



# Application Layer Functionality and Protocols



## Network Fundamentals – Chapter 3

Cisco | Networking Academy®  
Mind Wide Open™

# Objectives

- Define the application layer as the source and destination of data for communication across networks.
- Explain the role of protocols in supporting communication between server and client processes.
- Describe the features, operation, and use of well-known TCP/IP application layer services (HTTP, DNS, SMTP).

# Applications – The Interface Between Human and Data Networks

- Two important concepts:

- Application layer:**

The application layer of the OSI model provides the first step of **getting data onto the network**.

- Application software:**

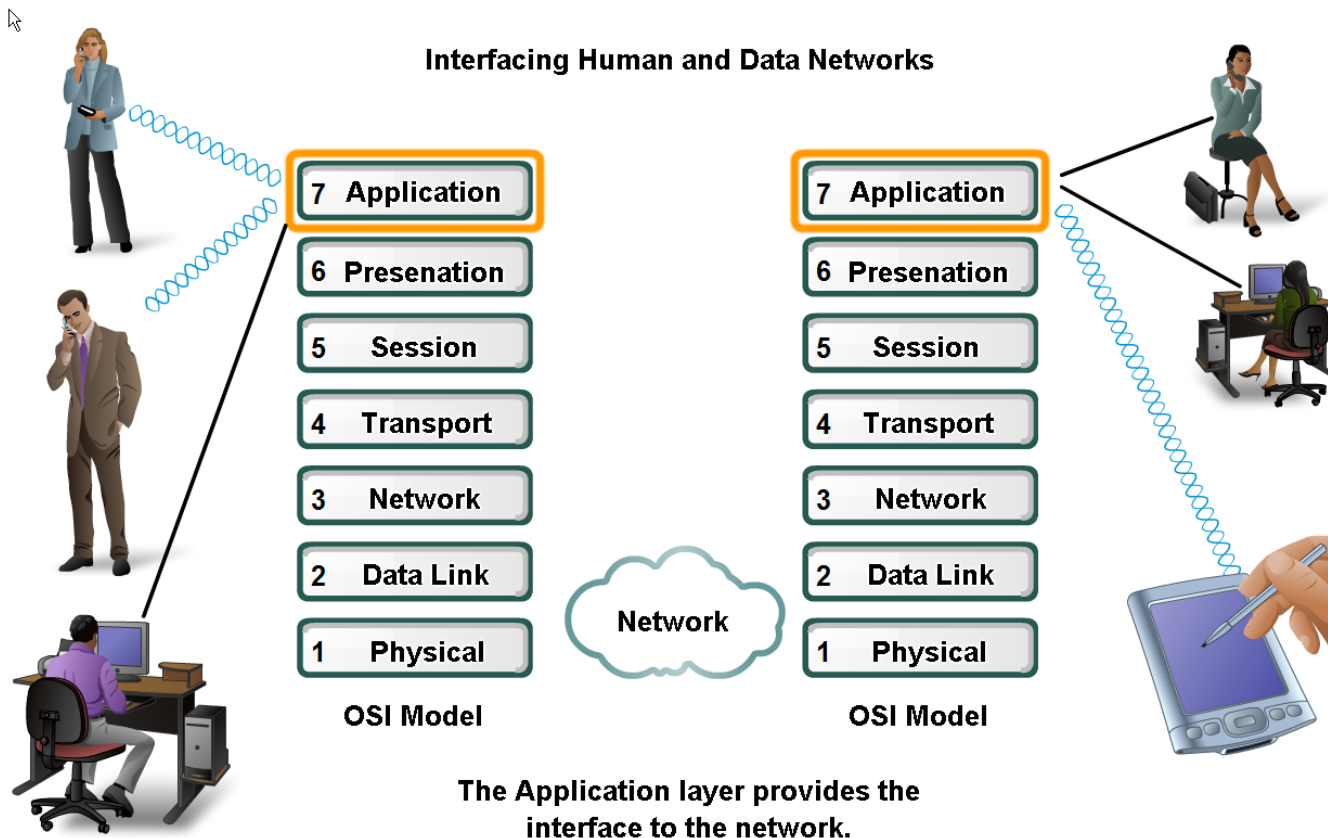
Applications are the **software programs used by people to communicate over the network**. Examples of application software, including HTTP, FTP, e-mail, and others, are used to explain the differences between these two concepts.



7 - Application - A  
 6 - Presentation - Person  
 5 - Session - Sent  
 4 - Transport - Through  
 3 - Network - Network  
 2 - Data Link - Data  
 1 - Physical - Packets

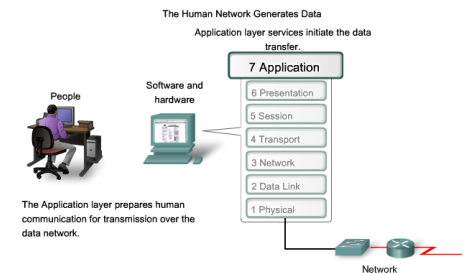
Physical - Please  
 Data Link - Do  
 Network - Not  
 Transport - Throw  
 Session - Sausage  
 Presentation - Pizza  
 Application - Away

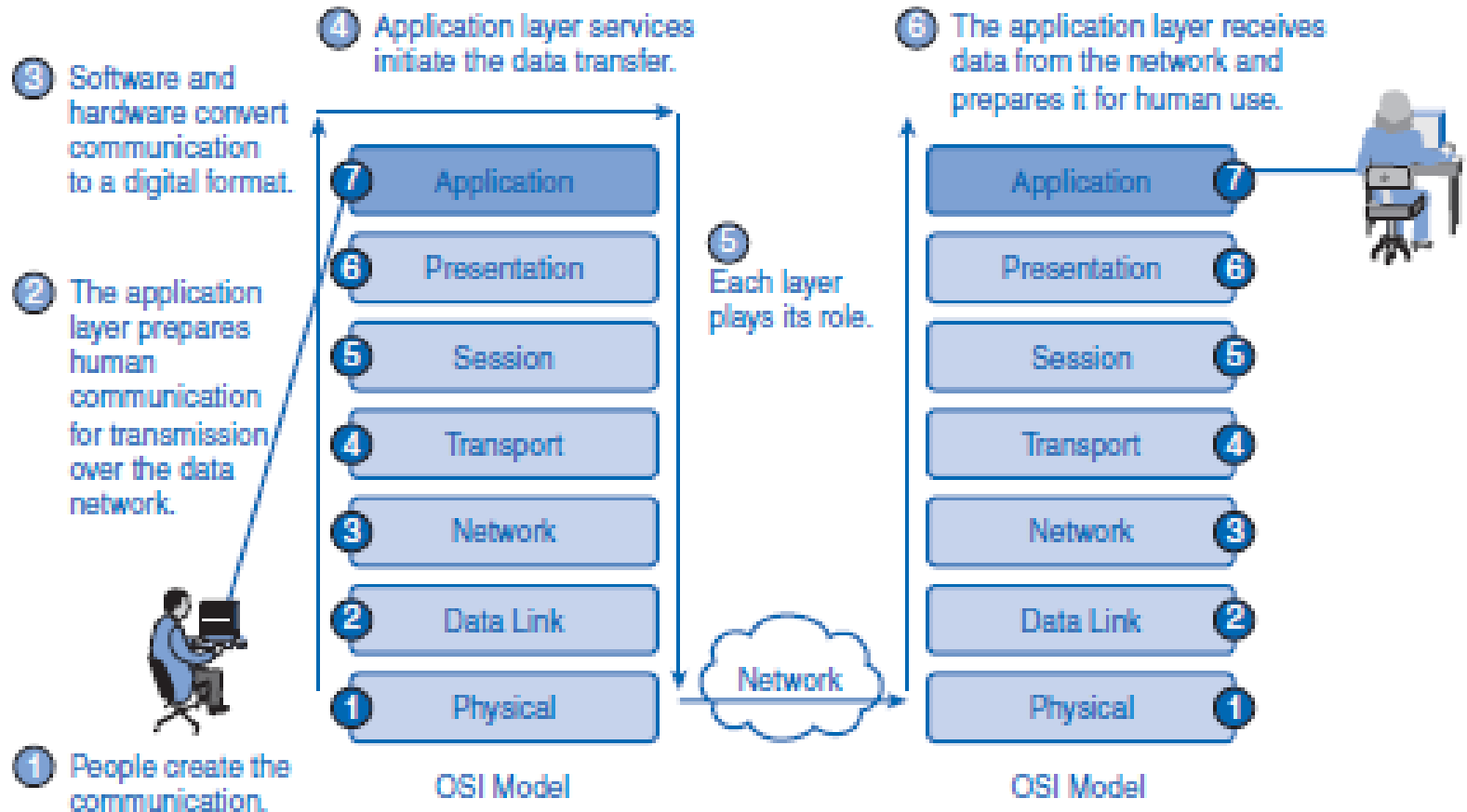
# Applications – The Interface Between Human and Data Networks



# Applications – The Interface Between Human and Data Networks

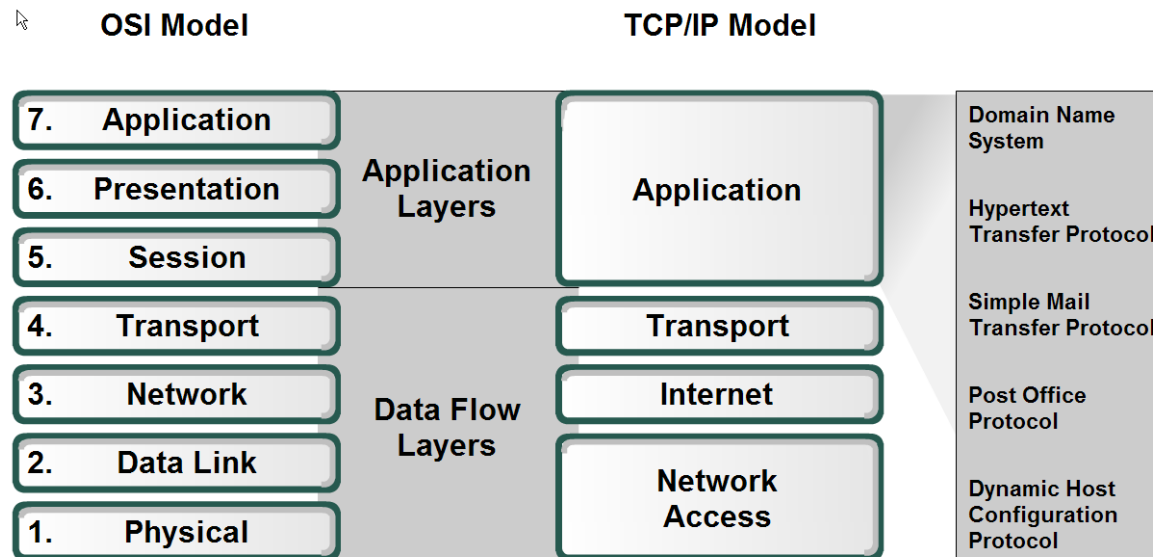
- The OSI reference model is a layered, abstract representation created as a guideline for network protocol design and instruction.
- The OSI model divides the networking process into **seven** logical layers, each of which has **unique functionality** and to which are assigned **specific services and protocols**.
- In the OSI model, **information is passed** from one layer to the next, **starting at the application layer** on the transmitting host and proceeding down the hierarchy to the physical layer, then passing over the communications channel to the destination host, where the information proceeds back up the hierarchy, **ending at the application layer**.



**Figure 3-2 OSI Encapsulation Process**


# Applications – The Interface Between Human and Data Networks

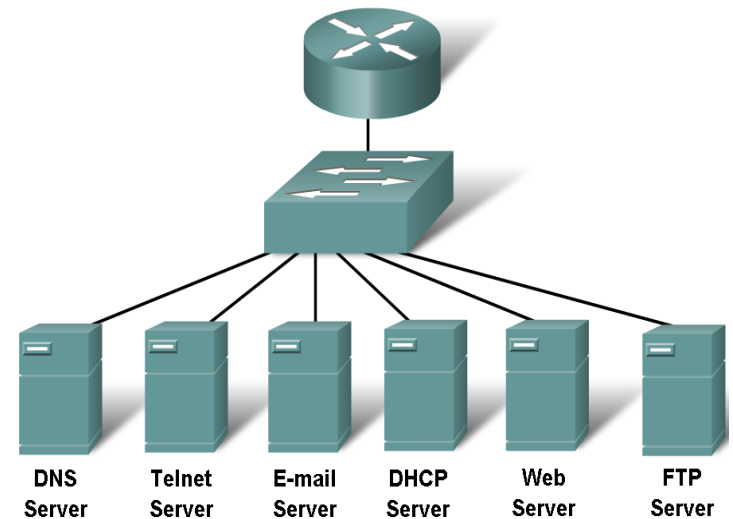
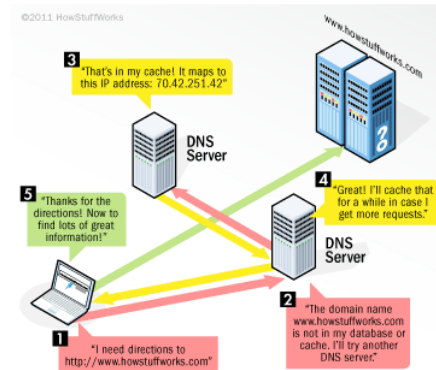
- Application layer protocols are used to exchange ***data between programs running*** on the source and destination hosts.
- Most applications, such as web browsers or e-mail clients, incorporate functionality of the OSI Layers 5, 6, and 7



# TCP/IP Application Layer Protocols

<http://techcrunch.com/2013/10/10/google-malaysia-site-hacked-credit-claimed-by-team-madlets/>

- Domain Name System (DNS) is used to **resolve Internet names** to IP addresses.
- Hypertext Transfer Protocol (HTTP) is used to **transfer files** that make up the web pages of the **World Wide Web**.
- Simple Mail Transfer Protocol (SMTP) is used for the **transfer of mail messages** and attachments.
- Telnet, a terminal emulation protocol, is used to provide **remote access** to servers and networking devices.
- File Transfer Protocol (FTP) is used for **interactive file transfer** between systems.



Server Farm



# Presentation Layer

- The presentation layer has three primary functions:
  - **Coding and conversion** of application layer data to ensure that data from the *source device* can be interpreted by the appropriate application on the destination device
  - **Compression of the data** in a manner that can be decompressed by the destination device
  - **Encryption of the data** for transmission and decryption of data upon receipt by the destination

# Application Layer Software

- Within the application layer, there are two forms of software programs or processes that provide access to the network: **applications and services.**
- Network-Aware **Applications**

Some end-user applications are network aware, meaning that they implement the application layer protocols and are able to communicate directly with the lower layers of the protocol stack. **E-mail clients and web browsers** are examples of these types of applications.
- Application Layer **Services**

Other programs, such as **file transfer or network print spooling**, might need the assistance of application layer services to use network resources. Although transparent to the user, these services **interface with the network and prepare the data for transfer.** Different types of data—whether it is text, graphics, or video—require different network services to ensure that it is properly prepared for processing by the functions occurring at the lower layers of OSI model.

# Application Layer Software

Each executing program loaded on a device is referred to as a process.

Figure 3-4 Software Processes

Processes are individual software programs running concurrently.

Applications

Services

One program can be running several times, each in its own process.

System Operations

Image Name	User Name	CPU	Mem Usage
Apoinet.exe	frances	00	5,288 K
justhed.exe	frances	00	1,920 K
EXCEL.EXE	frances	00	2,804 K
quicksel.exe	frances	00	4,244 K
DSEntry.exe	frances	00	1,940 K
Director.exe	frances	00	3,590 K
wdfmgr.exe	LOCAL SERVICE	00	1,716 K
svchost.exe	LOCAL SERVICE	00	4,384 K
alg.exe	LOCAL SERVICE	00	3,512 K
scardsvr.exe	LOCAL SERVICE	00	2,564 K
svchost.exe	NETWORK SERVICE	00	3,744 K
svchost.exe	NETWORK SERVICE	00	4,440 K
msdtc.exe	NETWORK SERVICE	00	4,952 K
System Idle Process	SYSTEM	96	16 K
System	SYSTEM	00	224 K
svchost.exe	SYSTEM	00	5,152 K
ViewpointService....	SYSTEM	00	2,208 K
WZTRNSVC.EXE	SYSTEM	00	1,368 K
WZTRNSVC.exe	SYSTEM	00	1,092 K

Show processes from all users
 End Process

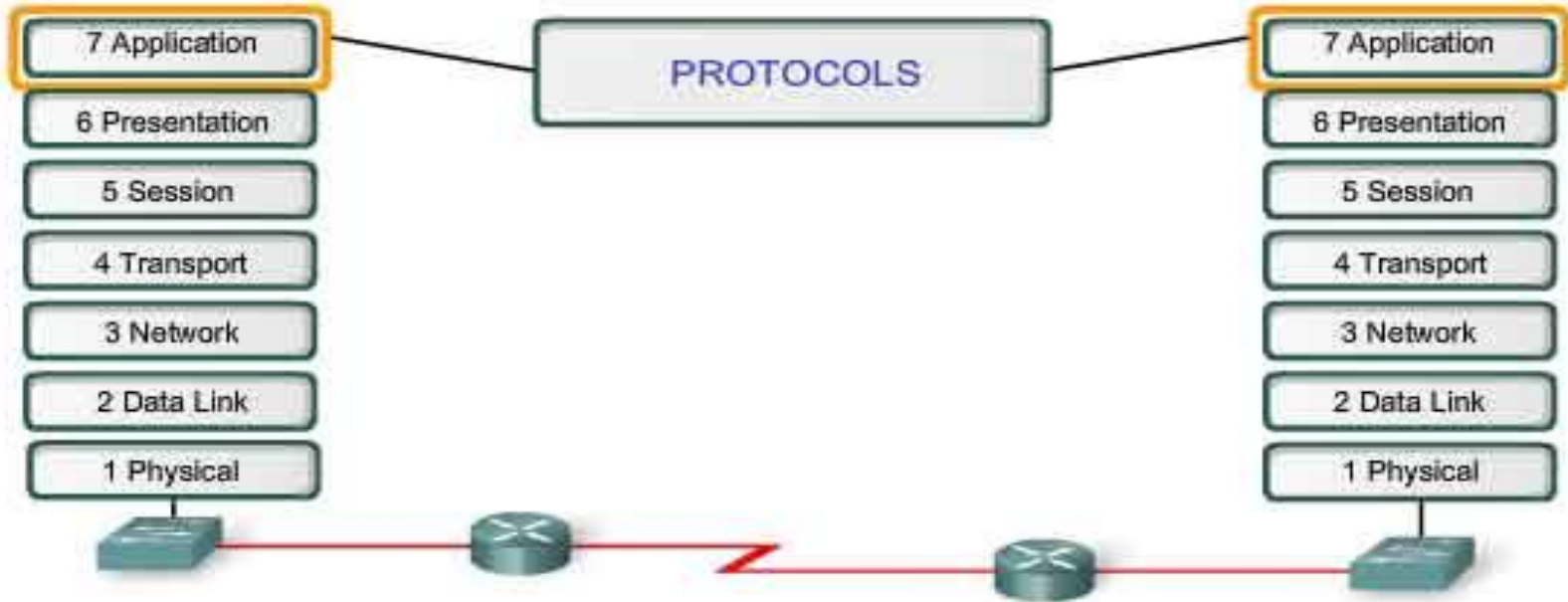
Processes: 64    CPU Usage: 4%    Commit Charge: 507M / 2463M

tasklist /SVC

# Application Layer Protocol Functions

- Both the source and destination devices use **application layer protocols** during a communication *session*
- Protocols perform the following tasks:
  - **Establish consistent rules** for exchanging data between applications and services loaded on the participating devices.
  - Specify how data inside the **messages is structured** and the **types of messages** that are sent between source and destination - can be **requests** for services, **acknowledgments**, **data** messages, **status** messages, or **error** messages.
  - Define **message dialogues**, ensuring that a message being sent is met by the expected response and that the correct services are invoked when data transfer occurs.

# Application Layer Protocol Functions



Application layer protocols provide the rules for communication between applications.

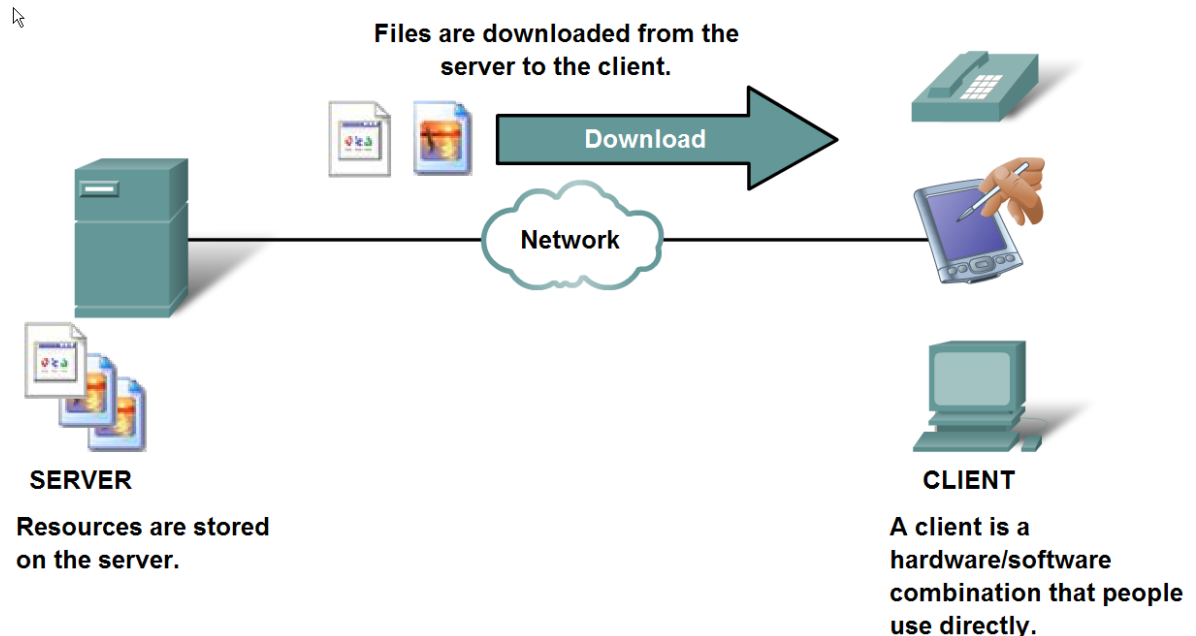
## Protocols:

- Define processes on either end of the communication
- Define the types of messages
- Define the syntax of messages
- Define the meaning of any informational fields
- Define how messages are sent and the expected response
- Define interaction with the next lower layer

# The Role of Protocols in Supporting Communication

- In the client/server model, the device **requesting the information** is called a **client** and the **device responding** to the request is called a **server**.
- The **client begins the exchange** by requesting data from the server, which responds by sending one or more streams of data to the client.

Client/Server Model

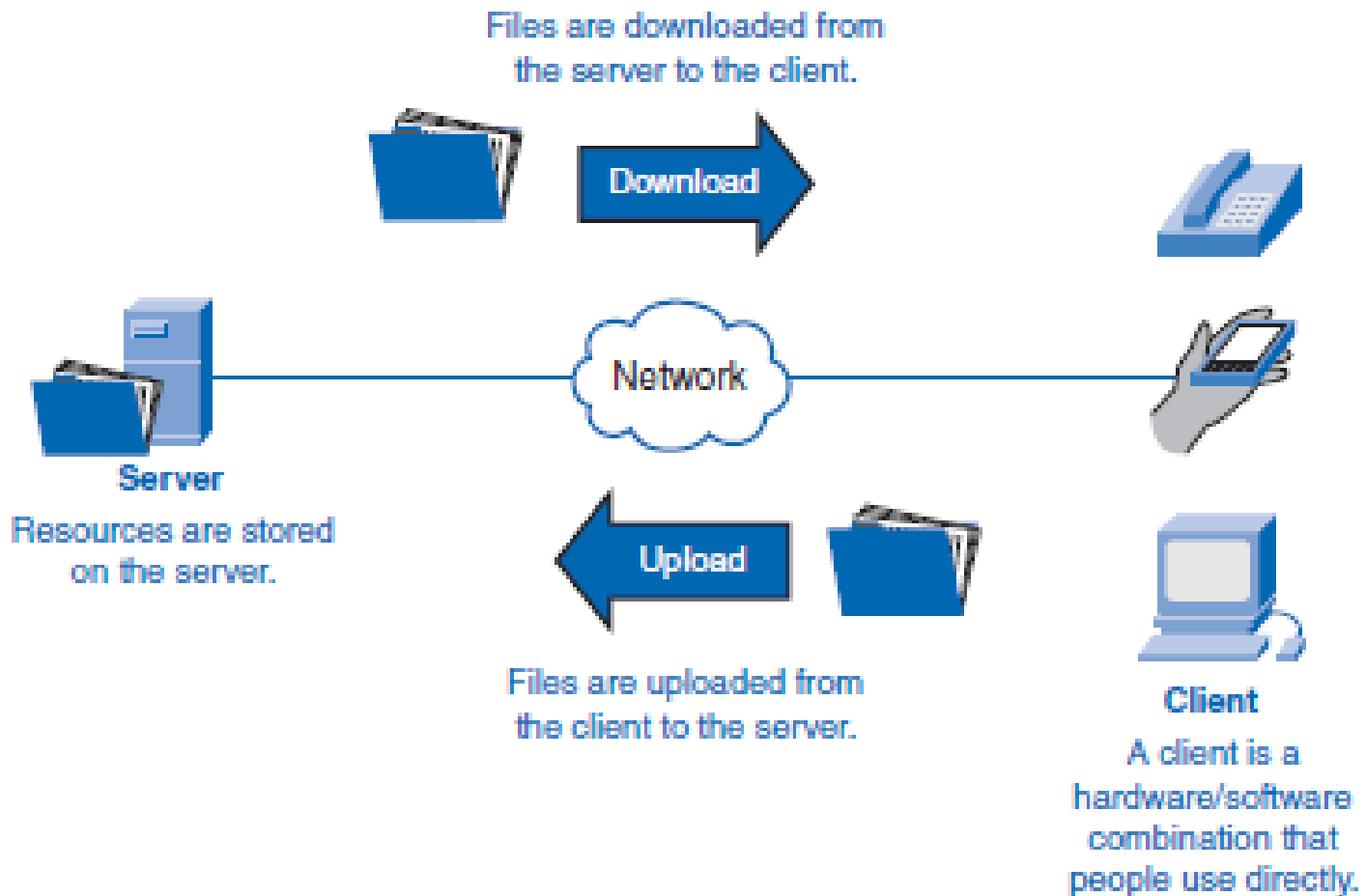


# The Role of Protocols in Supporting Communication

- One example of a client/server network is a corporate environment where employees use a company **e-mail server to send, receive, and store e-mail.**
  - The **e-mail client** on an employee computer issues a **request** to the e-mail server for any unread mail.
  - The **server responds** by sending the requested e-mail to the client.
- Although data is typically described as flowing from the server to the client, some data always flows from the client to the server. For example,
  - A client might transfer a file to the server for storage purposes.
- Data transfer from a **client to a server** is referred to as an **upload**, and data from a **server to a client** is a **download**.

# Client/Server Model

Figure 3-6 Client/Server Model

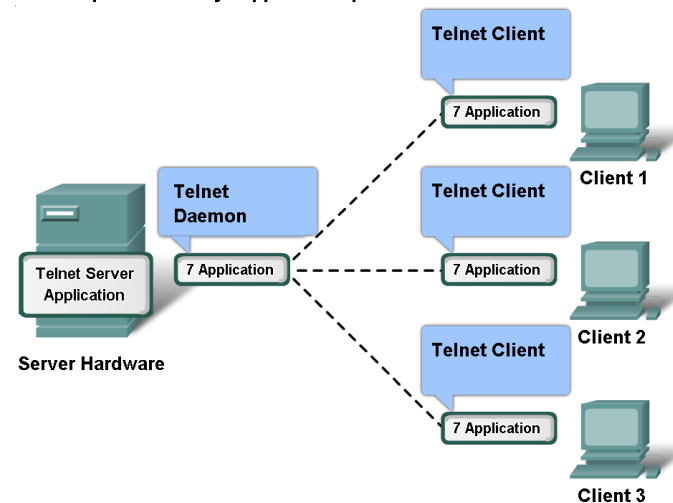




# Client/Server Model

- A single application can employ many different supporting application layer services. Thus, what appears to the user as one request for a web page might, in fact, amount to dozens of individual requests. (dns,www)
- Additionally, servers typically have multiple clients requesting information at the same time.
- For example, a Telnet server can have many clients requesting connections to it. These individual client requests must be handled simultaneously and separately for the network to succeed.

Server processes may support multiple clients.



# Peer-to-Peer (P2P) Networking and Applications

- P2P networking involves two distinct forms:
  - peer-to-peer network design and
  - peer-to-peer applications.
- Both forms have similar features but in practice work very differently.

# Peer-to-Peer (P2P) Networking and Applications

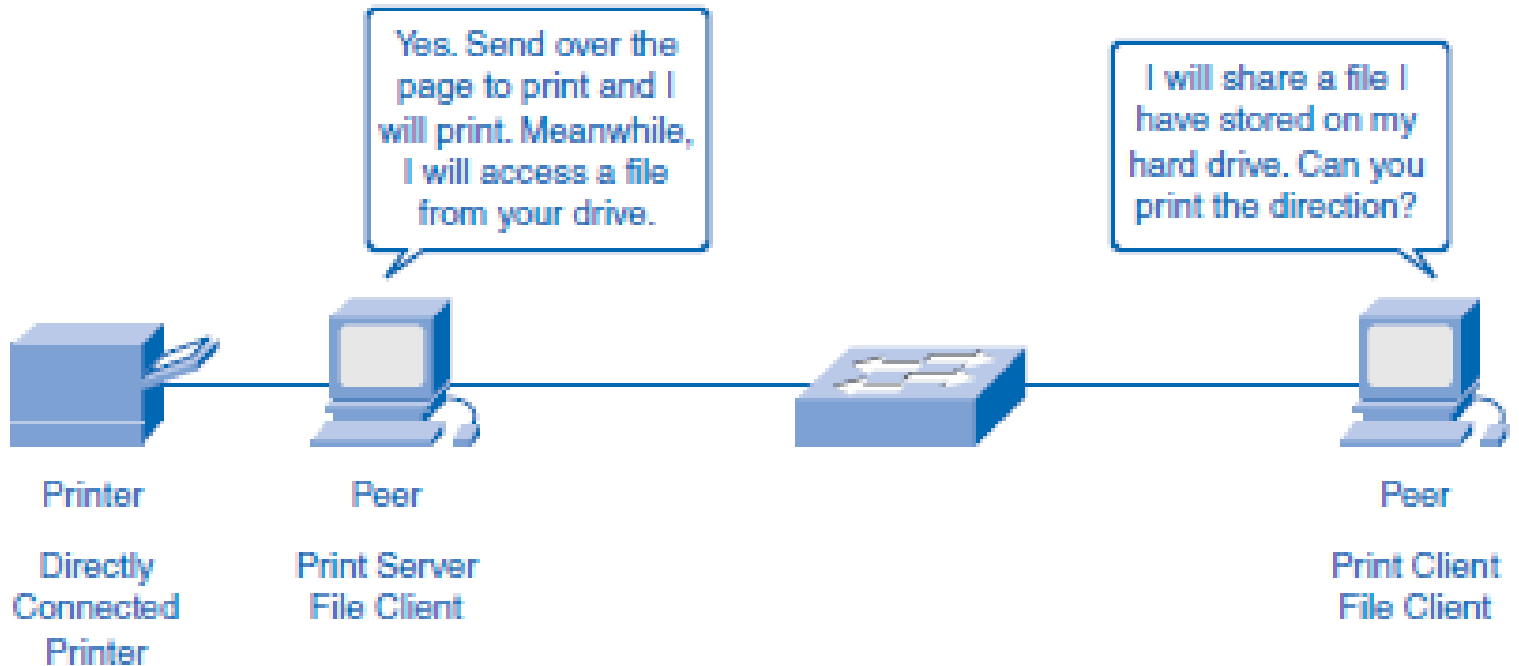
## ■ P2P Networks



- In a peer-to-peer network, two or more computers are **connected through a network** and can **share resources** such as printers and files **without having a dedicated server**.
- Every connected end device, known as a peer, can function as **either a server or a client**.
- A simple home network with two connected computers sharing a printer is an example of a peer-to-peer network. Each person can set his or her computer to share files, enable networked games, or share an Internet connection.
- Unlike the client/server model, which uses dedicated servers, peer-to-peer networks decentralize the resources on a network.

# Peer-to-Peer (P2P) Networking and Applications

**Figure 3-9 Peer-to-Peer Networking**



In a peer-to-peer exchange, both devices are considered equal in the communication process.

# Peer-to-Peer (P2P) Networking and Applications

- P2P Applications

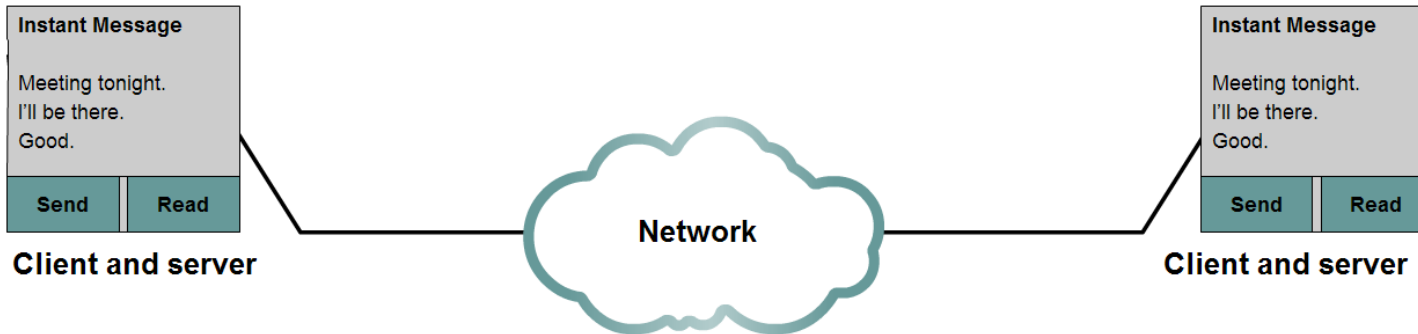
- Unlike a peer-to-peer network, allows a device to act **as both a client and a server** within the same communication session.
- In this model, every client is a server and every server a client.
- Both can initiate a communication and are considered equal in the communication process.
- Peer-to-peer applications can be used on peer-to-peer networks, in client/server networks, and across the Internet

# Peer-to-Peer (P2P) Networking and Applications

## ■ P2P Applications

### Peer-to-Peer Applications

Client and server in the same communication



**Both clients:**

- Initiate a message
- Receive a message

**Both clients simultaneously:**

- Send
- Receive



# Application Layer Protocols and Services Examples

- The transport layer uses an addressing scheme called a [port number](#).
- [Port numbers identify applications](#) and application layer services that are the source and destination of data.
- Server programs generally use predefined port numbers that are commonly known by clients. Some of these services are
  - Domain Name System (DNS): TCP/UDP port 53
  - HTTP: TCP port ?
  - Simple Mail Transfer Protocol (SMTP): TCP port 25
  - Post Office Protocol (POP): UDP port 110
  - Telnet: TCP port 23
  - DHCP: UDP port 67
  - FTP: TCP ports 20 and 21

# DNS Services and Protocol

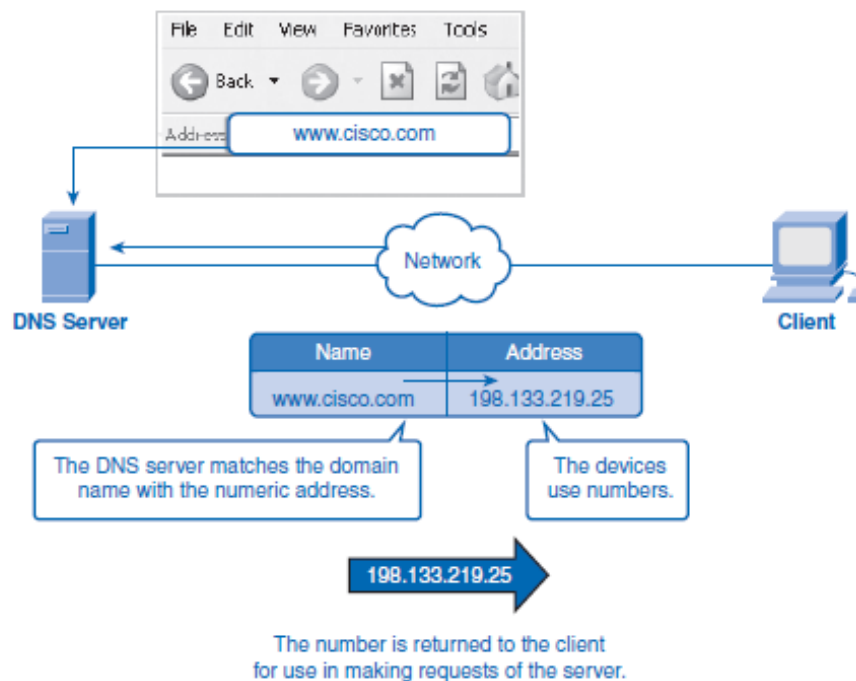
- In data networks, devices are assigned ***IP addresses*** so that they can participate in sending and receiving messages over the network.
- However, most people have a hard time remembering this numeric address.
- Hence, domain names were created to convert the numeric address into a simple, recognizable name.
- On the Internet, these domain names, such as `http://www.cisco.com`, are much easier for people to remember than `?.?.?.?`
- If Cisco decides to change the numeric address, it is transparent to the user, because the ***domain name*** will remain `http://www.cisco.com`. The new address will simply be linked to the existing domain name and connectivity is maintained



# DNS Services and Protocol

- <http://www.usim.edu.my>
- [nslookup](#)

Figure 3-11 Resolving DNS Addresses



# DNS Hierarchy

- DNS uses a hierarchical system to create a name database to provide name resolution.
- At the top of the hierarchy, the root servers maintain records about how to reach the toplevel domain servers, which in turn have records that point to the secondary-level domain servers and so on.
- The different top-level domains represent either the type of organization or the country of origin.

**.au:** Australia

**.co:** Colombia

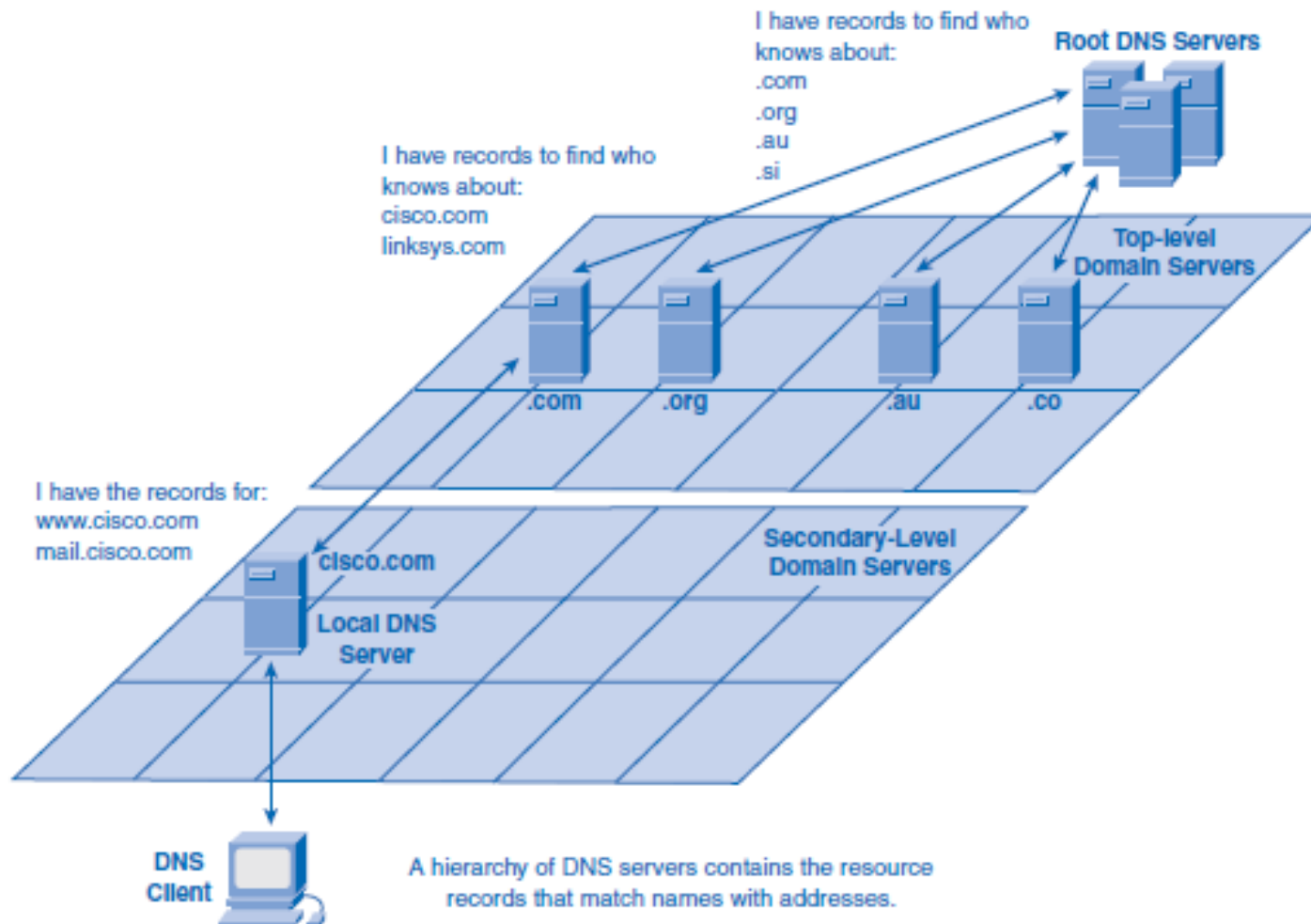
**.com:** A business or industry

**.jp:** Japan

**.org:** A nonprofit organization

# DNS Hierarchy

Figure 3-12 DNS Server Hierarchy



# WWW Service and HTTP

- When a web address (or URL) is typed into a web browser, the web browser establishes a connection to the web service running on the server using HTTP. URLs and URIs (uniform resource identifiers) are the names most people associate with web addresses.
- The URL `http://www.cisco.com/index.html` refers to a specific resource—a web page named `index.html` on a server identified as `cisco.com`.
- Web browsers are the client applications computers use to connect to the World Wide Web and access resources stored on a web server

# WWW Service and HTTP

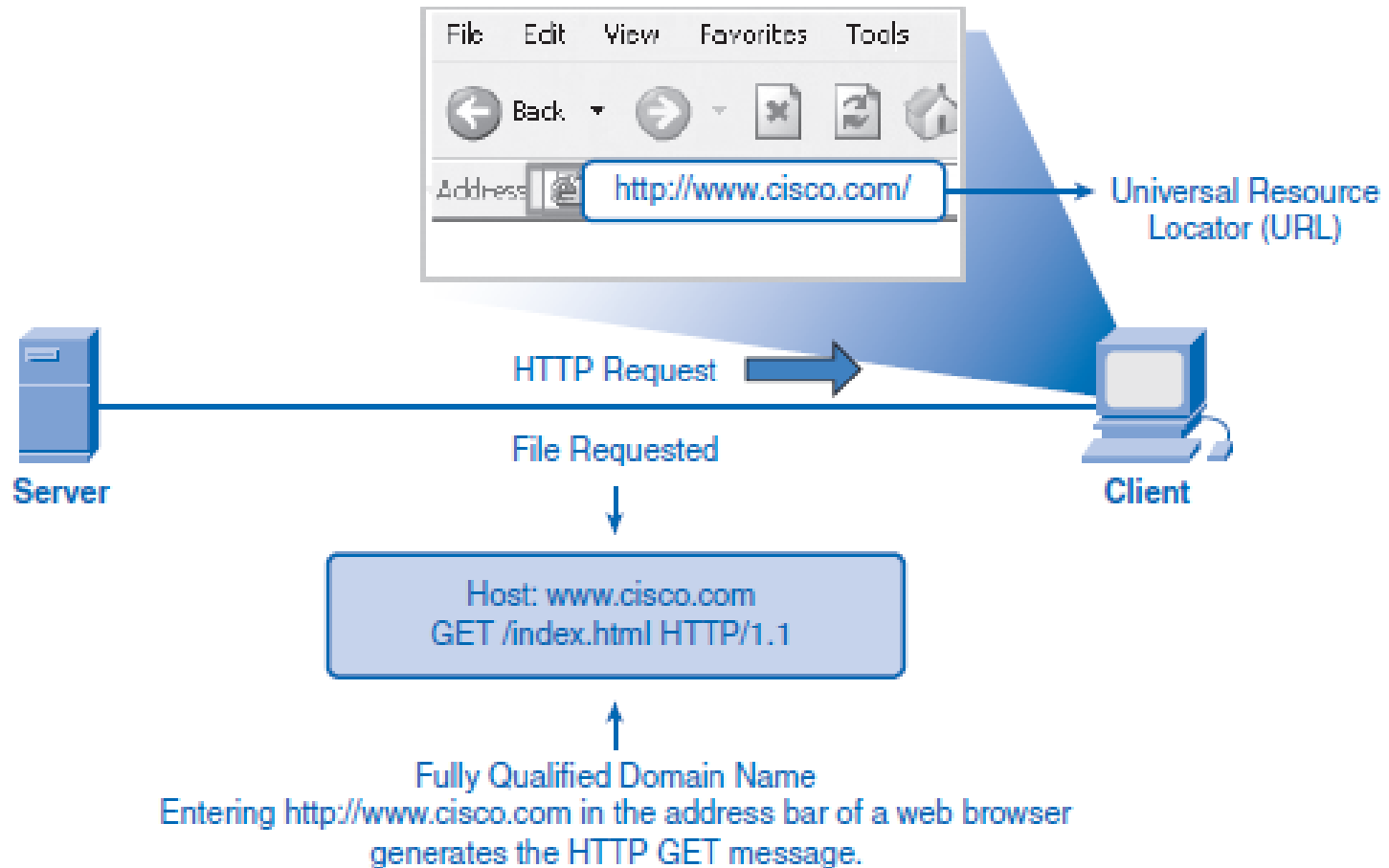
- consider the URL <http://www.cisco.com/web-server.htm>.
  - First, the browser interprets the three parts of the URL:
    - http: The protocol or scheme
    - www.cisco.com: The server name
    - web-server.htm: The specific filename requested
  - Using the HTTP requirements, the browser sends a GET request to the server and asks for the file web-server.htm.
  - The server in turn sends the HTML code for this web page to the browser. Finally, the browser deciphers the HTML code and formats the page for the browser window.

# WWW Service and HTTP

- HTTP specifies a request/response protocol. When a client, typically a web browser, sends a request message to a server, the HTTP protocol defines the message types the client uses to request the web page and the message types the server uses to respond. The three common message types are:
  - GET
  - POST
  - PUT
- GET is a client request for data. A web browser sends the GET message to request pages from a web server.
- POST and PUT are used to send messages that upload data to the web server. For example, when the user enters data into a form embedded in a web page, POST includes the data in the message sent to the server. PUT uploads resources or content to the web server.

# WWW Service and HTTP

Figure 3-13 HTTP Protocol Using GET



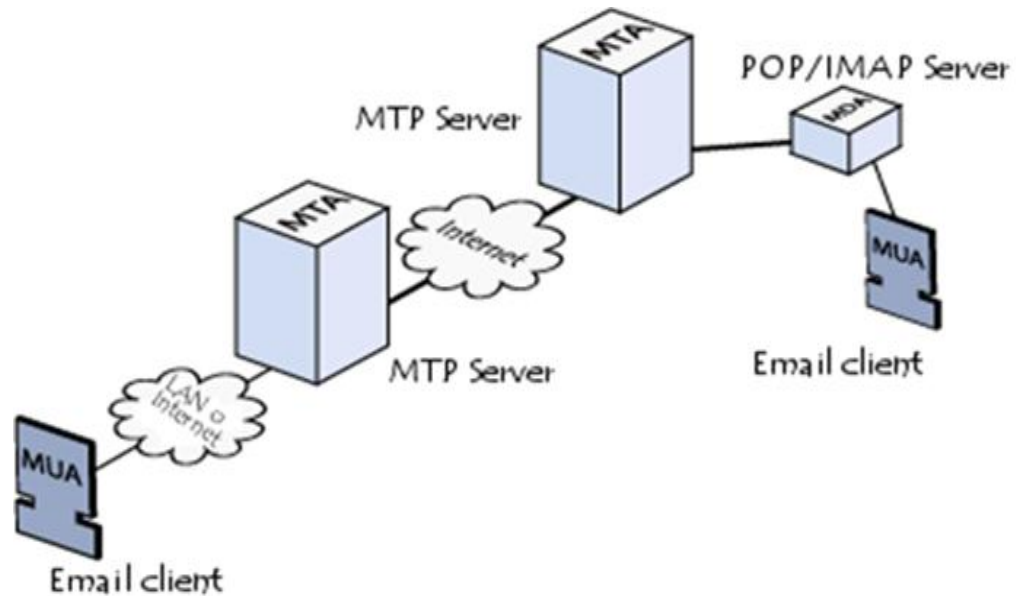
## E-Mail Services and SMTP/POP Protocols

- To run on a computer or other end device, e-mail requires several applications and services
- Two examples of application layer protocols are ***Post Office Protocol (POP)*** and ***Simple Mail Transfer Protocol (SMTP)***.
- POP and POP3 (Post Office Protocol, version 3) are inbound mail delivery protocols and are typical client/server protocols. They deliver e-mail from the e-mail server to the client (MUA).
- SMTP, on the other hand, governs the transfer of outbound e-mail from the sending client to the e-mail server (MDA), as well as the transport of e-mail between e-mail servers (MTA).



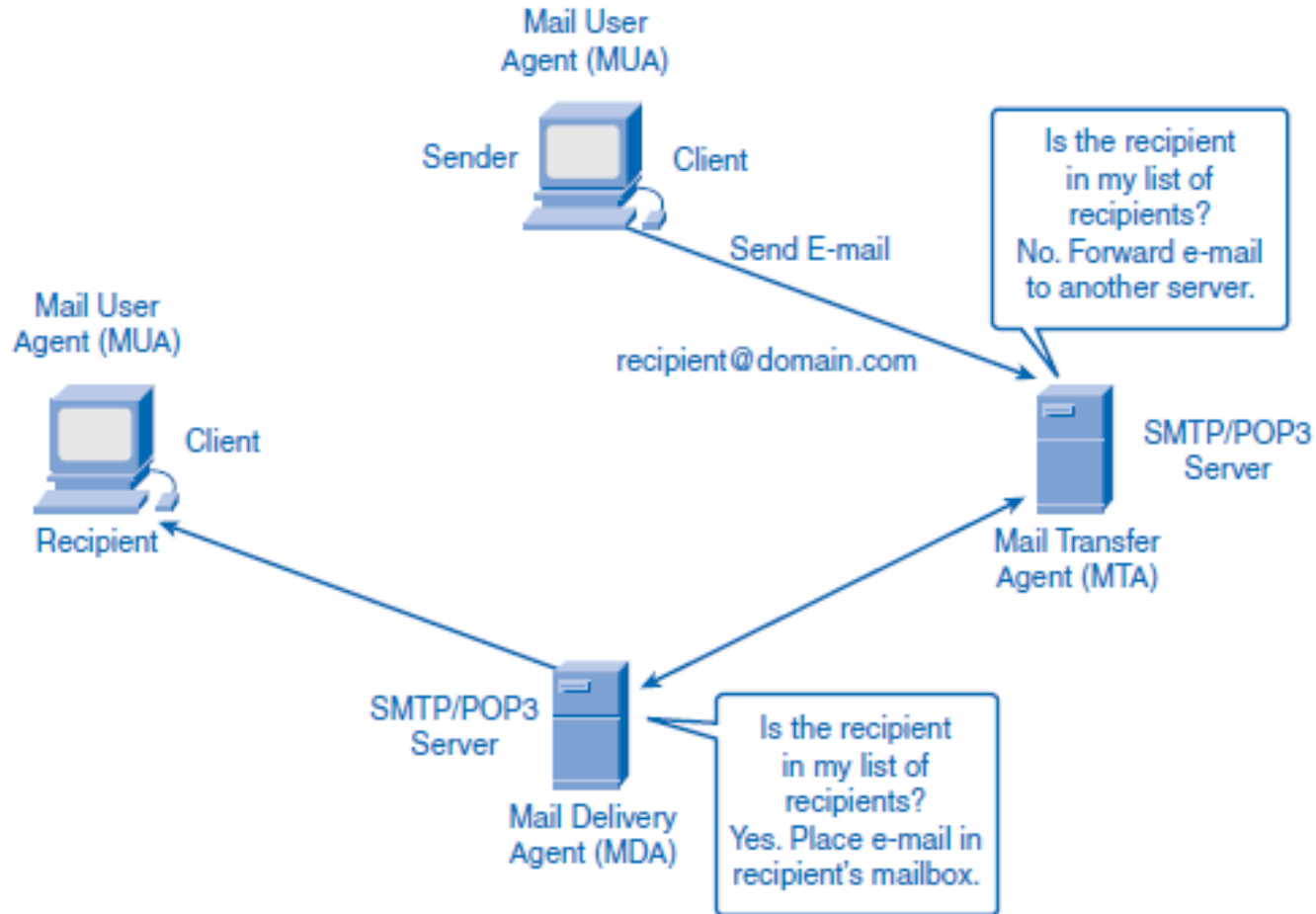
# E-Mail Services and SMTP/POP Protocols

- The e-mail server operates two separate processes:
  - Mail Transfer Agent (MTA)
  - Mail Delivery Agent (MDA)
- The Mail Transfer Agent (MTA) process is used to forward e-mail.
- The MTA receives messages from the MUA or from another MTA on another e-mail server.



# E-Mail Services and SMTP/POP Protocols

Figure 3-16 E-Mail Server: MDA



# Summary

## In this chapter, you learned to:

- Describe how the functions of the three upper OSI model layers provide network services to end user applications.
- Describe how the TCP/IP Application layer protocols provide the services specified by the upper layers of the OSI model.
- Define how people use the Application layer to communicate across the information network.
- Describe the function of well-known TCP/IP applications, such as the World Wide Web and email, and their related services (HTTP, DNS, SMB, DHCP, STMP/POP, and Telnet).
- Describe file-sharing processes that use peer-to-peer applications and the Gnutella protocol.
- Explain how protocols ensure services running on one kind of device can send to and receive data from many different network devices.
- Use network analysis tools to examine and explain how common user applications work.

