrieve e-mail from a mail server. Most e-mail applications use the POP protocol.

# Self Assessment Questions

A ___________________ is a system or group of systems that enforces an access control policy between two or more networks.

_______ is full form of URL.

________ is used primarily as a tool to efficiently uploading and downloading files on the internet

The full form of HTML is _______

____________ is a widely used image editing application from adobe.

What is the difference between Internet and WWW.

# WWW

**World-Wide Web** hinges on three enabling protocols, the **HyperText Markup Language (HTML)** that specifies a simple markup language for describing hypertext pages, the **Hypertext Transfer Protocol (HTTP)** which is used by web browsers to communicate with web clients, and **Uniform Resource Locators (URL's)** which are used to specify the links between documents.

HTML: a simple language consisting of a small number of tags to delineate logical constructs within the text.

HTTP: Web browsers are able to communicate using a variety of protocols, such as FTP, Gopher and WAIS, the most common protocol in use on the Web is that designed specifically for the WWW project, the HyperText Transfer Protocol. It is a very simple protocol which uses a single round trip between the client and the server is used.

HTML is made up of numerous "TAGS" which are always included between the "less than" **<** or "greater than" **>** brackets. The text that is in between these brackets tells your Web Browser what to do, and how to recognize and display the content of your Web Page.

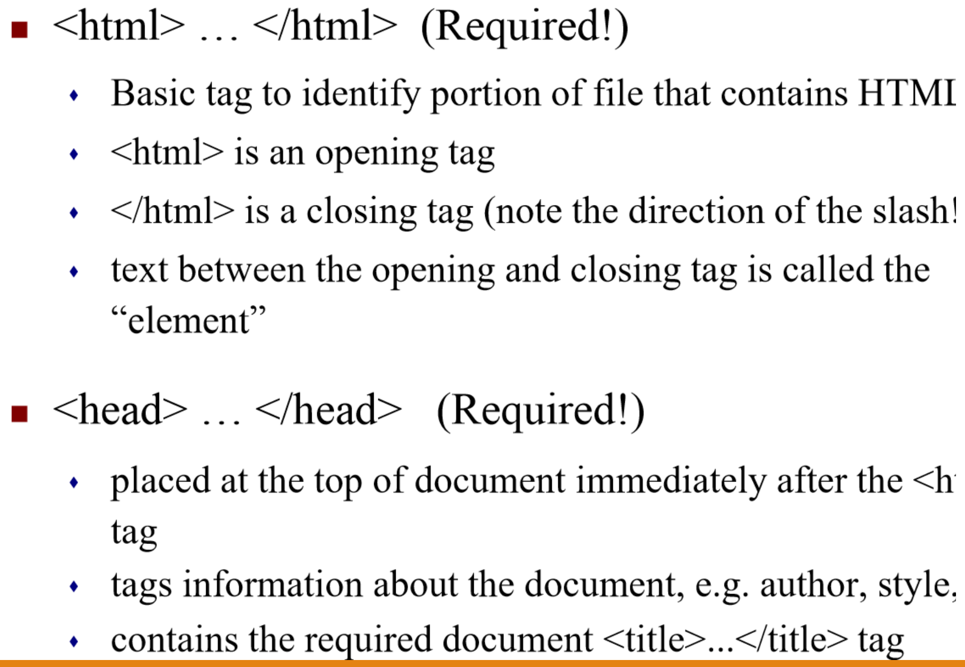
…

<b>hello</b>

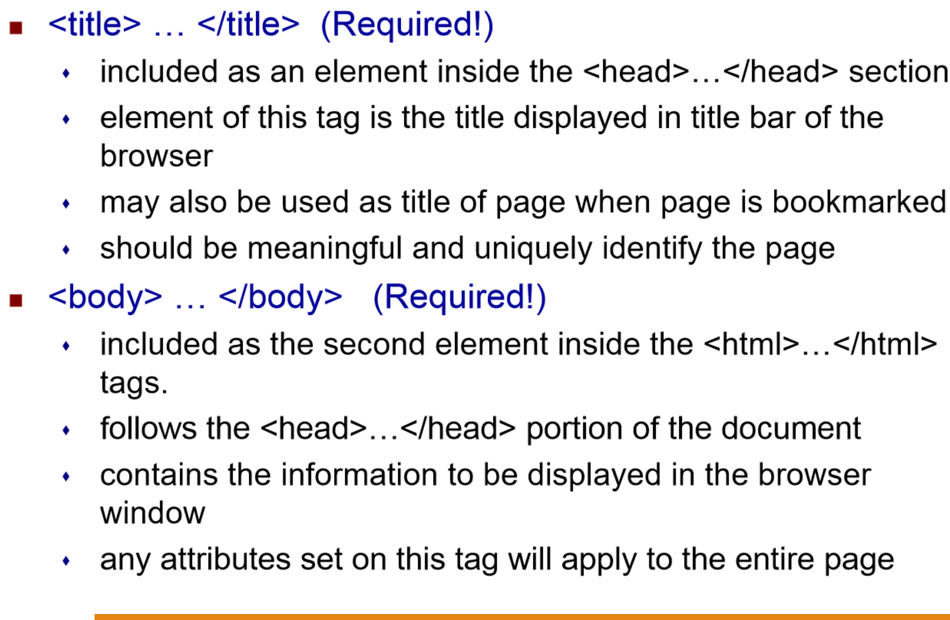
<i>hello</i>

<u>hello</u>

…



- <html> … </html> (Required!)
  - Basic tag to identify portion of file that contains HTMI
  - <html> is an opening tag
  - </html> is a closing tag (note the direction of the slash!
  - text between the opening and closing tag is called the "element"
- <head> … </head> (Required!)
  - placed at the top of document immediately after the <h tag
  - tags information about the document, e.g. author, style,
  - contains the required document <title>...</title> tag

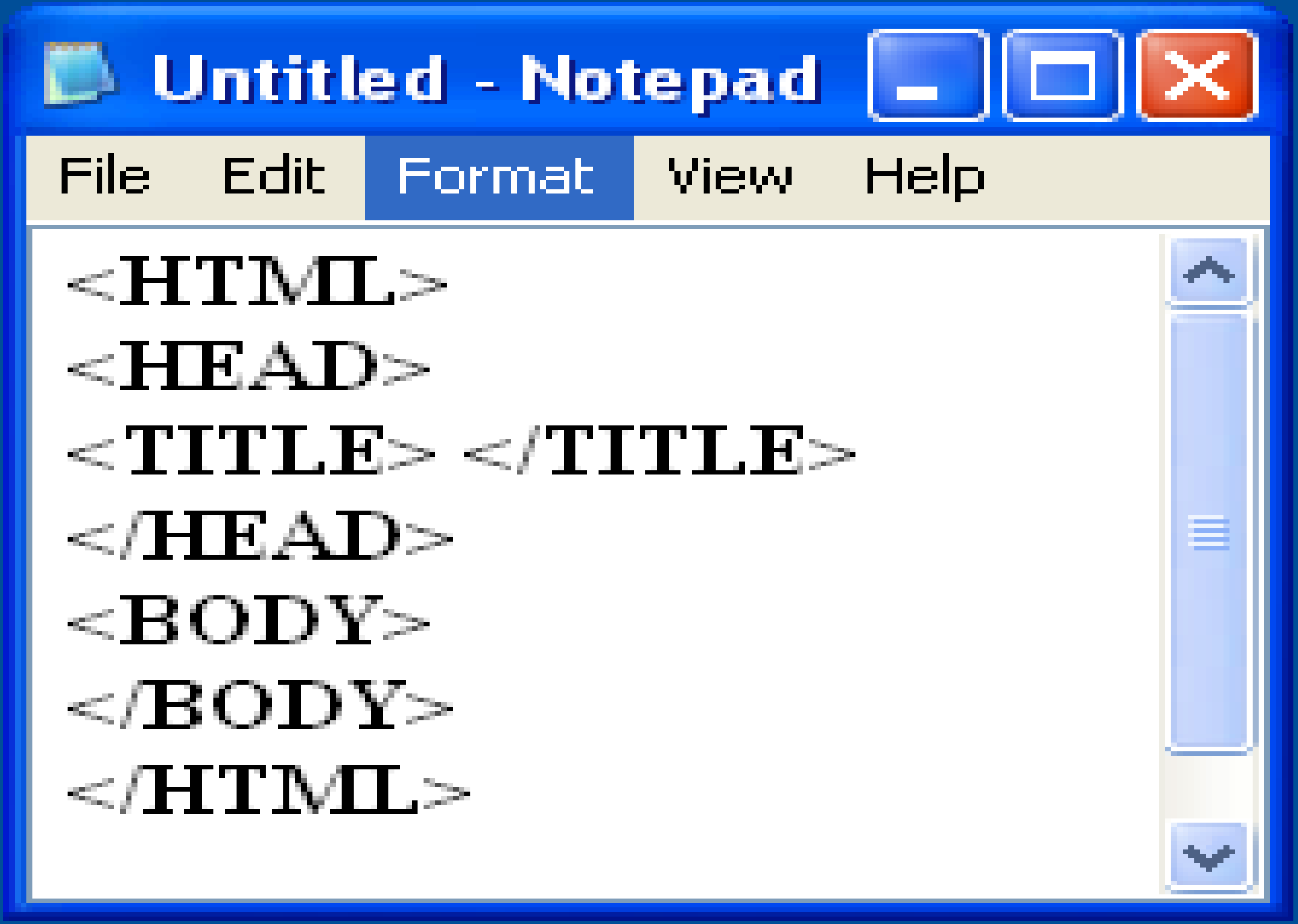


- <title> … </title> (Required!)
  - included as an element inside the <head>…</head> section
  - element of this tag is the title displayed in title bar of the browser
  - may also be used as title of page when page is bookmarked
  - should be meaningful and uniquely identify the page
- <body> … </body> (Required!)
  - included as the second element inside the <html>…</html> tags.
  - follows the <head>…</head> portion of the document
  - contains the information to be displayed in the browser window
  - any attributes set on this tag will apply to the entire page

# HTML Tags

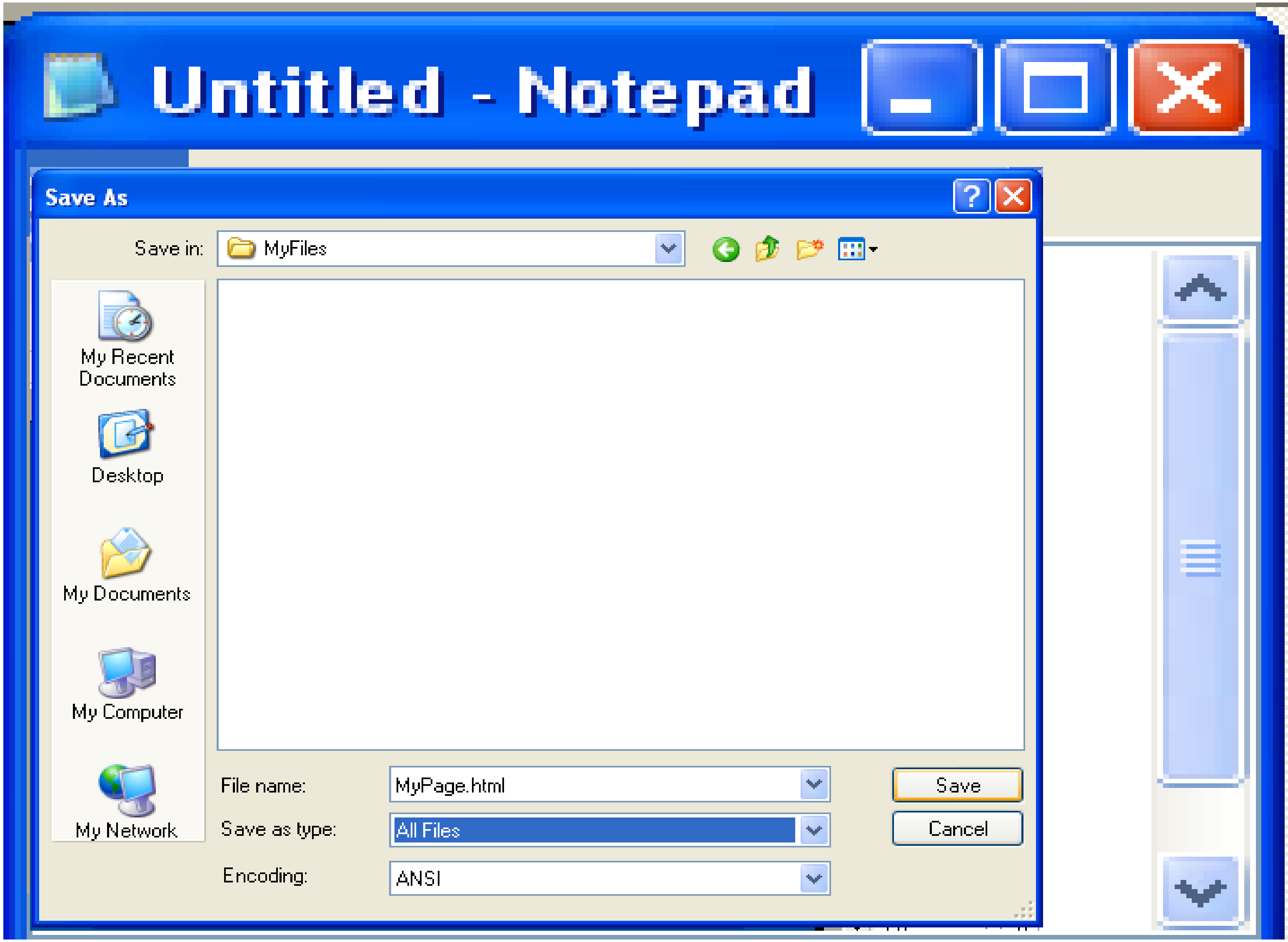
Create a new folder. and name it as "MyFiles“

Open notepad and type the following text in it



Untitled - Notepad
File
Edit
Format
View
Help

```
<HTML>
<HEAD>
<TITLE> </TITLE>
</HEAD>
<BODY>
</BODY>
</HTML>
```

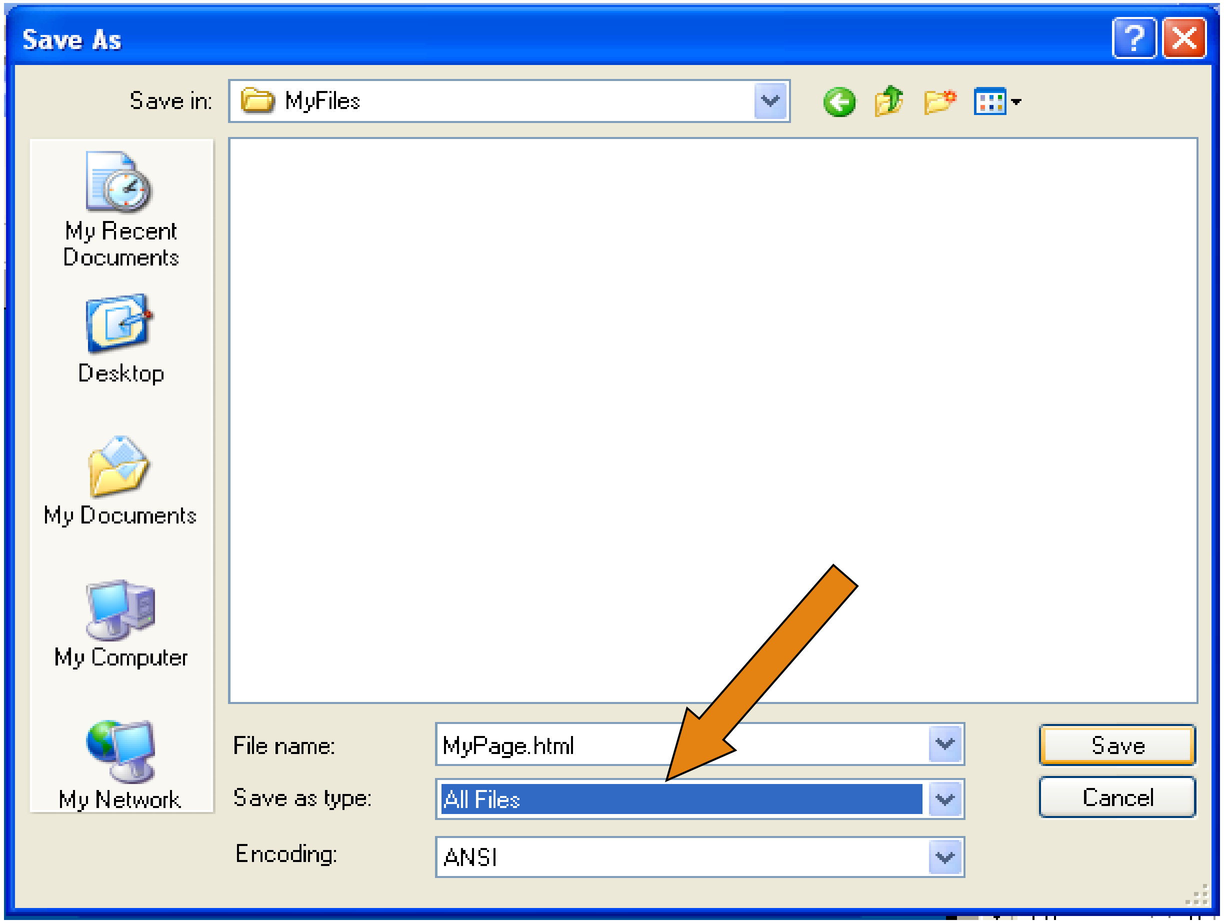


Save in: MyFiles

File name: MyPage.html

Save as type: All Files

Encoding: ANSI

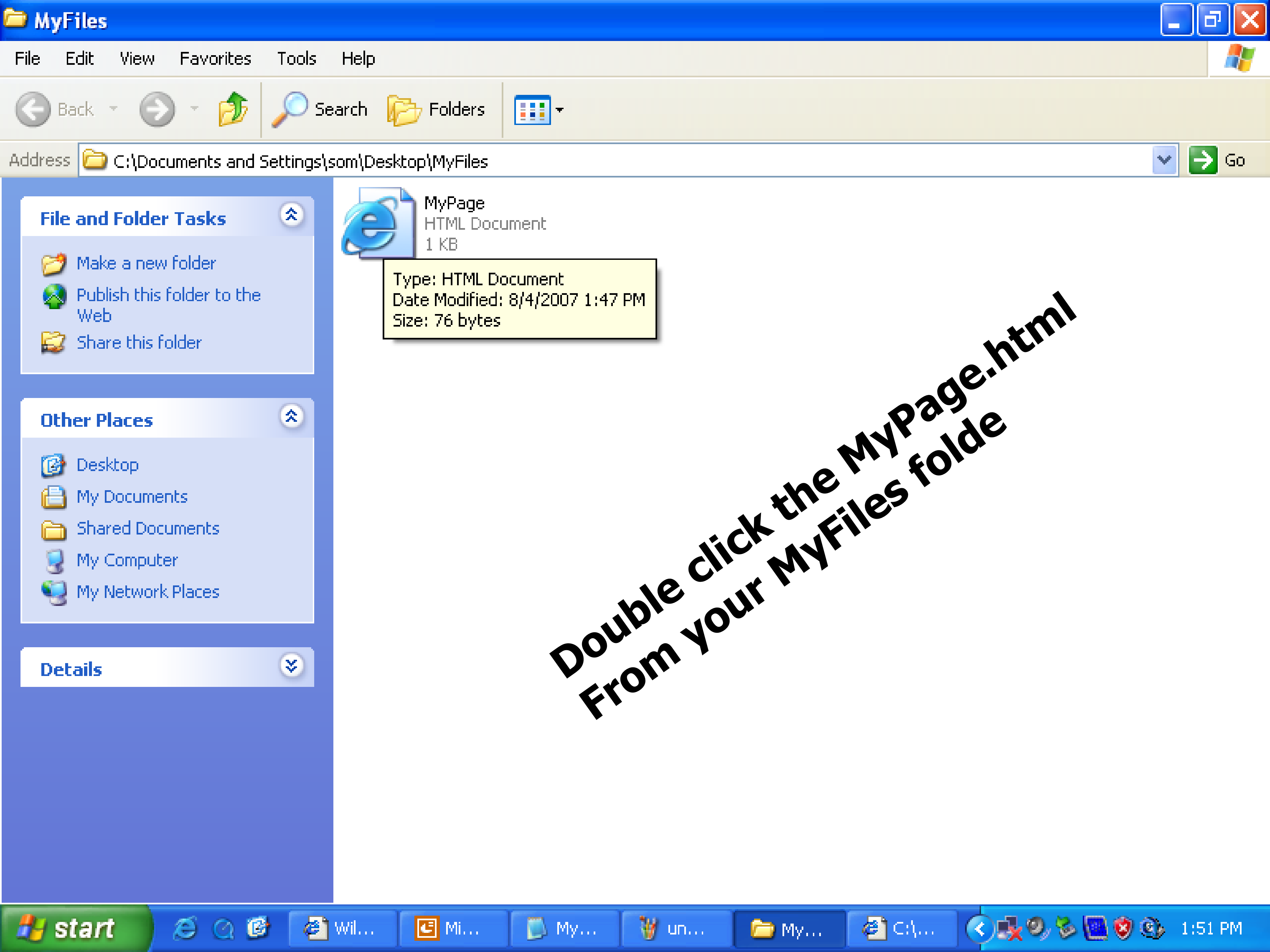


Save in: MyFiles

File name: MyPage.html

Save as type: All Files

Encoding: ANSI

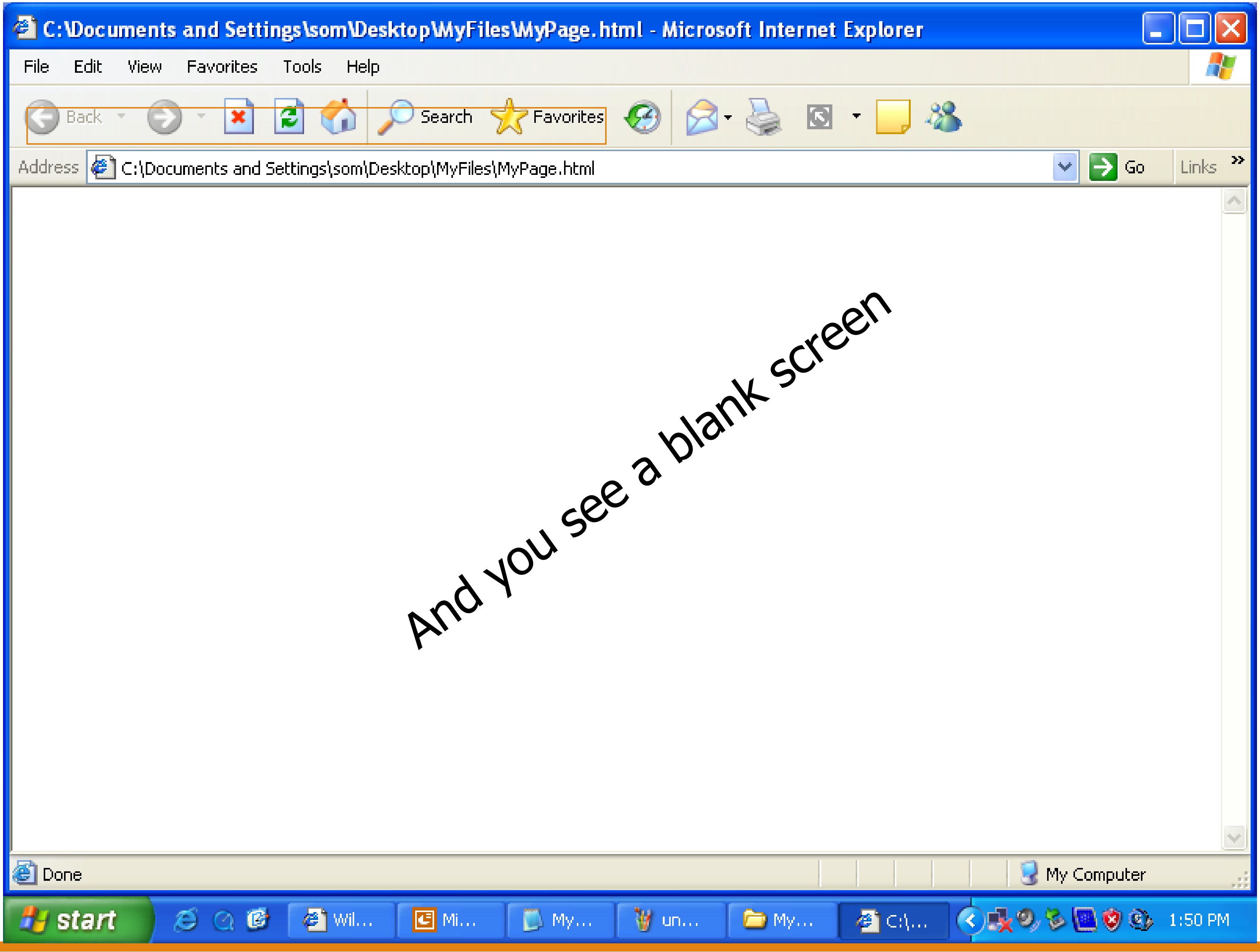


C:\Documents and Settings\som\Desktop\MyFiles

MyPage
HTML Document
1 KB

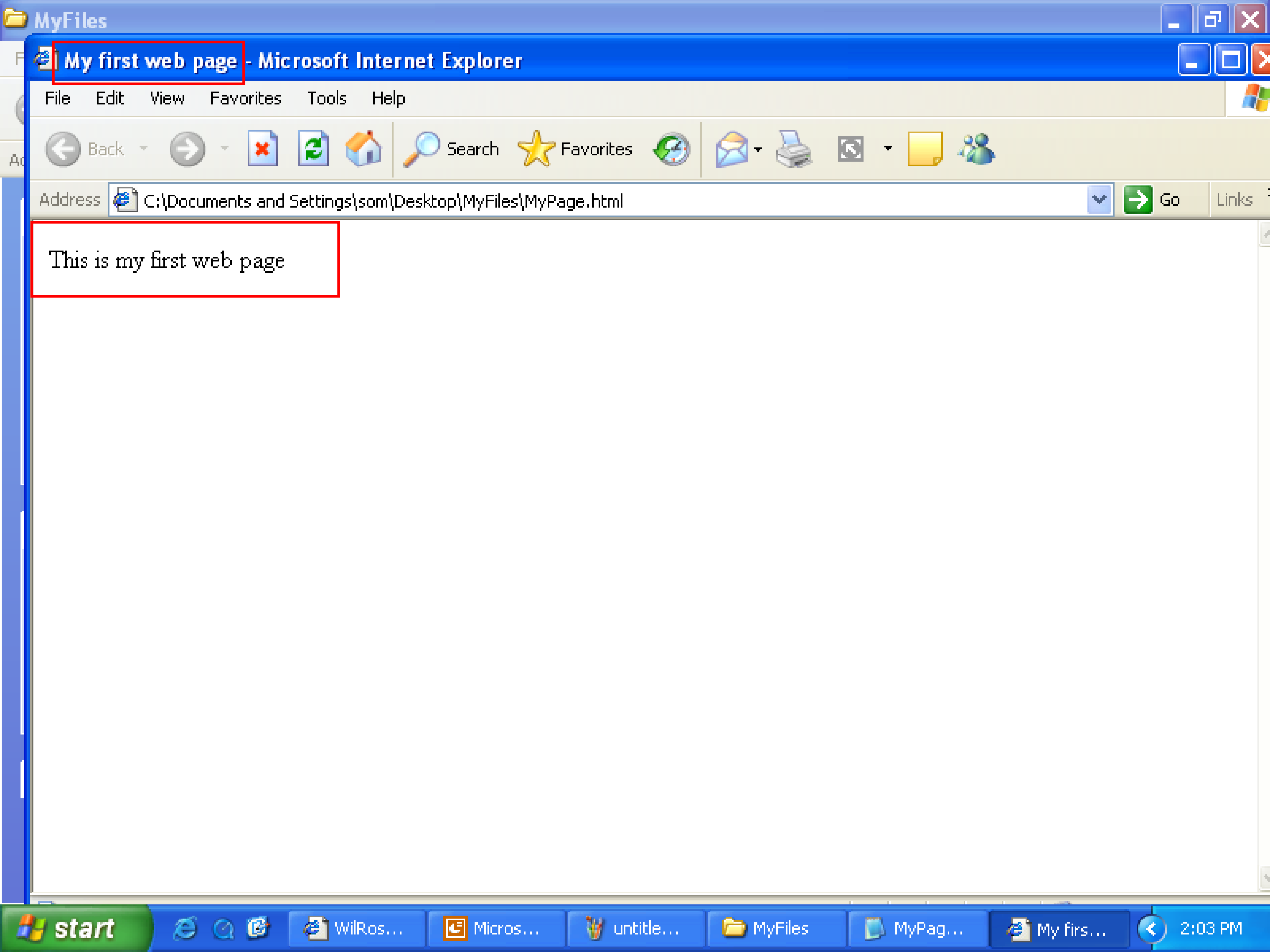
Type: HTML Document
Date Modified: 8/4/2007 1:47 PM
Size: 76 bytes

**Double click the MyPage.html**
**From your MyFiles folde**


C:\Documents and Settings\som\Desktop\MyFiles\MyPage.html - Microsoft Internet Explorer
File
Edit
View
Favorites
Tools
Help
Back
Search
Favorites
Address
C:\Documents and Settings\som\Desktop\MyFiles\MyPage.html
Go
Links
Done
My Computer
start
Wil...
Mi...
My...
un...
My...
C:\...
1:50 PM

And you see a blank screen

```
<HTML>
<HEAD>
<TITLE> My first web page</TITLE>
</HEAD>
<BODY> This is my first web page
</BODY>
</HTML>
```



My first web page - Microsoft Internet Explorer

Address C:\Documents and Settings\som\Desktop\MyFiles\MyPage.html

This is my first web page

# Mosaic

Mosaic was developed at the National Center for Supercomputing Applications (NCSA)at the University of Illinois Urbana-Champaign beginning in late 1992.

Mosaic was the web browser which led to the Internet boom of the 1990s.

# Internet

**What is the Internet?**

a network of networks – an inter-network, or Internet

**What are Internet protocols?**

the rules for transferring information between programs

**HTTP - hypertext transfer protocol**

**FTP - file transfer protocol**

**SMTP – simple mail transfer protocol**

**What is the World Wide Web?**

a set of HTML pages accessible using the HTTP protocol

# HTTP

1. HTTP transfer the browser sends a request for a document to the server. Included in this request is the description of the document being requested, as well as a list of document types that the browser is capable of handling.

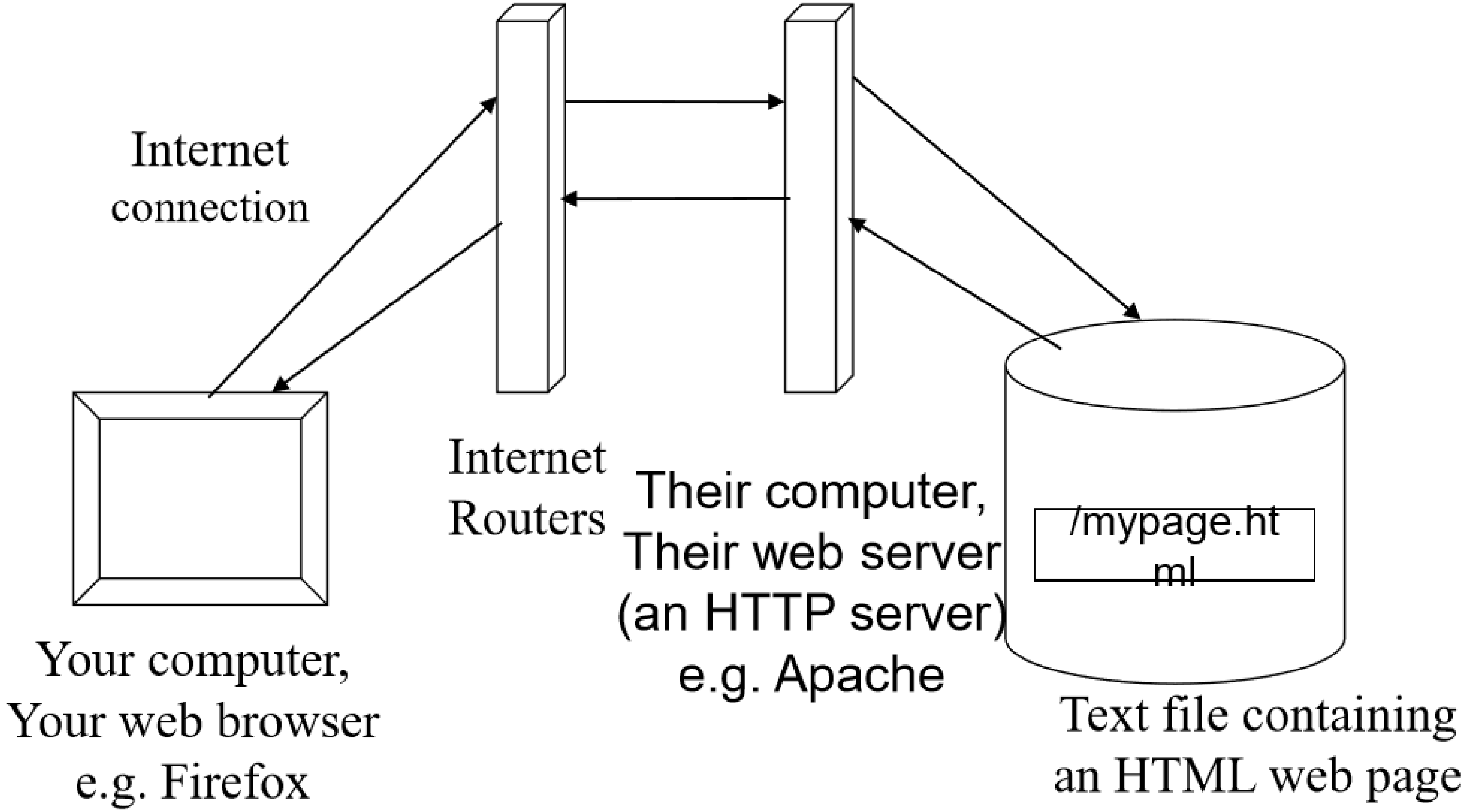
The Multipurpose Internet Mail Extensions (MIME) standard is used to specify the document types that the browser can handle. The browser is able to specify weights for each document type, in order to inform the server about the relative desirability of different document types.

2. In response to a query the server returns the document to the browser using one of the formats acceptable to the browser.

If necessary the server can translate the document from the format it is stored in into a format acceptable to the browser.

# How does a Web Browser (Firefox) fit in this picture?

- your browser connects, using the HTTP protocol, to a web server
    - the web server fetches an HTML web page and sends the HTML to your browser
    - your browser turns the HTML page into a nice-looking page on your screen


Internet connection
Internet Routers
Their computer, Their web server (an HTTP server) e.g. Apache
/mypage.html
Your computer, Your web browser e.g. Firefox
Text file containing an HTML web page

# HTTP

For example the server might have an image stored in the highly compressed JPEG image format, and if a browser capable of displaying JPEG images requests the image it would be returned in this format. However if a browser capable of displaying images only if they are in GIF format requested the same document the server would be able to translate the image and return the (larger) GIF image.

This provides a way of introducing more sophisticated document formats in future but still enabling older or less advanced browser to access the same information.

In addition to the basic **"GET" transaction** : HTTP is also able to support a number of other transaction types, such as **"POST"** for sending the data for fill-out forms back to the server and **"PUT"** which might be used in the future to allow authors to save modified versions of documents back to the server.

# WWW Service and HTTP

Steps:

- 1) URL is typed in the address bar.

2) Browser checks with DNS server to convert it to an IP address

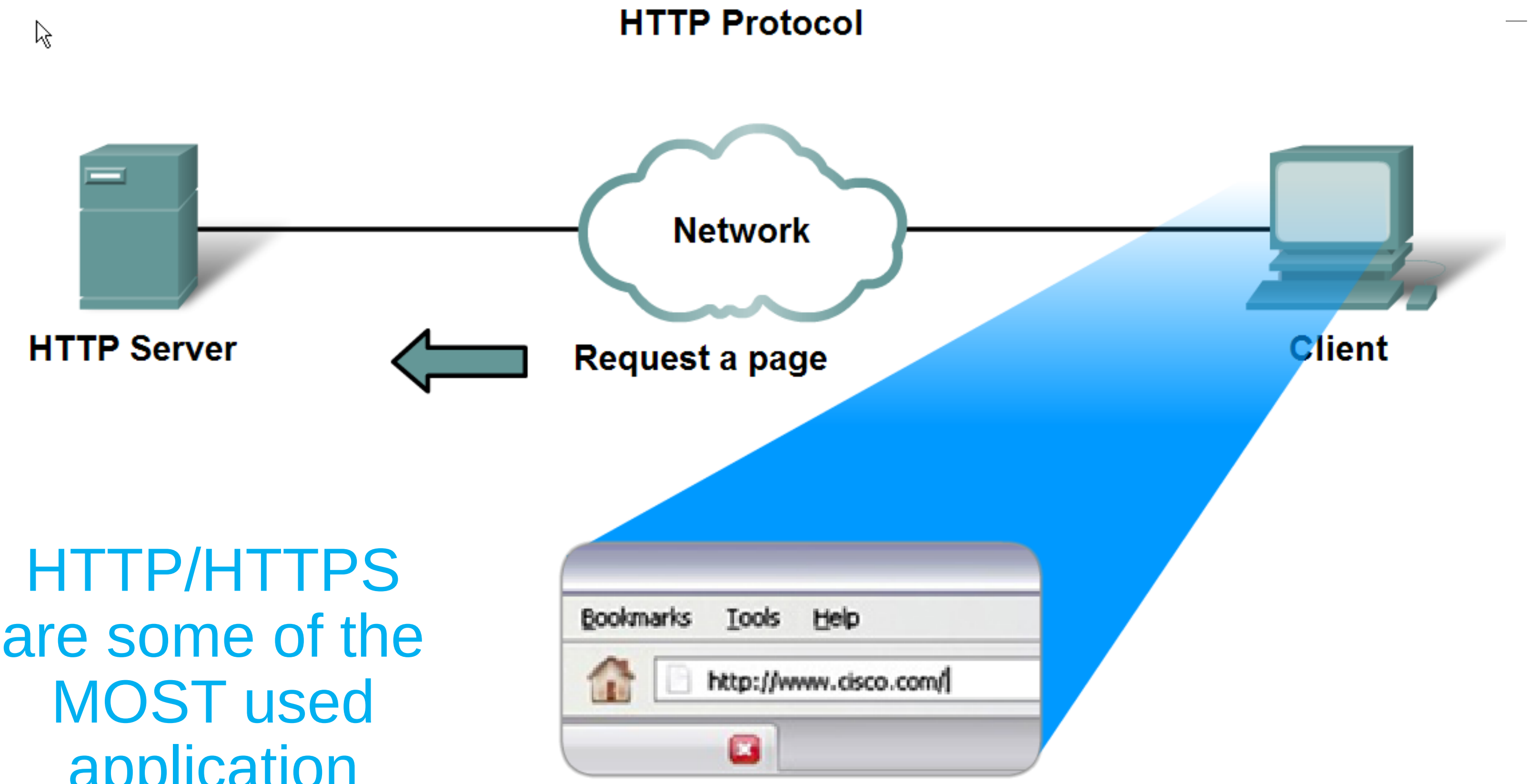
3) Connects to the server requested

4) Using **HTTP** or **HTTPS** protocol requirements, the browser sends a GET request to the server to ask for the desired html document (usually index.html)

5) The server sends the HTML code for the web page to the browser.

6) The browser interprets the HTML code and formats the page to fit the browser window.

# WWW Service and HTTP



HTTP/HTTPS are some of the MOST used application protocols!

# E-mail services and SMTP/POP protocols

E-mail is the most popular network service.

E-mail client (when people compose e-mail) is called Mail User Agent (MUA)

MUA allows messages to be sent/retrieved to and from your mailbox

Requires several applications and services

- POP or POP3 – deliver email from server to client (incoming messages)
- SMTP – handles outbound messages from clients

# Simple Mail Transfer Protocol (SMTP)

Email is emerging as the one of the most valuable service in internet today.

Most of the internet systems use SMTP as a method to transfer mail from one user to another.

SMTP is a push protocol and is used to send the mail whereas POP (post office protocol) or IMAP (internet message access protocol) are used to retrieve those mails at the receiver’s side.

# Simple Mail Transfer Protocol (SMTP)

SMTP is an application layer protocol.

The client who wants to send the mail opens a TCP connection to the SMTP server and then sends the mail across the connection.

The SMTP server is always on listening mode.

As soon as it listens for a TCP connection from any client, the SMTP process initiates a connection on that port (25).

After successfully establishing the TCP connection the client process sends the mail instantly.

# SMTP Protocol

The SMTP model is of two type :

End-to- end method

Store-and- forward method

The end to end model is used to communicate between different organizations whereas the store and forward method is used within an organization.

A SMTP client who wants to send the mail will contact the destination’s host SMTP directly in order to send the mail to the destination.

# SMTP Protocol

The SMTP server will keep the mail to itself until it is successfully copied to the receiver’s SMTP.

The client SMTP is the one which initiates the session let us call it as client-

SMTP and the server SMTP is the one which responds to the session request and let us call it as receiver-SMTP.

The client- SMTP will start the session and the receiver-SMTP will respond to the request.

# SMTP Protocol

In the SMTP model user deals with the **user agent** (UA) for example Microsoft outlook, netscape, Mozilla etc.

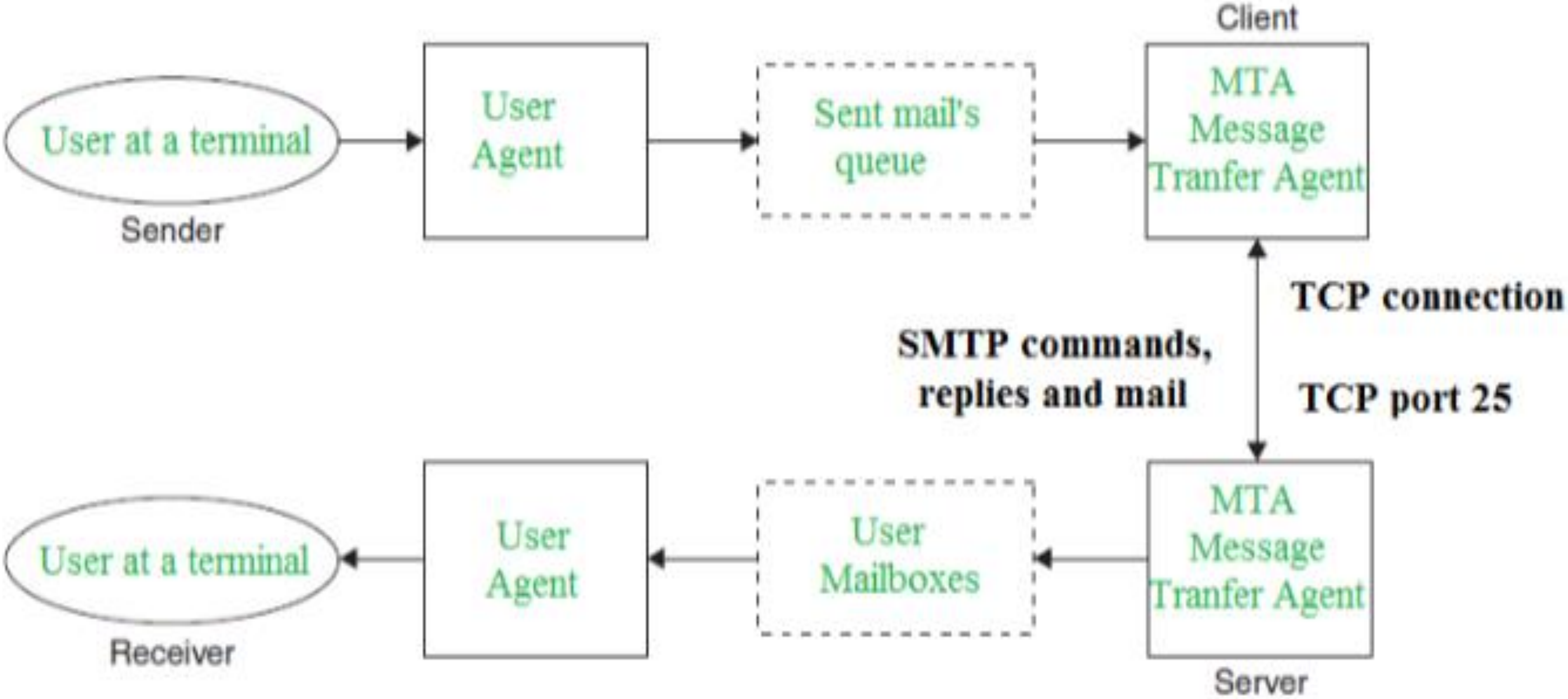
In order to exchange the mail using TCP, MTA is used.

The users sending the mail do not have to deal with the MTA

It is the responsibility of the **system admin** to set up the local MTA.

The **MTA** maintains a small queue of mails so that it can schedule repeat delivery of mail in case the receiver is not available.

The MTA delivers the mail to the mailboxes and the information can later be downloaded by the user agents.

Client
User at a terminal
Sender
User Agent
Sent mail's queue
MTA Message Tranfer Agent
TCP connection
SMTP commands, replies and mail
TCP port 25
User at a terminal
Receiver
User Agent
User Mailboxes
MTA Message Tranfer Agent
Server

**Both the SMTP-client and MSTP-server should have 2 components:**

User agent (UA)

Local MTA

# Communication between sender and the receiver :

The senders, user agent prepare the message and send it to the MTA .

The MTA functioning is to transfer the mail across the network to the receivers MTA.

# SENDING EMAIL:

Mail is send by a series of request and response messages between the client and a server.

The message which is send across consists of a header and the body.

A null line is used to terminate the mail header.

Everything which is after the null line is considered as body of the message which is a sequence of ASCII characters.

The message body contains the actual information read by the receipt.

# RECEIVING EMAIL:

The user agent at the server side checks the mailboxes at a particular time of intervals.

If any information is received it informs the user about the mail. When user tries to read the mail it displays a list of mails with a short description of each mail in the mailbox.

By selecting any of the mail user can view its contents on the terminal.

# Main Components of SMTP

## Three parts of a mail message:

**Envelope** - Includes recipient and sender addresses using the MAIL and RCPT commands.

**Headers** - Each header has a name followed by a colon and its value. Some headers are From, Date, Reply To, Received, Message ID, To, and Subject.

**Body -** The contents of the message sent in 7 bit ASCII code.

**SMTP Commands:**

**HELO -** Sent by client with domain name such as mymachine.mycompany.com.

**MAIL -** From <myself@mymachine.mycompany.com>

**RCPT -** To <myfriend@theirmachine.theirorg.org>

**DATA -** Sends the contents of the message. The headers are sent, then a blank line, then the message body is sent. A line with "." and no other characters indicates the end of the message.

**QUIT**

# Main Components of SMTP

| Property | Description |
|---|---|
| **Body** | Body text of the email. |
| **From** | Sender's email address. |
| **FromName** | Sender's name. |
| **IsBodyHTML** | Boolean value which is set to True if the body of the email message is in HTML format. |
| **SmtpServerName** | Name of server through which mail is routed. |
| **SmtpServerPort** | Port through which mail is routed. |
| **SmtpUserName** | Username for SMTP server. |
| **SmtpUserPassword** | Password for SMTP server. |
| **Subject** | Subject of e-mail note. |
| **To** | Email address to which you are sending the note. To send note to more than one recipient, separate addresses using commas. |

| Method | Description |
|---|---|
| **AddAttachment** | Use this method to include an attachment with an email sent using the Send method. |
| **Send** | Use this method to send an email via SMTP. |

# E-mail services and SMTP/POP protocols

What do servers require?

1) Must be running SMTP!
2) Also operates
    1) Mail Transfer Agent (MTA) – used to forward email

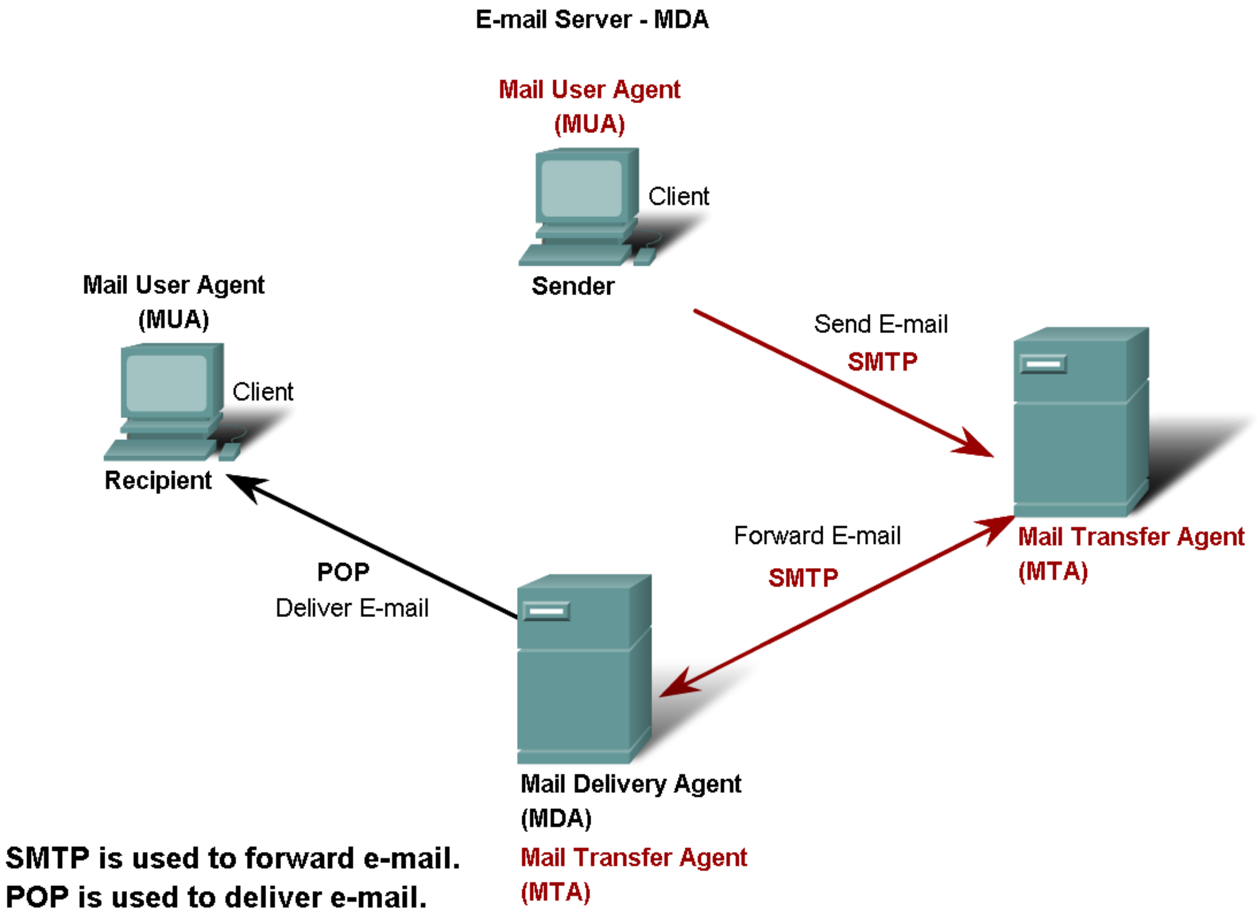
  1) Receives email from the clients MUA

  2) Uses SMTP to route email between SERVERS!

  3) Passes email  to the MDA for final delivery

    2) Mail Delivery Agent (MDA) – receives messages from MUA or from the MTA on another server
3) For two e-mail servers to talk – MUST run SMTP and MTA in order to transfer mail between the 2 servers!
4) Some clients run Lotus Notes, Groupwise, or MS Exchange.  They have their own proprietary protocol for handling e-mail.

# E-mail services and SMTP/POP protocols



**SMTP is used to forward e-mail.**
**POP is used to deliver e-mail.**

# Other Mail Protocols

SMTP - Simple Mail Transport Protocol is used on the internet, it is not a transport layer protocol but is an application layer protocol.

POP3 - Post Office Protocol version 3 is used by clients to access an internet mail server to get mail. It is not a transport layer protocol.

IMAP4 - Internet Mail Access Protocol version 4 is the replacement for POP3.

MIME - Multipurpose Internet Mail Extension is the protocol that defines the way files are attached to SMTP messages.

X.400 - International Telecommunication Union standard defines transfer protocols for sending mail between mail servers.

MHS - Message Handling Service by Novell is used for mail on Netware networks.

# FTP

Commonly used application layer protocol

Allows for the transfer of files between clients/servers.

Requires 2 connections to the server

1) Commands – uses TCP port 21
2) Actual data – uses TCP port 20

# DHCP

Dynamic Host Configuration Protocol – enables devices to obtain IP addresses, subnet masks, gateways, DNS server information, etc. from a DHCP server.

An IP address that is not being used is assigned from a range of available addresses

Not permanently assigned – only leased for a specific period of time (usually 24 hours – 7 days)

If the host logs off or the power is lost, the IP address they were using is returned to the pool to be re-assigned to another host when needed.

This is how you are able to use Wi-Fi at various places in the world!

Don’t use DHCP for devices such as servers, printers, routers, switches, etc. These should be statically assigned.

This will be covered in greater detail in CCNA 4.

# Telnet

Developed in the early 1970's – among the oldest of the application layer protocols and services in the TCP/IP protocol suite.

Allows users to emulate text-based terminal devices over the network using software.

A connection is known as a 'virtual terminal (vty)' session.

Can be run from the command prompt on a PC.

You can use the device as if you were sitting there with all the rights and priorities that you username will offer you.

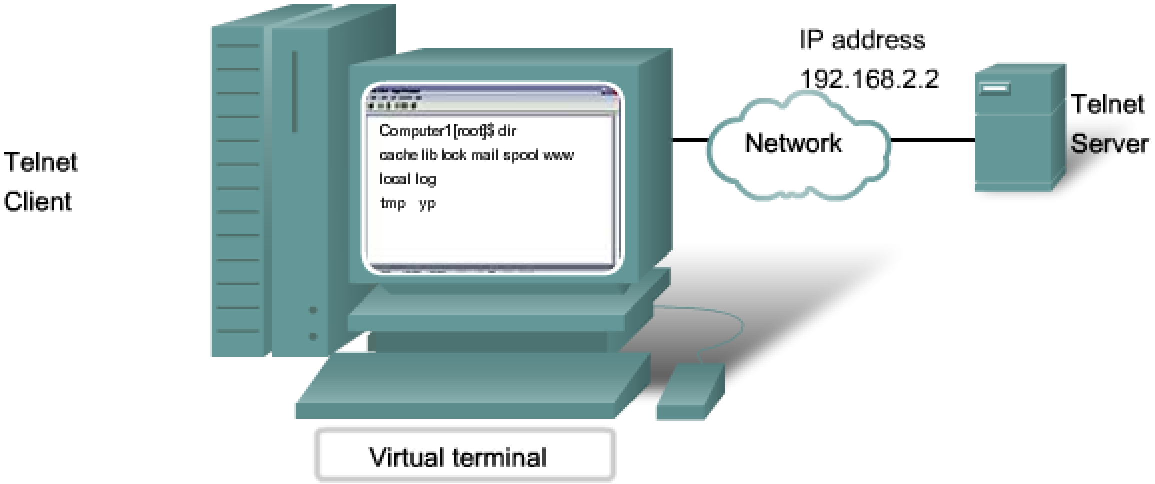
Disadvantages: Doesn't support encryption like SSH.  All data is transferred as plain text.  It can be easily intercepted and understood.

If security is a concern, you should use Secure Shell (SSH) protocol.  Provides for remote logins with stronger authentication than telnet.

Network Professionals should always use SSH whenever possible.

# Telnet

Telnet



```
Computer1 [root]$ dir
cache lib lock mail spool www
local log
tmp   yp
```

Telnet provides a way to use a computer, connected via the network, to access a network device as if the keyboard and monitor were directly connected to the device.

# FTP

## FILE TRANSFER PROTOCOL

# File Transfer Protocol (FTP)

File Transfer Protocol (FTP) is a standard Internet protocol for transmitting files between computers on the Internet over TCP/IP connections.

FTP is a client-server protocol that relies on two communications channels between client and server: a command channel for controlling the conversation and a data channel for transmitting file content. Clients initiate conversations with servers by requesting to download a file. Using FTP, a client can upload, download, delete, rename, move and copy files on a server. A user typically needs to log on to the FTP server, although some servers make some or all of their content available without login, also known as anonymous FTP.

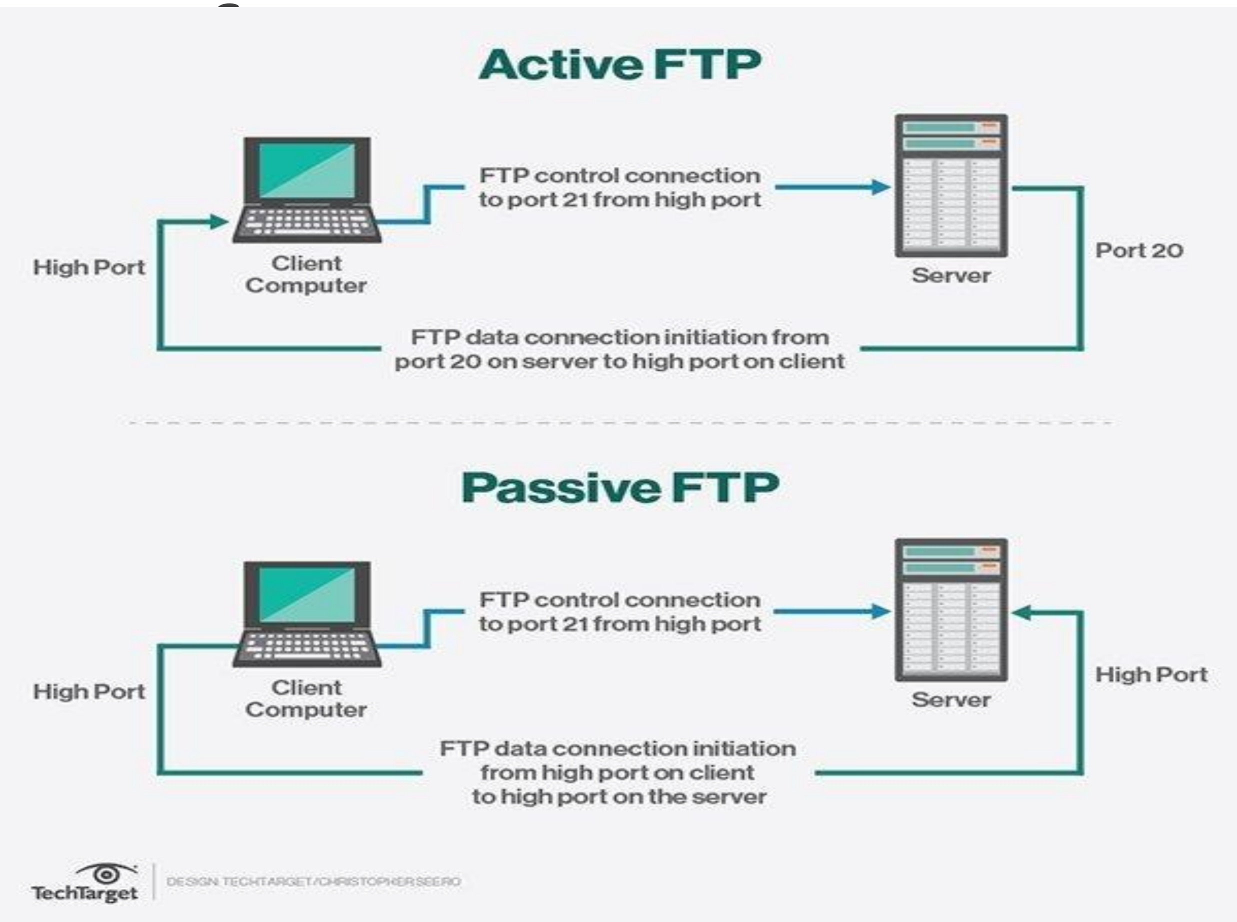
# FTP Sessions

FTP sessions work in passive or active modes. In active mode, after a client initiates a session via a command channel request, the server initiates a data connection back to the client and begins transferring data.

In passive mode, the server instead uses the command channel to send the client the information it needs to open a data channel.

Because passive mode has the client initiating all connections, it works well across firewalls and Network Address Translation (NAT) gateways.

# Typ


Active FTP
FTP control connection to port 21 from high port
High Port
Client Computer
Server
Port 20
FTP data connection initiation from port 20 on server to high port on client
Passive FTP
FTP control connection to port 21 from high port
High Port
Client Computer
Server
High Port
FTP data connection initiation from high port on client to high port on the server
TechTarget

DESIGN: TECHTARGET/CHRISTOPHERSEERO

Users can work with FTP via a simple command line interface (for example, from a console or terminal window in Microsoft Windows, Apple OS X or Linux ) or with a dedicated graphical user interface (GUI). Web browsers can also serve as FTP clients.

Although a lot of file transfer is now handled using HTTP, FTP is still commonly used to transfer files "behind the scenes" for other applications -- e.g., hidden behind the user interfaces of banking, a service that helps build a website, such as Wix or SquareSpace, or other services. It is also used, via Web browsers, to download new applications.

# TCP Port No

TCP/IP Well Known Port Numbers (0 to 1023)

| Port # | Portocol | Description |
|---|---|---|
| 20 | TCP | FTP - data port (FTP-d) |
| 21 | TCP | FTP - control (command) port (FTP-c) |
| 22 | TCP, UDP | SSH (Secure Shell) - used for secure logins, file transfers (scp, sftp) and port forwarding |
| 23 | TCP, UDP | Telnet protocol - unencrypted text communications |

# Telnet

**Telnet** is a user command and an underlying TCP/IP protocol for accessing remote computers. Through **Telnet**, an administrator or another user can access someone else's computer remotely.

Typically, this protocol is used to establish a connection to Transmission Control Protocol (TCP) port number 23, where a **Telnet** server application (telnetd) is listening. **Telnet**, however, predates TCP/IP and was originally run over Network Control Program (NCP) protocols.

# To establish a Telnet session, follow these steps:

Use your Dial-up Networking connection to connect to the Internet through your ISP.

Click the Start button, point to Programs, and then click Command Prompt.

In the MS-DOS window, type the following: telnet <POP server name or IP address> 110.

Press the ENTER key.

# Telnet

**Telnet** is a protocol that enables you to connect to remote computers and local computers over a TCP/IP network, over TCP port 23.

By default, **Telnet** is disabled in recent **Windows** environments.

To enable **Telnet command** line utilities: Click Start > Control Panel.

# Telnet and SSH

**SSH** is a network protocol used to remotely access and manage a device.

The key **difference between Telnet** and **SSH** is that **SSH** uses encryption, which means that all data transmitted over a network is secure from eavesdropping.

Like **Telnet**, a user accessing a remote device must have an **SSH** client installed.

# Control Functions

TELNET includes support for a series of control functions commonly supported by servers.

This provides a uniform mechanism for communication of (the supported) control functions.

# Control Functions

Interrupt Process (IP)

- suspend/abort process.

Abort Output (AO)

- send no more output to user’s terminal.

Are You There (AYT)

- check to see if system is still running.

Erase Character (EC)

- delete last character sent

Erase Line (EL)

- delete all input in current line.

# Introduction to Cryptography

# What is Cryptography

Cryptography

- In a narrow sense
  - Mangling information into apparent unintelligibility
  - Allowing a secret method of un-mangling
- In a broader sense
  - Mathematical techniques related to information security
  - About secure communication in the presence of adversaries

Cryptanalysis

- The study of methods for obtaining the meaning of encrypted information without accessing the secret information

Cryptology

- Cryptography + cryptanalysis

# Threats & Attack

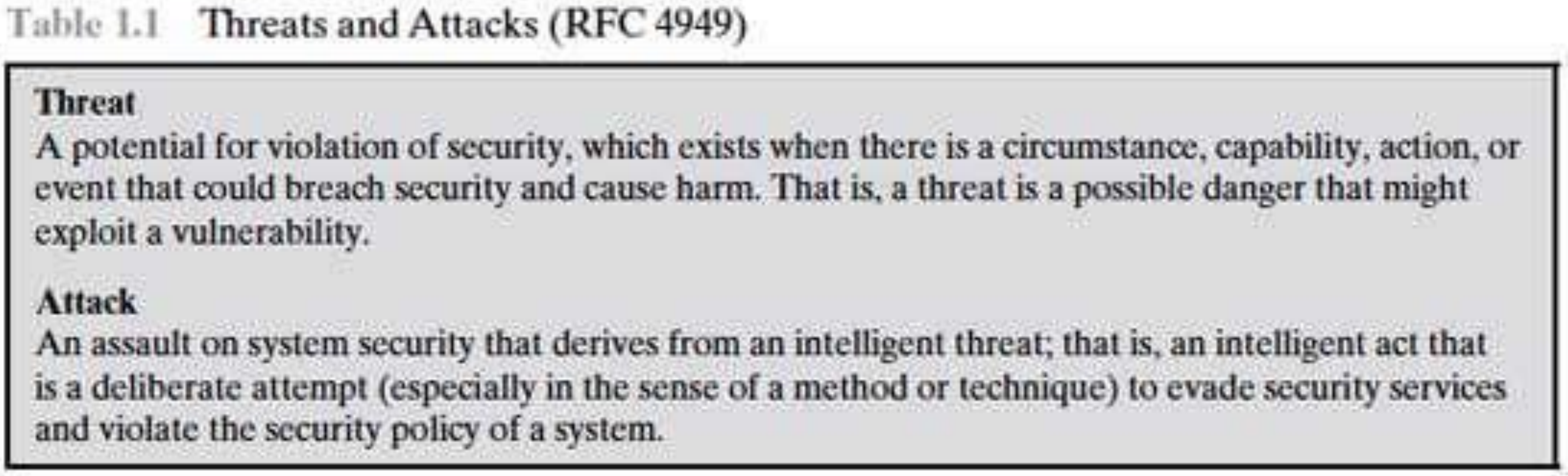


Table 1.1 Threats and Attacks (RFC 4949)

**Threat**
A potential for violation of security, which exists when there is a circumstance, capability, action, or event that could breach security and cause harm. That is, a threat is a possible danger that might exploit a vulnerability.

**Attack**
An assault on system security that derives from an intelligent threat; that is, an intelligent act that is a deliberate attempt (especially in the sense of a method or technique) to evade security services and violate the security policy of a system.

# Security Attacks

**A passive attack attempts to learn or make use of information from the system but does not affect system resources.**

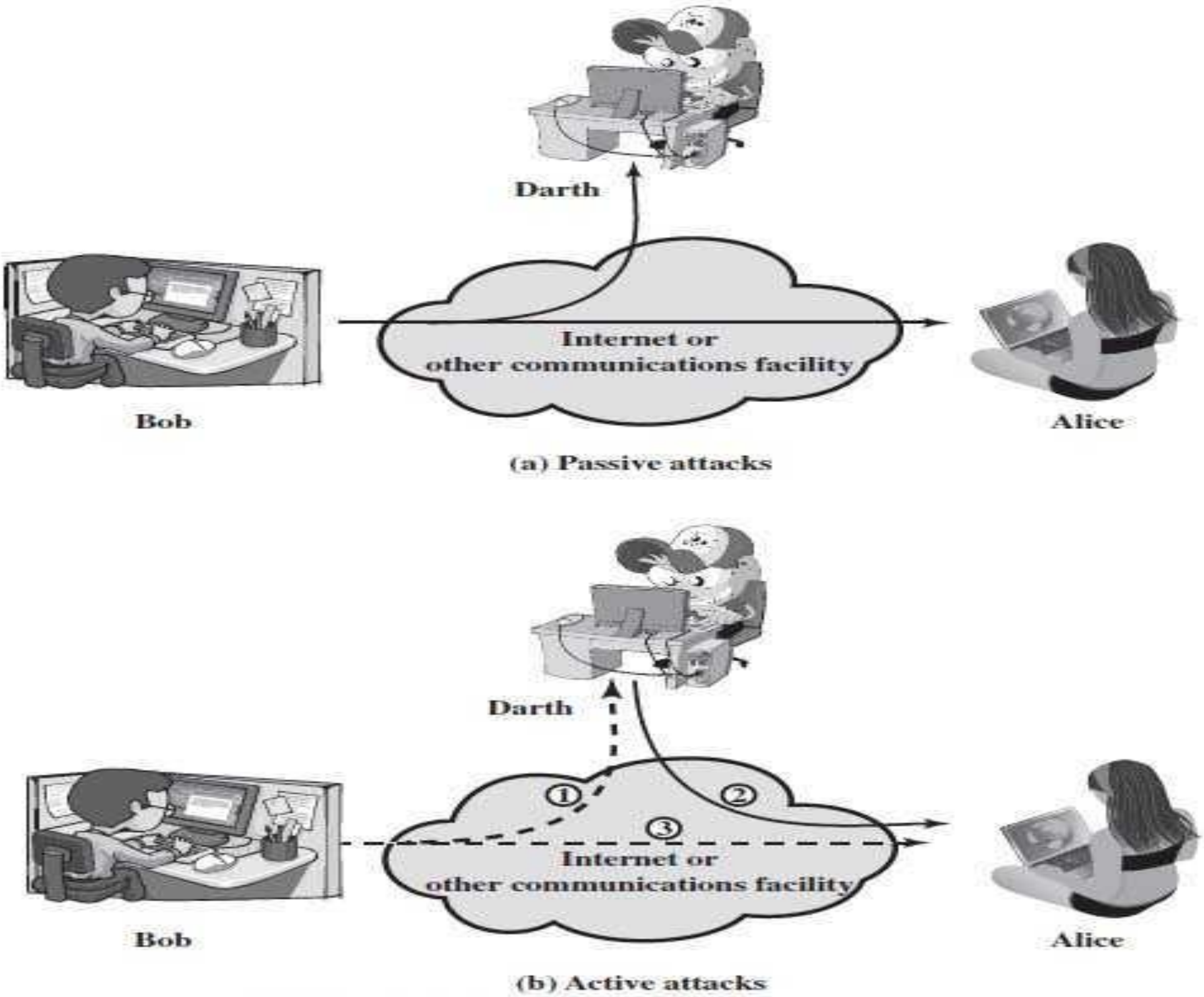
Passive attacks

- Obtain message contents
- Monitoring traffic flows

**An active attack attempts to alter system resources or affect their operation.**

Active attacks

- Masquerade of one entity as some other
- Replay previous messages
- Modify messages in transmit
- Add, delete messages
- Denial of service



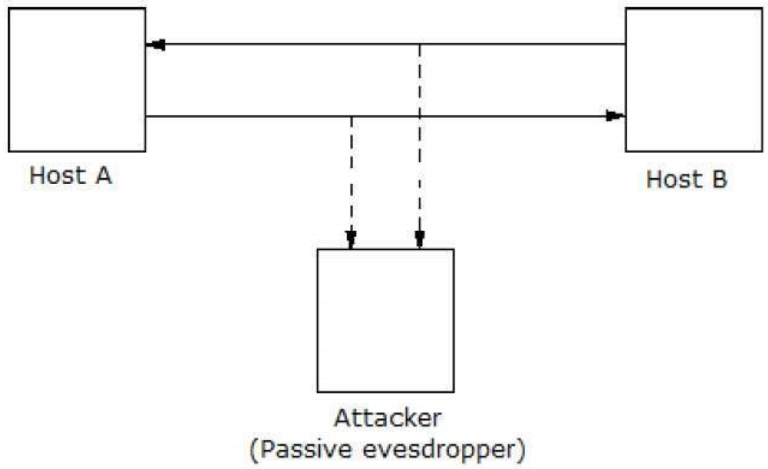
(a) Passive attacks

(b) Active attacks

Figure 1.1 Security Attacks

# Active Attacks

An active attack attempts to alter system resources or affect their operation.



# Active Attacks

Active attacks involve some modification of the data stream or the creation of a false stream and can be subdivided into four categories:

Masquerade

Replay

Modification of messages

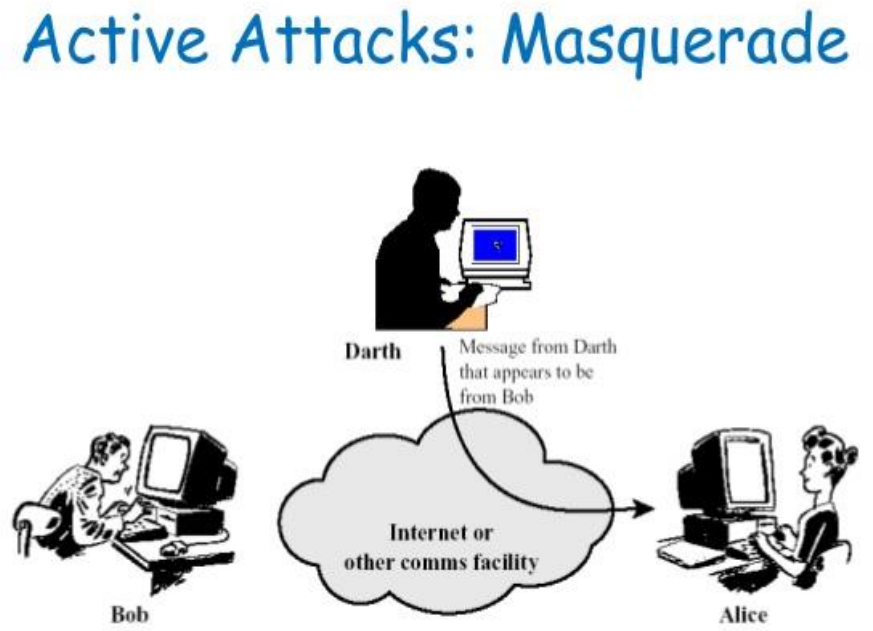
Denial of service.

# Active Attacks

A masquerade  takes place when one entity pretends to be a different entity (path 2 of Figure 1.1b is active).

~~A masquerade attack usually includes one of the other forms of active attack.~~

**For example,** authentication sequences can be captured and replayed after a valid authentication sequence has taken place, thus enabling an authorized entity with few privileges to obtain extra privileges by impersonating an entity that has those privileges.

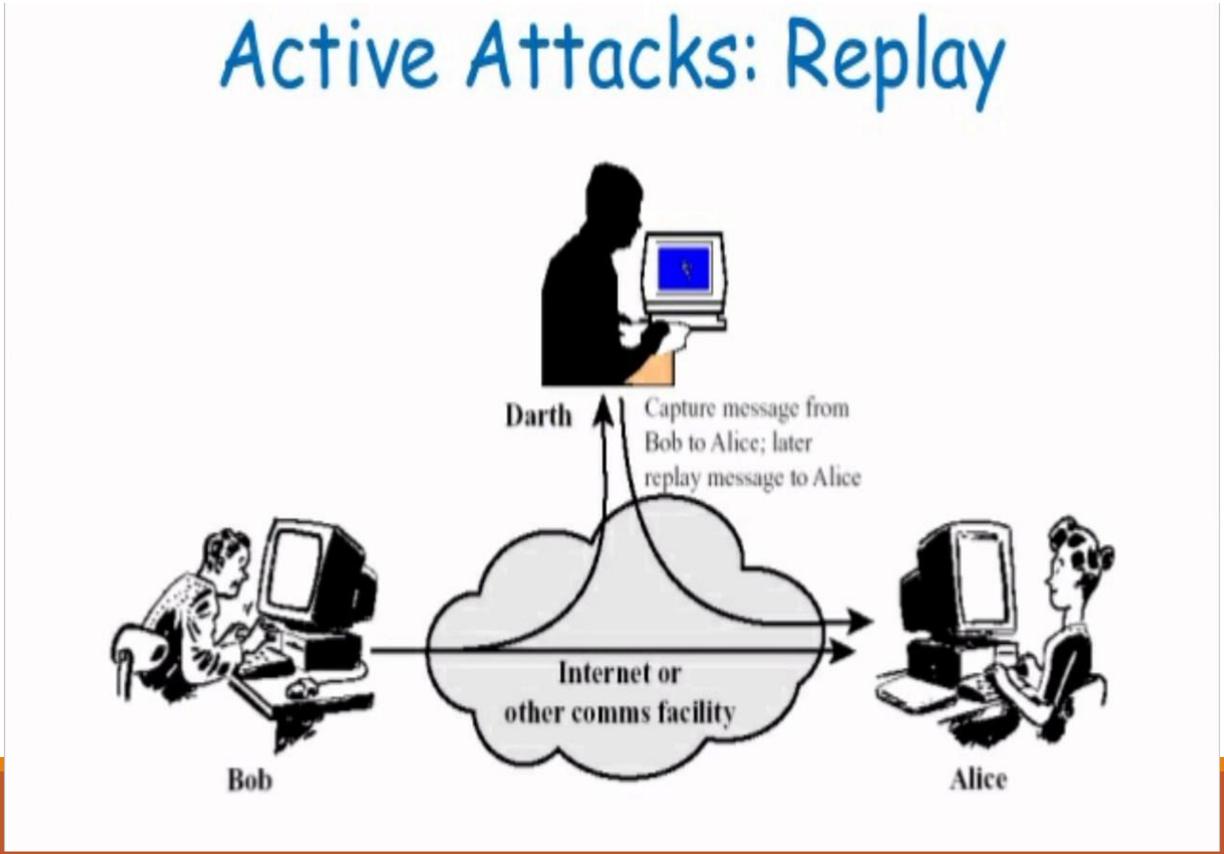


Active Attacks: Masquerade

Darth
Message from Darth that appears to be from Bob
Internet or other comms facility
Bob
Alice

# Active Attacks

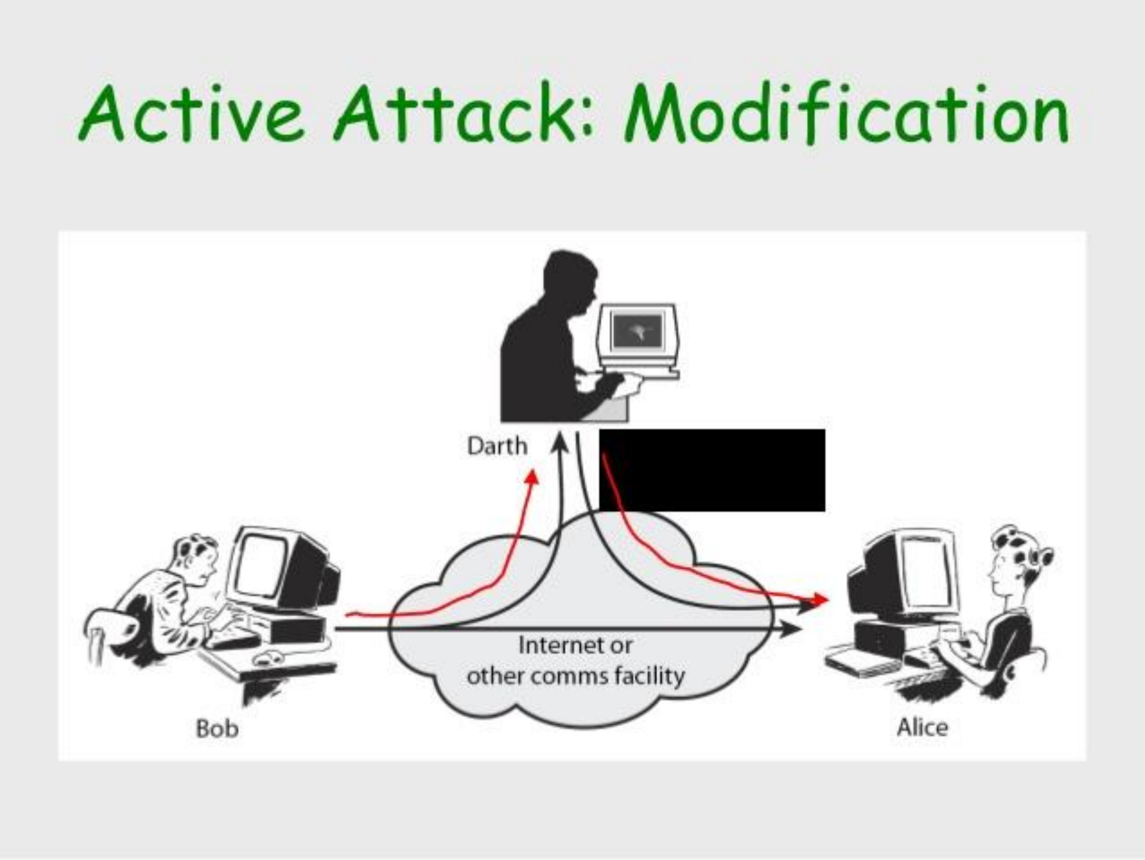
**Replay** involves the passive capture of a data unit and its subsequent retransmission to produce an unauthorized effect (paths 1, 2, and 3 active).


Active Attacks: Replay
Darth
Capture message from Bob to Alice; later replay message to Alice
Internet or other comms facility
Bob
Alice

# Active Attacks

Modification of messages simply means that some portion of a legitimate message is altered, or that messages are delayed or reordered, to produce an unauthorized effect (paths 1 and 2 active).

**For example,** a message meaning “Allow John Smith to read confidential file accounts ” is modified to mean “Allow Fred Brown to read confidential file accounts.”



Active Attack: Modification

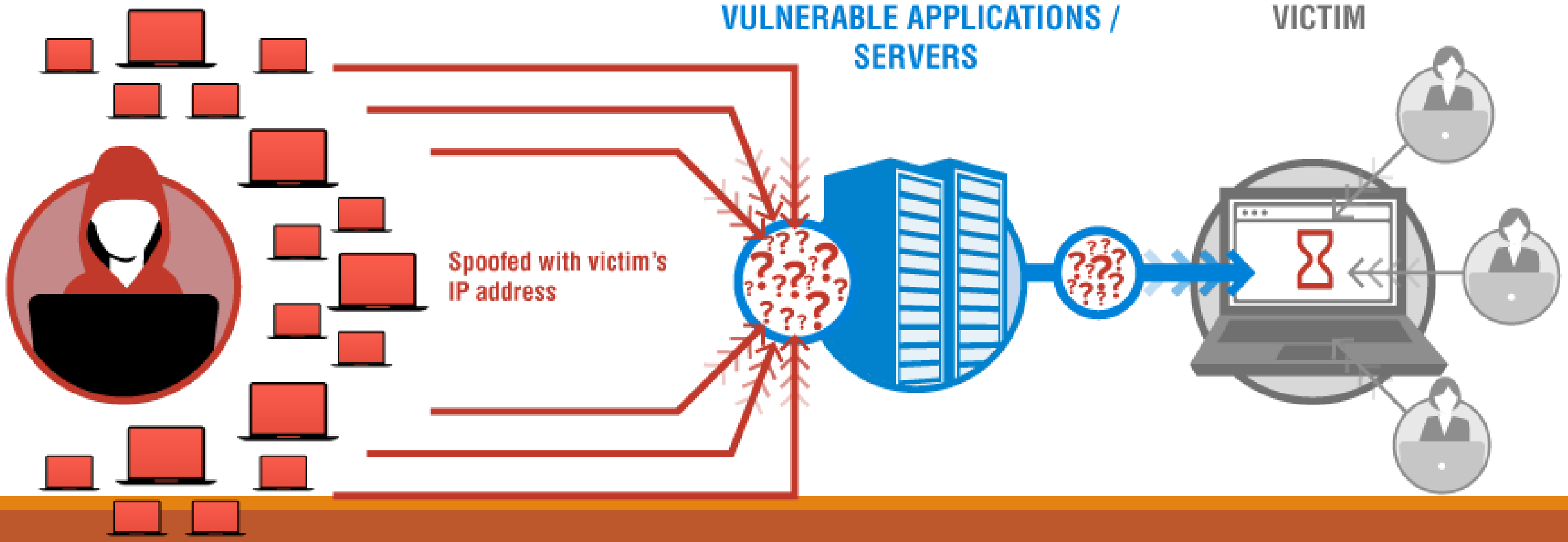
[redacted]

# Active Attacks

The denial of service  prevents or inhibits the normal use or management of communications facilities (path 3 active).

This attack may have a specific target; for example, an entity may suppress all messages directed to a particular destination (e.g., the security audit service).

Another form of service denial is the disruption of an entire network, either by disabling the network or by overloading it with messages so as to degrade performance.


VULNERABLE APPLICATIONS / SERVERS
VICTIM
Spoofed with victim's IP address

# Active Attacks

Active attacks present the opposite characteristics of passive attacks. Whereas passive attacks are difficult to detect, measures are available to prevent their success.

On the other hand, it is quite difficult to prevent active attacks absolutely because of the wide variety of potential physical, software, and network vulnerabilities.

Instead, the goal is to detect active attacks and to recover from any disruption or delays caused by them. If the detection has a deterrent effect, it may also contribute to prevention.

# Passive Attacks

Passive attacks are in the nature of eavesdropping on, or monitoring of, transmissions.

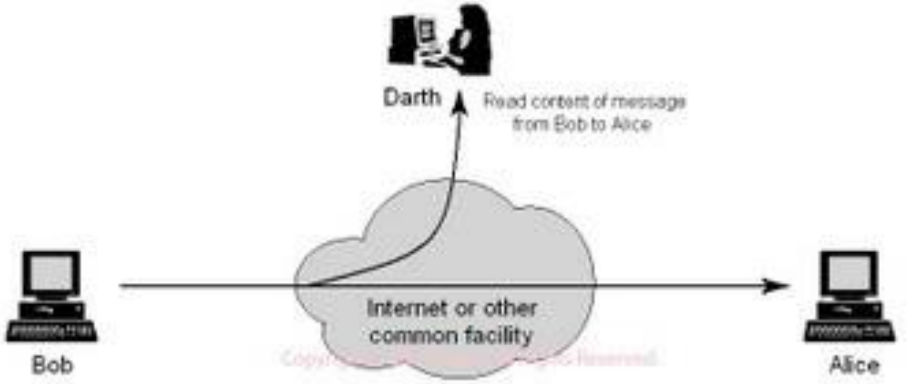
The goal of the opponent is to obtain information that is being transmitted.

Two types of passive attacks are the release of message contents and traffic analysis.

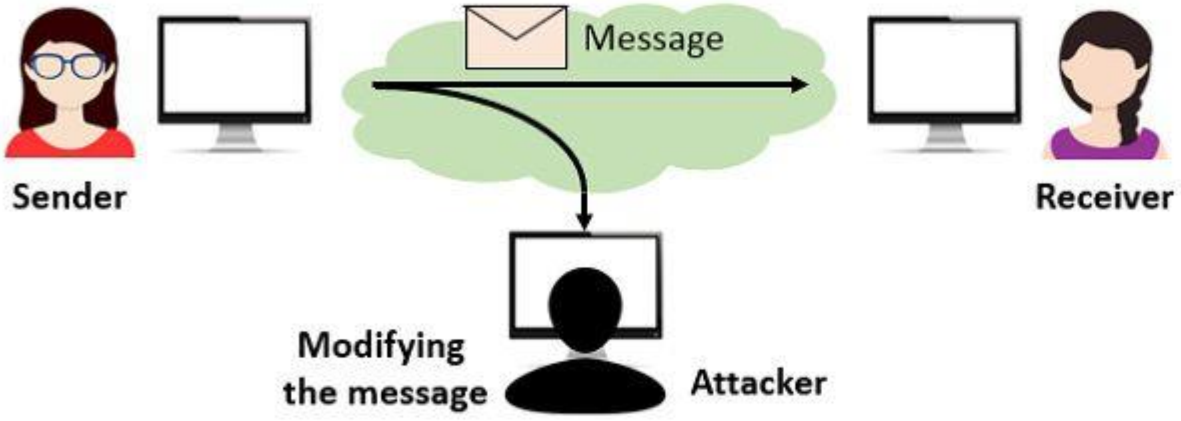
The release of message content is easily understood.

A telephone conversation, an electronic mail message, and a transferred file may contain sensitive or confidential information.

We would like to prevent an opponent from learning the contents of these transmissions.


Darth
Read content of message from Bob to Alice
Internet or other common facility
Bob
Alice

Release of message contents (Passive Attacks)



# Passive Attack

# Passive Attacks

A second type of passive attack, traffic analysis, is subtler. Suppose that we had a way of masking the contents of messages or other information traffic so that opponents, even if they captured the message, could not extract the information from the message.

The common technique for masking contents is encryption.

If we had encryption protection in place, an opponent might still be able to observe the pattern of these messages.

The opponent could determine the location and identity of communicating hosts and could observe the frequency and length of messages being exchanged.

This information might be useful in guessing the nature of the communication that was taking place.

# Passive Attacks

**Passive attacks** are very difficult to detect, because they do not involve any alteration of the data.

Typically, the message traffic is sent and received in an apparently normal fashion, and neither the sender nor receiver is aware that a third party has read the messages or observed the traffic pattern.

However, it is feasible to prevent the success of these attacks, usually by means of encryption. Thus, the emphasis in dealing with passive attacks is on prevention rather than detection.

# Objectives of Information Security

Confidentiality (secrecy)

- ~~Only the sender and intended receiver should be able to understand~~ the contents of the transmitted message

Authentication

- Both the sender and receiver need to confirm the identity of other party involved in the communication

Data integrity

- The content of their communication is not altered, either maliciously or by accident, in transmission.

Availability

- Timely accessibility of data to authorized entities.

# Objectives of Information Security

Non-repudiation

- An entity is prevented from denying its previous commitments or actions

Access control

- An entity cannot access any entity that it is not authorized to.

Anonymity

- The identity of an entity if protected from others.

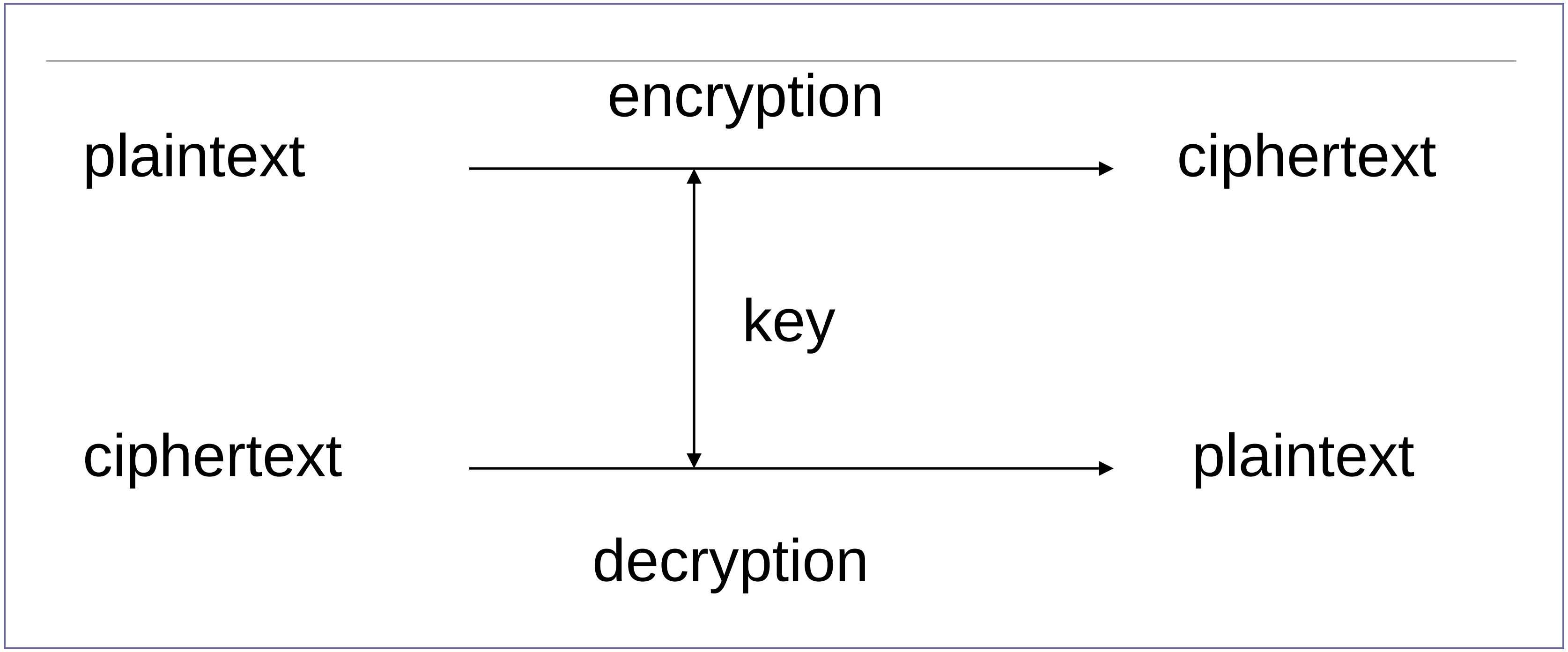
# Types of Cryptographic Functions

Secret key functions

Public key functions

Hash functions

# Secret Key Cryptography



Using a single key for encryption/decryption.

The plaintext and the ciphertext having the same size.

Also called *symmetric* key cryptography

# SKC: Security Uses

Transmitting over an insecure channel

- The transmitted message is encrypted by the sender and can be decrypted by the receiver, with the same key
- Prevent attackers from eavesdropping

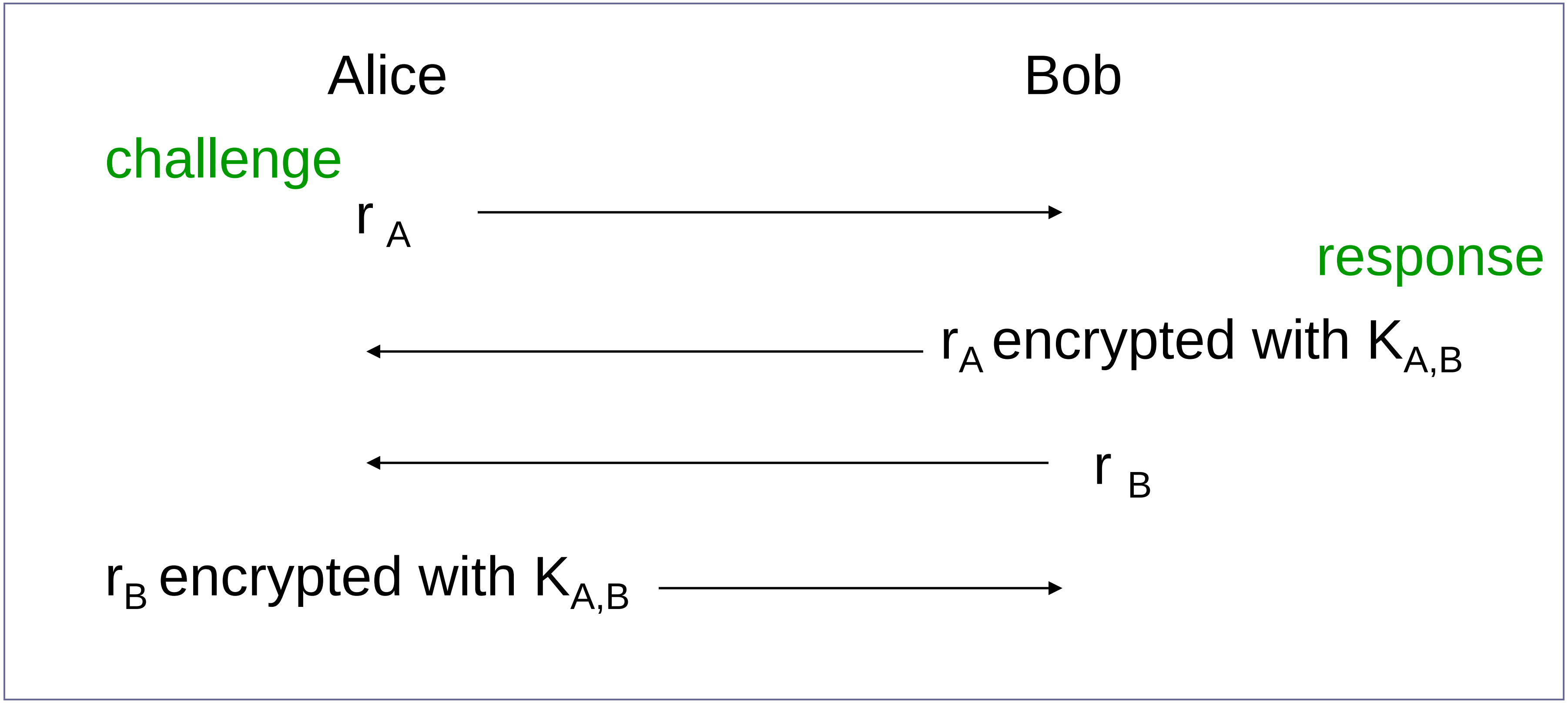
Secure storage on insecure media

- Data is encrypted before being stored somewhere
- Only the entities knowing the key can decrypt it

# SKC: Security Uses

## Authentication

- Strong authentication: proving knowledge of a secret without revealing it.



Alice

Bob

challenge

$r_A$

response

$r_A$ encrypted with $K_{A,B}$

$r_B$

$r_B$ encrypted with $K_{A,B}$

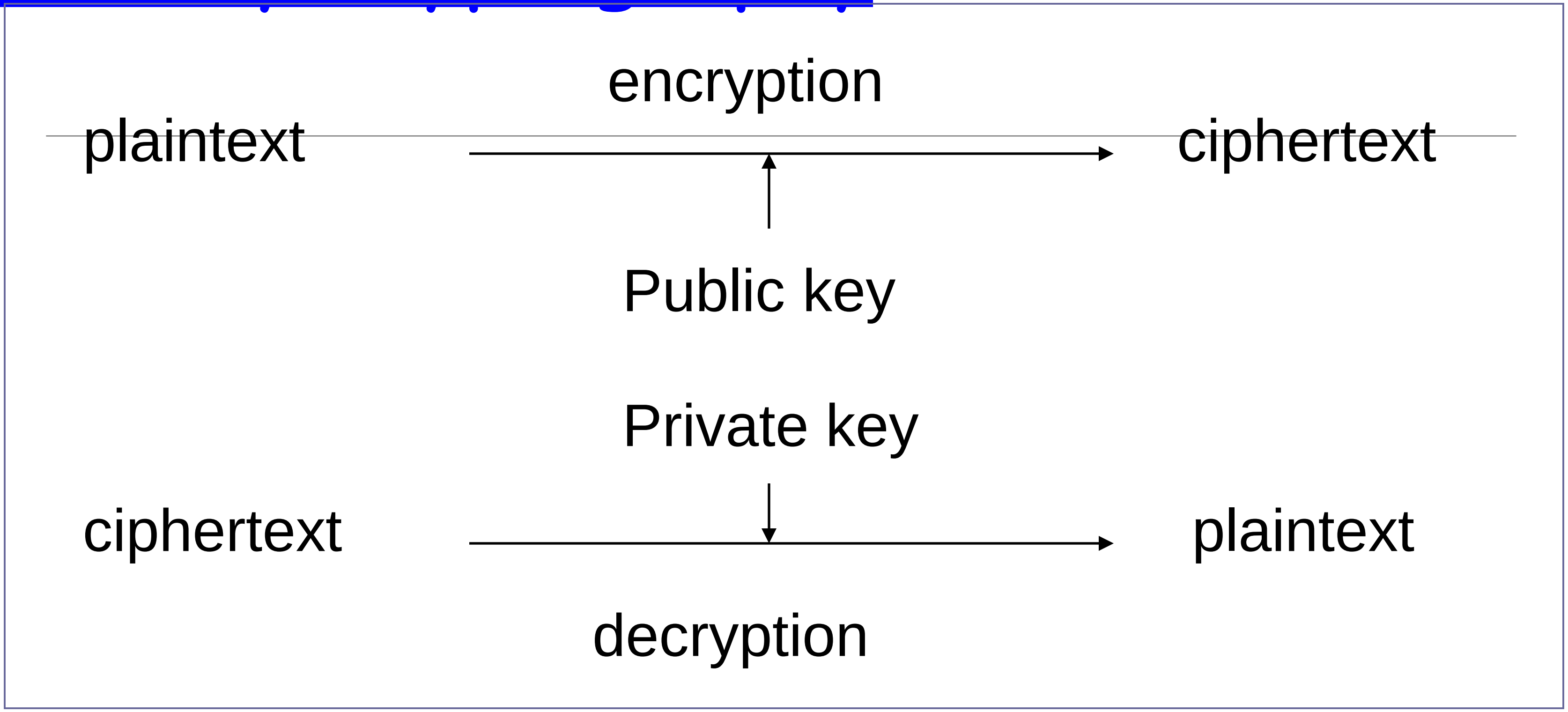
# SKC: Security Uses

Integrity Check

- Noncryptographic checksum
  - Using a well-known algorithm to map a message (of arbitrary length) to a fixed-length checksum
  - Protecting against accidental corruption of a message
  - Example: CRC

- Cryptographic checksum
  - A well-know algorithm
  - Given a key and a message
  - The algorithm produces a fixed-length message authentication code (MAC) that is sent with the message

# Public Key Cryptography


encryption

plaintext

ciphertext

Public key

Private key

ciphertext

plaintext

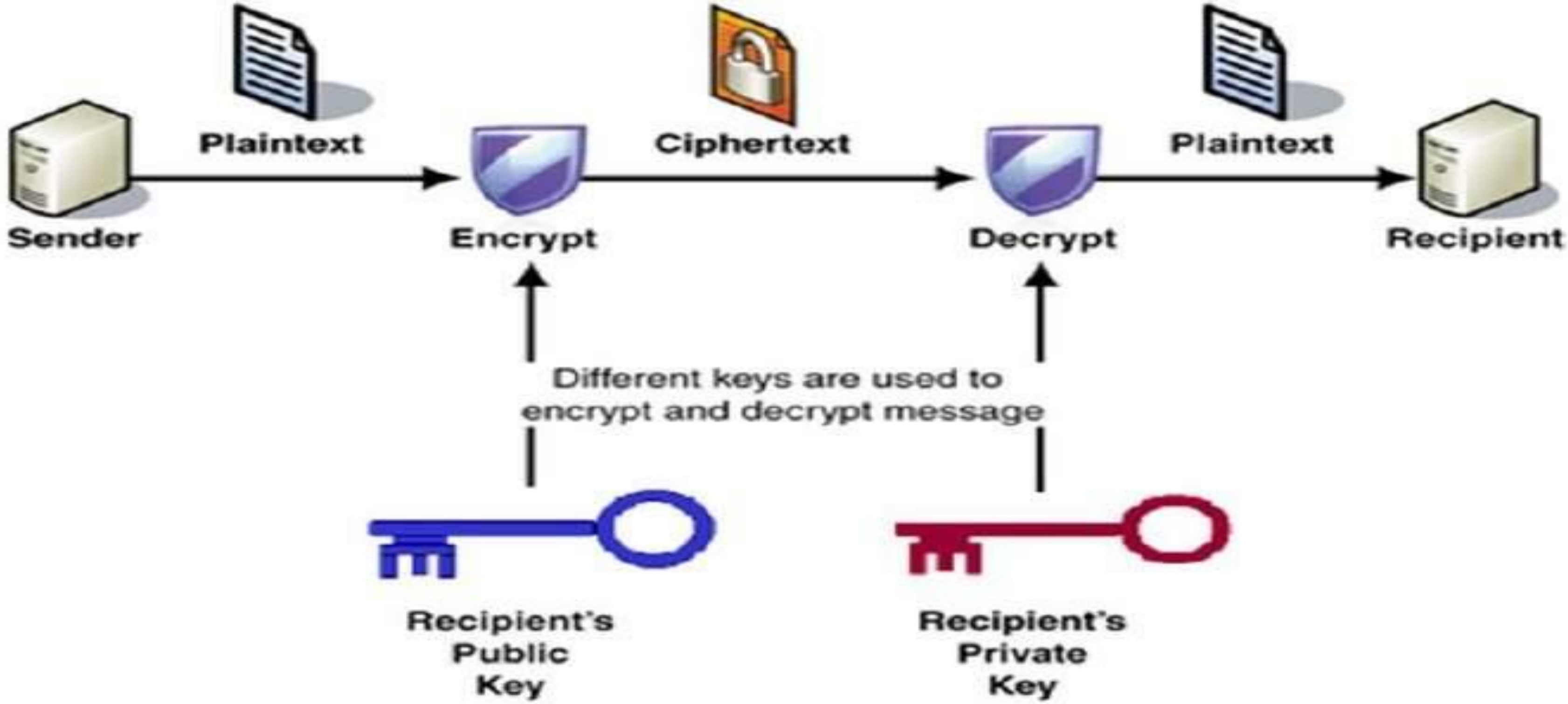
decryption

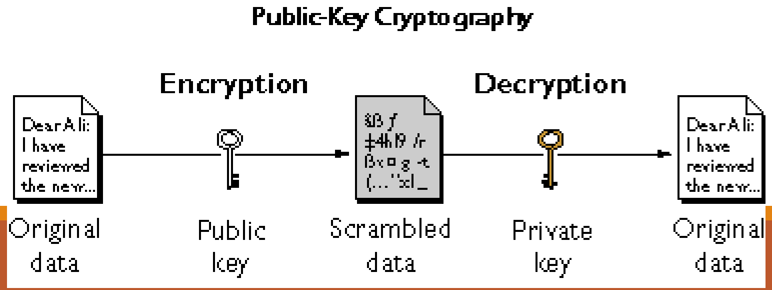
Each individual has two keys

- a private key (d): need not be reveal to anyone
- a public key (e): preferably known to the entire world

Public key crypto is also called asymmetric crypto.

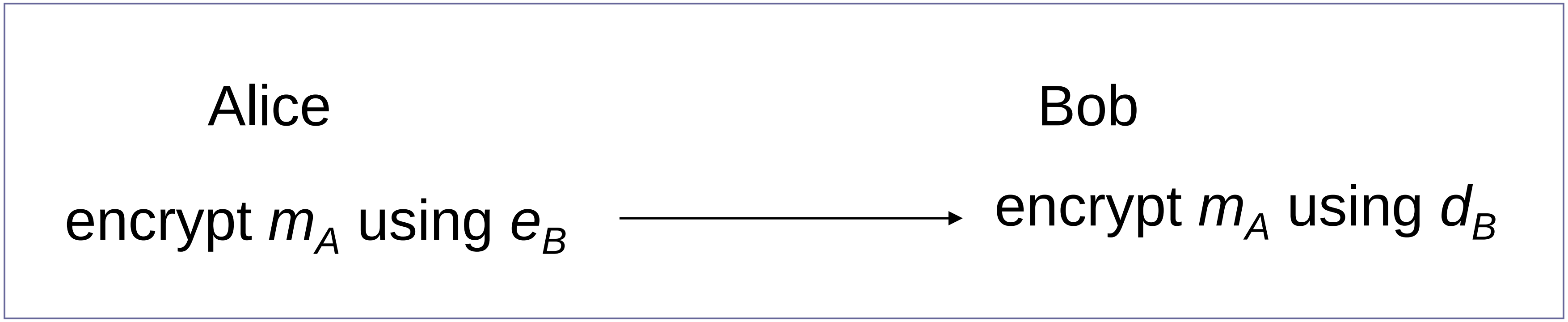
# Public Key


Plaintext
Ciphertext
Plaintext
Sender
Encrypt
Decrypt
Recipient
Different keys are used to encrypt and decrypt message
Recipient's Public Key
Recipient's Private Key


Public-Key Cryptography
Encryption
Decryption
Dear Ali: I have reviewed the new...
Original data
Public key
Scrambled data
Private key
Dear Ali: I have reviewed the new...
Original data

# PKC: Security Uses

- Transmitting over an insecure channel



Alice

Bob

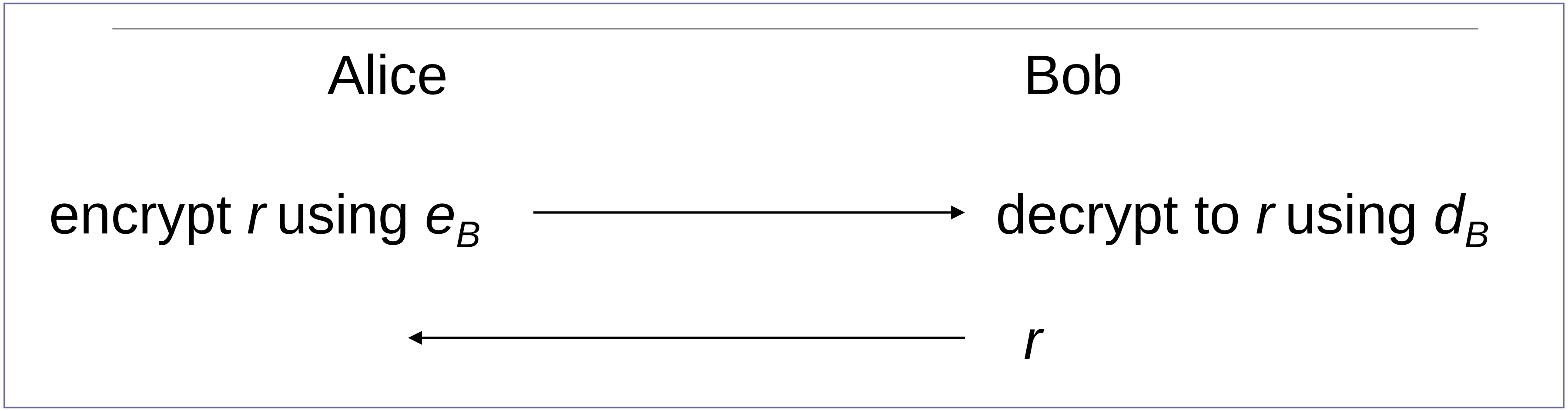
encrypt $m_A$ using $e_B$ $\longrightarrow$ encrypt $m_A$ using $d_B$

Secure storage on insecure media

- Data is encrypted with the public key of the source, before being stored somewhere
- Nobody else can decrypt it (not knowing the private key of the data source)

# PKC: Security Uses

## Authentication



Alice

Bob

encrypt $r$ using $e_B$

decrypt to $r$ using $d_B$

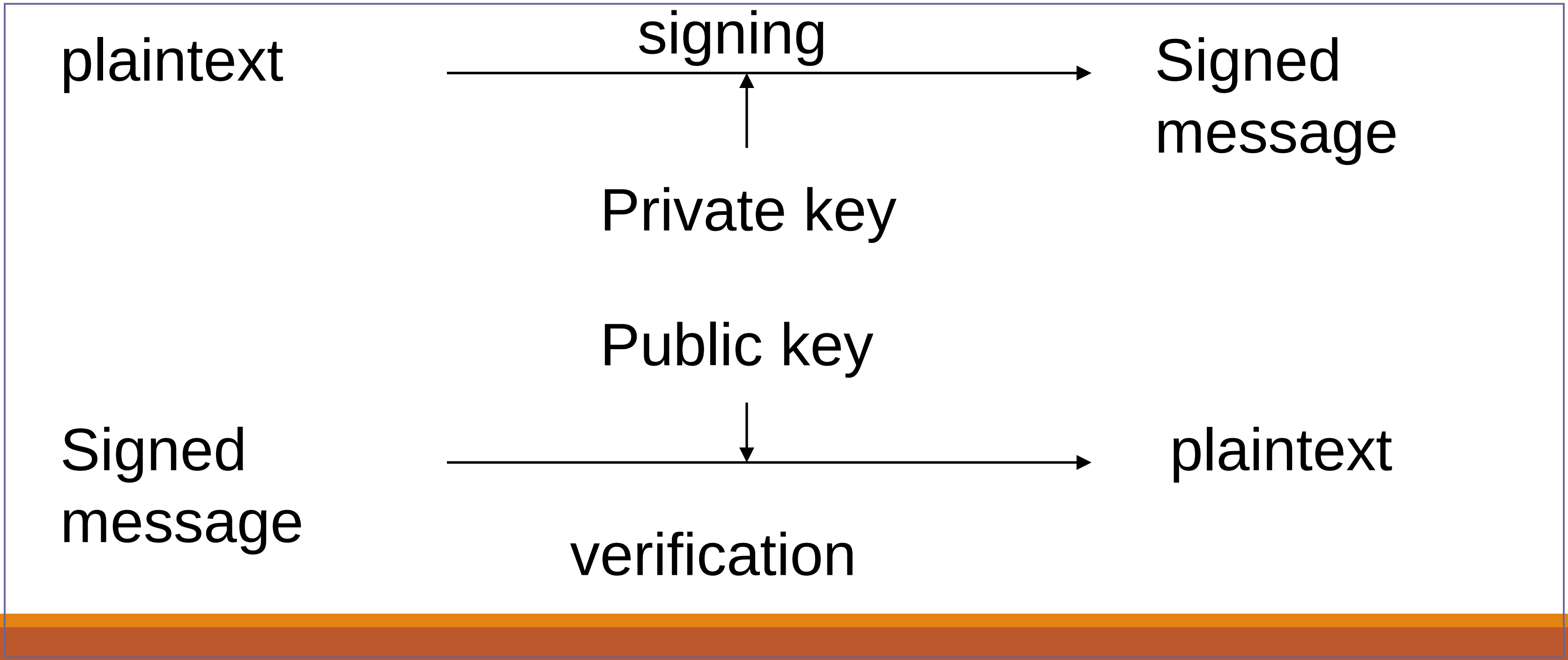
$r$

# PKC: Security Uses

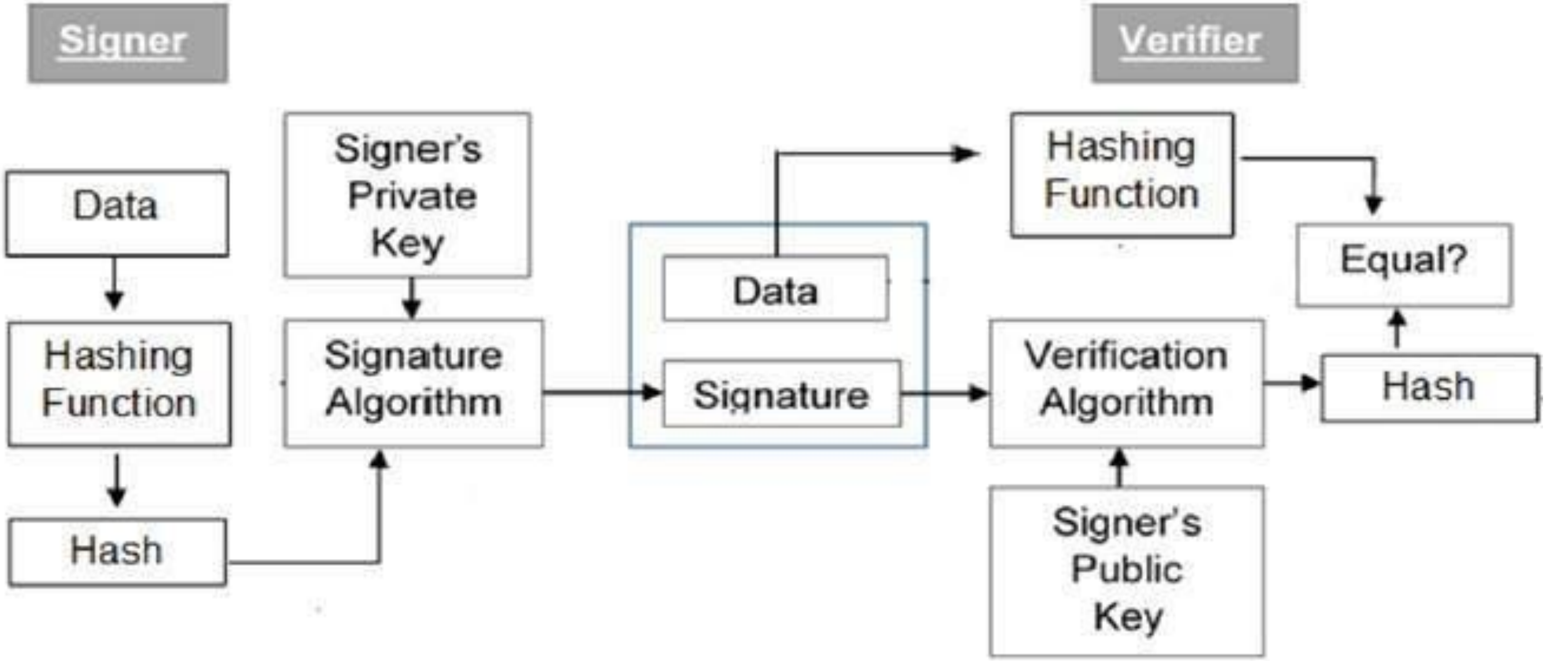
## Digital Signatures

**a digital code (generated and authenticated by public key encryption) which is attached to an electronically transmitted document to verify its contents and the sender's identity.**

- Proving that a message is generated by a particular individual
- Non-repudiation: the signing individual can not be denied, because only him/her knows the private key.



# Process


Signer
Verifier
Data
Hashing Function
Hash
Signer's Private Key
Signature Algorithm
Data
Signature
Hashing Function
Equal?
Verification Algorithm
Hash
Signer's Public Key

# Process Steps: Digital Signature

Each person adopting this scheme has a public-private key pair.

Generally, the key pairs used for encryption/decryption and signing/verifying are different. The private key used for signing is referred to as the signature key and the public key as the verification key.

Signer feeds data to the hash function and generates hash of data.

Hash value and signature key are then fed to the signature algorithm which produces the digital signature on given hash. Signature is appended to the data and then both are sent to the verifier.

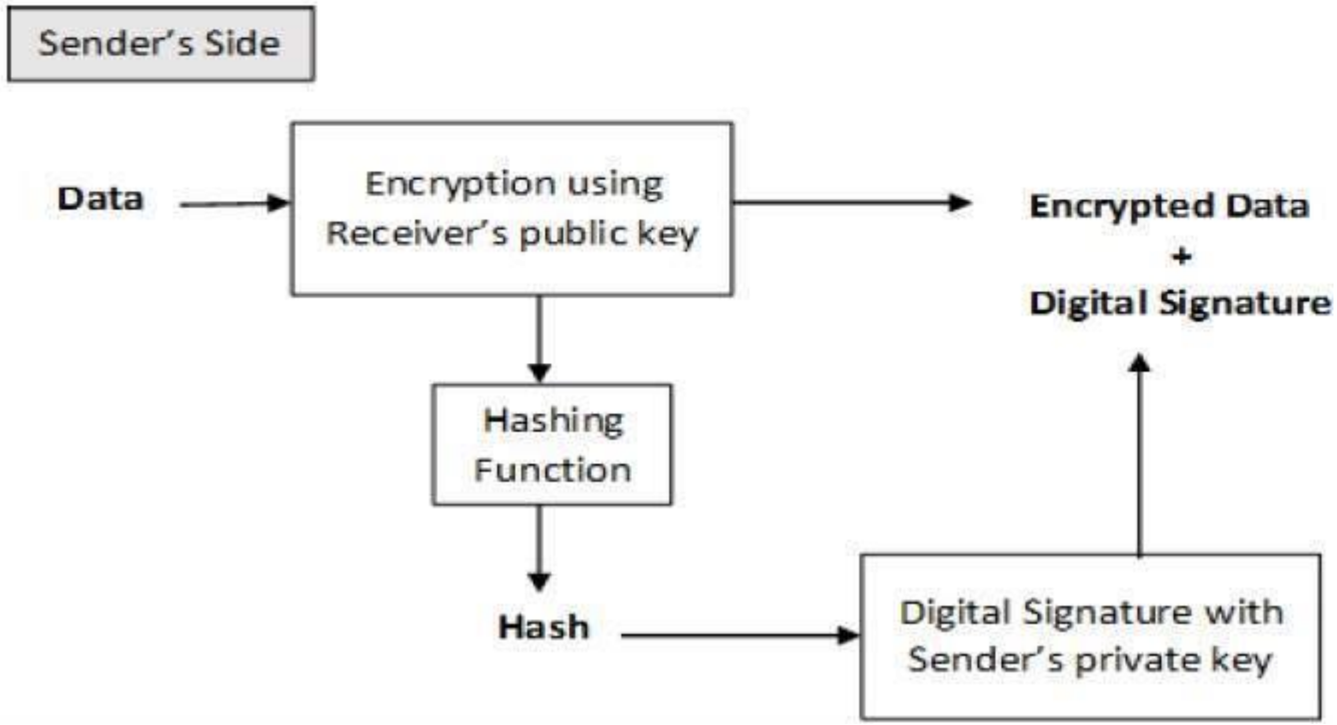
Verifier feeds the digital signature and the verification key into the verification algorithm. The verification algorithm gives some value as output.

Verifier also runs same hash function on received data to generate hash value.

For verification, this hash value and output of verification algorithm are compared. Based on the comparison result, verifier decides whether the digital signature is valid.

Since digital signature is created by 'private' key of signer and no one else can have this key; the signer cannot repudiate signing the data in future.

# Encryption Process


Sender's Side
Data
Encryption using Receiver's public key
Encrypted Data + Digital Signature
Hashing Function
Hash
Digital Signature with Sender's private key

# Importance:Digital Signature

**Message authentication** − When the verifier validates the digital signature using public key of a sender, he is assured that signature has been created only by sender who possess the corresponding secret private key and no one else.

**Data Integrity** − In case an attacker has access to the data and modifies it, the digital signature verification at receiver end fails. The hash of modified data and the output provided by the verification algorithm will not match. Hence, receiver can safely deny the message assuming that data integrity has been breached.

**Non-repudiation** − Since it is assumed that only the signer has the knowledge of the signature key, he can only create unique signature on a given data. Thus the receiver can present data and the digital signature to a third party as evidence if any dispute arises in the future.

# Hash Functions

## Cryptographic hash function

- A ~~mathematical transformation that takes a message of arbitrary~~ length and computes it a fixed-length (short) number.

## Properties

( Let the hash of a message *m* be *h(m)* )

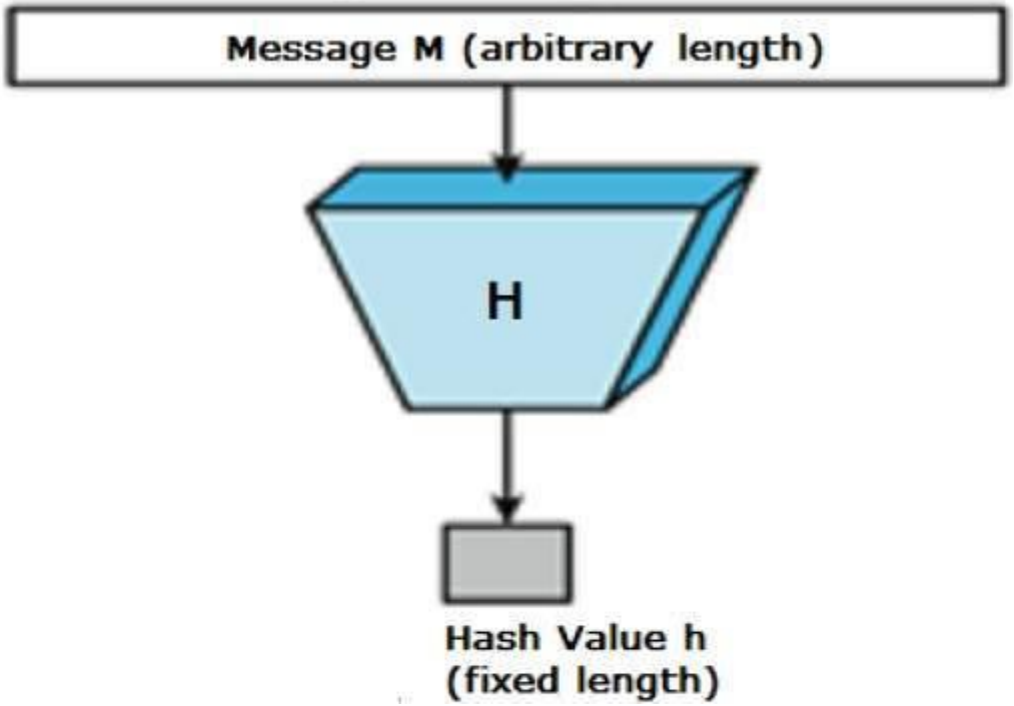
- For any *m*, it is relatively easy to compute *h(m)*
- Given *h(m)*, there is no way to find an m that hashes to *h(m)* in a way that is substantially easier than going through all possible values of m and computing *h(m)* for each one.
- It is computationally infeasible to find two values that hash to the same thing.

# Hash Functions

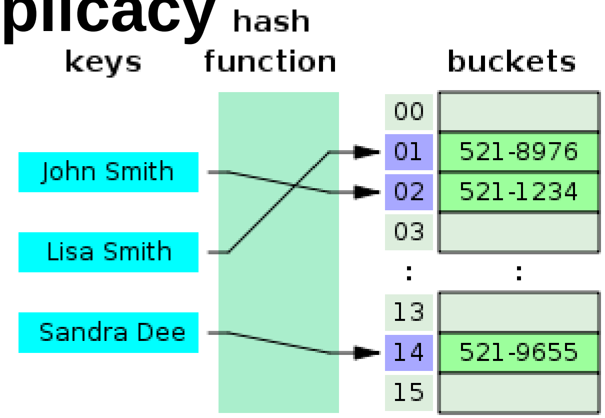
Hash functions are extremely useful and appear in almost all information security applications.

A hash function is a mathematical function that converts a numerical input value into another compressed numerical value. The input to the hash function is of arbitrary length but output is always of fixed length.

Values returned by a hash function are called **message digest** or simply **hash values**. The following picture illustrated hash function −


Message M (arbitrary length)
H
Hash Value h
(fixed length)

## For removing duplicacy



## division method

Assume a table with 8 slots:

Hash key = key % table size

4 = 36 % 8

2 = 18 % 8

0 = 72 % 8

3 = 43 % 8

6 = 6 % 8

| | |
|---|---|
| [0] | 72 |
| [1] | |
| [2] | 18 |
| [3] | 43 |
| [4] | 36 |
| [5] | |
| [6] | 6 |
| [7] | |

# Folding method

The **folding method** for constructing hash functions begins by dividing the item into equal-size pieces (the last piece may not be of equal size).

These pieces are then added together to give the resulting hash value.

For example, if our item was the phone number 436-555-4601, we would take the digits and divide them into groups of 2 (43,65,55,46,01).

After the addition, 43+65+55+46+0143+65+55+46+01, we get 210.

If we assume our hash table has 11 slots, then we need to perform the extra step of dividing by 11 and keeping the remainder. In this case 210 % 11210 % 11 is 1,

so the phone number 436-555-4601 hashes to slot 1. Some folding methods go one step further and reverse every other piece before the addition.

For the above example, we get 43+56+55+64+01=21943+56+55+64+01=219 which gives 219 % 11=10219 % 11=10.

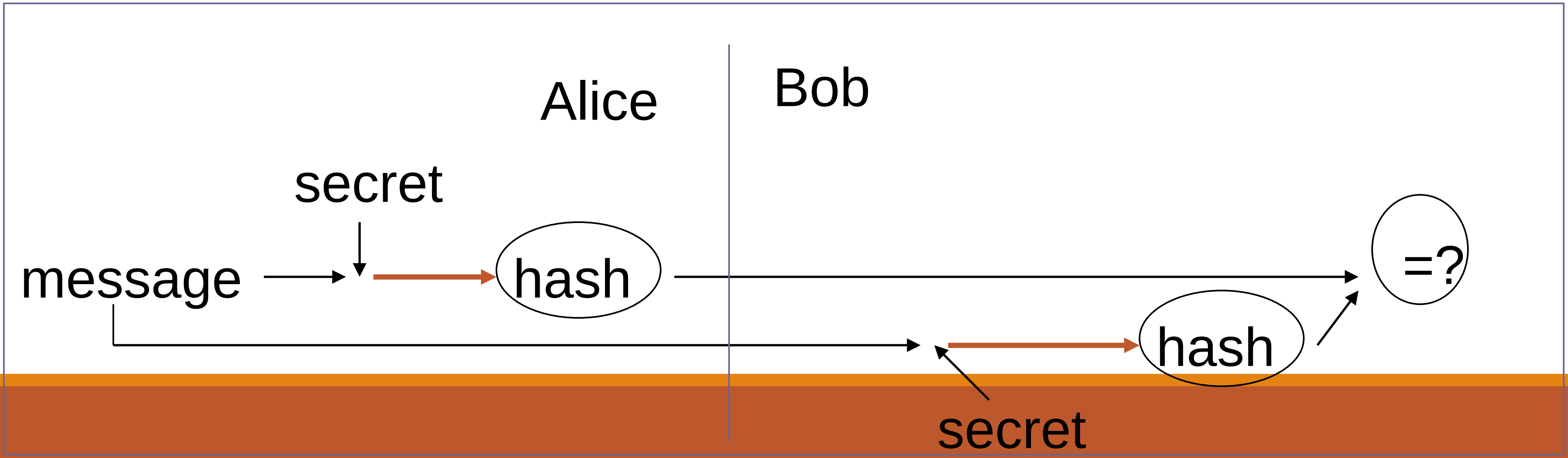
# Hash Functions: Security Uses

## Password hashing

- ~~The system store a hash of the password (not the password itself)~~
- When a password is supplied, it computes the password’s hash and compares it with the stored value.

## Message integrity

- Using cryptographic hash functions to generate a MAC



# Hash Functions: Security Uses

Message fingerprint

- ~~Save the message digest of the data on a tamper-proof backing~~ store
- Periodically re-compute the digest of the data to ensure it is not changed.

Downline load security

- Using a hash function to ensure a download program is not modified

Improving signature efficiency

- Compute a message digest (using a hash function) and sign that.

# Digital Certificates

A certificate can be considered as the ID card issued to the person. People use ID cards such as a driver's license, passport to prove their identity. A digital certificate does the same basic thing in the electronic world, but with one difference.

Digital Certificates are not only issued to people but they can be issued to computers, software packages or anything else that need to prove the identity in the electronic world.

Digital certificates are based on the ITU standard X.509 which defines a standard certificate format for public key certificates and certification validation. Hence digital certificates are sometimes also referred to as X.509 certificates.

# Digital Certificates

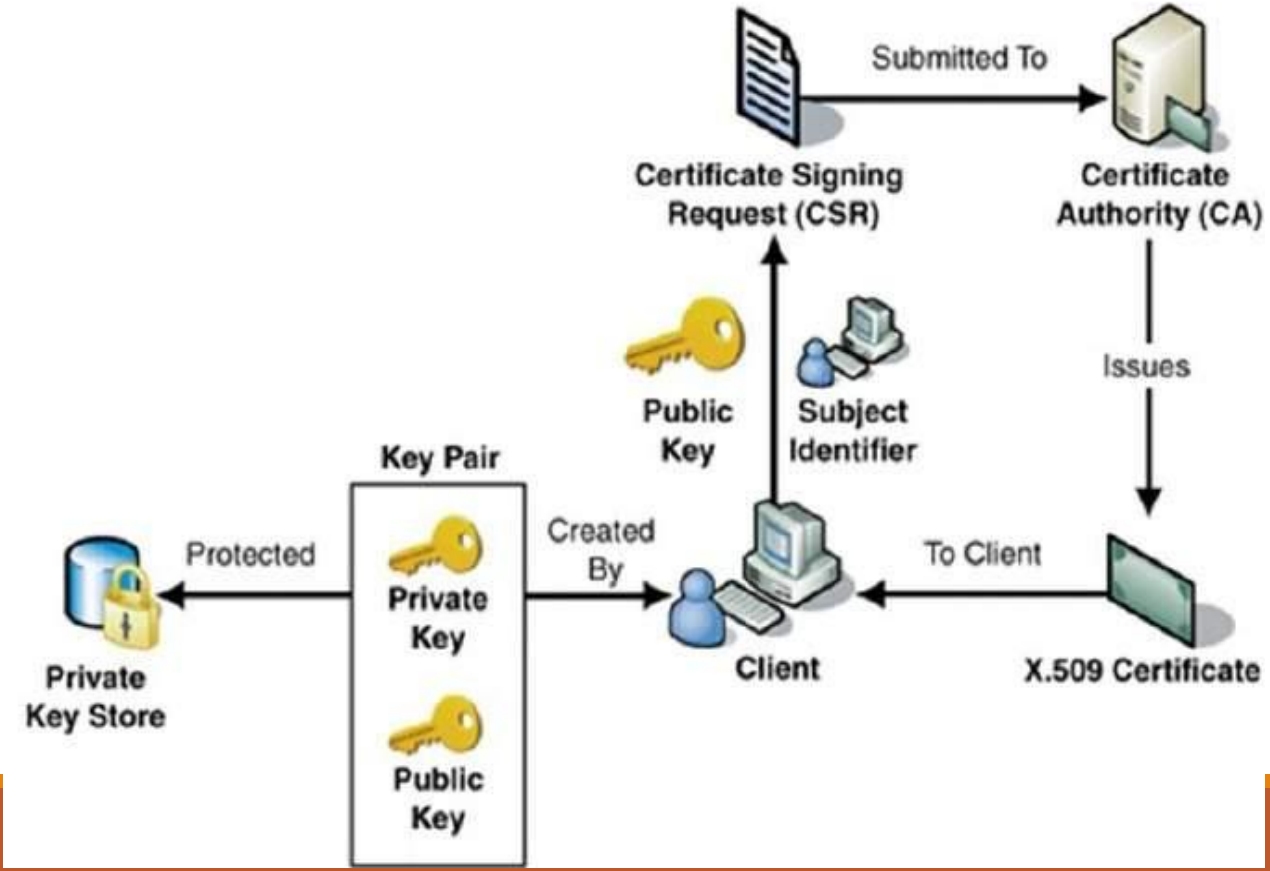
Public key pertaining to the user client is stored in digital certificates by The Certification Authority (CA) along with other relevant information such as client information, expiration date, usage, issuer etc.

CA digitally signs this entire information and includes digital signature in the certificate.

Anyone who needs the assurance about the public key and associated information of client, he carries out the signature validation process using CA’s public key. Successful validation assures that the public key given in the certificate belongs to the person whose details are given in the certificate.

## Certificate Authority

the CA issues certificate to a client and assist other users to verify the certificate. The CA takes responsibility for identifying correctly the identity of the client asking for a certificate to be issued, and ensures that the information contained within the certificate is correct and digitally signs it.



# The key functions of a CA are as follows −

**Generating key pairs** − The CA may generate a key pair independently or jointly with the client.

**Issuing digital certificates** − the CA issues a certificate after client provides the credentials to confirm his identity.

**Publishing Certificates** − The CA need to publish certificates so that users can find them.

**Verifying Certificates** − The CA makes its public key available in environment to assist verification of his signature on clients' digital certificate.

**Revocation of Certificates** − At times, CA revokes the certificate issued due to some reason such as compromise of private key by user or loss of trust in the client. After revocation, CA maintains the list of all revoked certificate that is available to the environment.

# Classes of Certificates

There are four typical classes of certificate

**Class 1** − These certificates can be easily acquired by supplying an email address.

**Class 2** − These certificates require additional personal information to be supplied.

**Class 3** − These certificates can only be purchased after checks have been made about the requestor’s identity.

**Class 4** − They may be used by governments and financial organizations needing very high levels of trust.

# SNMP

Simple **Network** Management **Protocol** (**SNMP**) is a popular **protocol** for **network** management.

It is used for collecting information from, and configuring, **network** devices, such as servers, printers, hubs, switches, and routers on an Internet **Protocol** (IP) **network.**

**SNMP** was designed to be used as a request/response protocol. The protocol details are simple (hence the name, "simple network management protocol").

And **UDP** is a very simple transport. Try implementing **TCP** on your basic agent - it's considerably more complex than a simple agent coded using **UDP.**

**SNMP** allows monitoring of network devices such as servers, workstations, printers, routers, bridges, and hubs, as well as services such as Dynamic Host Configuration Protocol (DHCP) or Windows Internet Name Service (WINS). The following sections describe the architecture, components, and processes used by **SNMP.**

# SNMP: Basic Concepts

SNMP is a protocol that is implemented on the application layer of the networking stack (click here to learn about networking layers).

The protocol was created as a way of gathering information from very different systems in a consistent manner.

Although it can be used in connection to a diverse array of systems, the method of querying information and the paths to the relevant information are standardized.

There are multiple versions of the SNMP protocol, and many networked hardware devices implement some form of SNMP access.

The most widely used version is SNMPv1, but it is in many ways insecure. Its popularity largely stems from its ubiquity and long time in the wild. Unless you have a strong reason not to, we recommend you use SNMPv3, which provides more advanced security features.

# SNMP Managers

An SNMP manager is a computer that is configured to poll SNMP agent for information. The management component, when only discussing its core functionality, is actually a lot less complex than the client configuration, because the management component simply requests data.

The manager can be any machine that can send query requests to SNMP agents with the correct credentials. Sometimes, this is implemented as part of a monitoring suite, while other times this is an administrator using some simple utilities to craft a quick request.

# SNMP Agents

SNMP agents do the bulk of the work. They are responsible for gathering information about the local system and storing them in a format that can be queried. updating a database called the "management information base", or **MIB**.

The MIB is a hierarchical, pre-defined structure that stores information that can be queried or set. This is available to well-formed SNMP requests originating from a host that has authenticated with the correct credentials (an SNMP manager).

The agent computer configures which managers should have access to its information. It can also act as an intermediary to report information on devices it can connect to that are not configured for SNMP traffic. This provides a lot of flexibility in getting your components online and SNMP accessible.

# SNMP Messages

**Get**: A Get message is sent by a manager to an agent to request the value of a specific OID. This request is answered with a Response message that is sent back to the manager with the data.

**GetNext**: A GetNext message allows a manager to request the next sequential object in the MIB.

**Set**: A Set message is sent by a manager to an agent in order to change the value held by a variable on the agent.

**Response**: This message, sent by an agent, is used to send any requested information back to the manager. It serves as both a transport for the data requested, as well as an acknowledgement of receipt of the request.

**Trap**: A trap message is generally sent by an agent to a manager.

**Inform**: To confirm the receipt of a trap, a manager sends an Inform message back to the agent. If the agent does not receive this message, it may continue to resend the trap message.

# Thank you!!

Note: These units are compilation of contents from Reference books/Text books/resources available online related to subject.

For details, pl read Books.