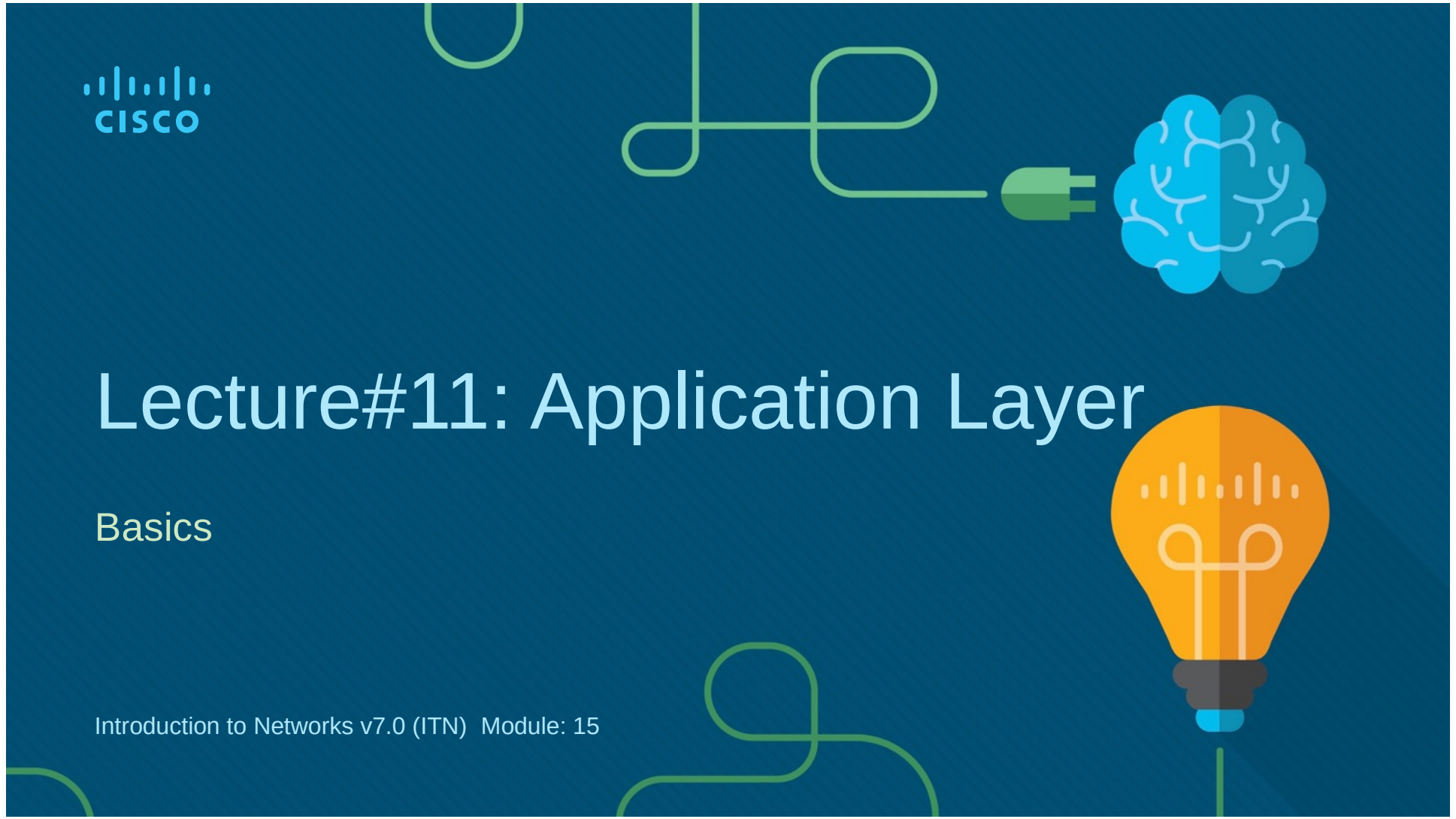


Lecture#11: Application Layer

Basics

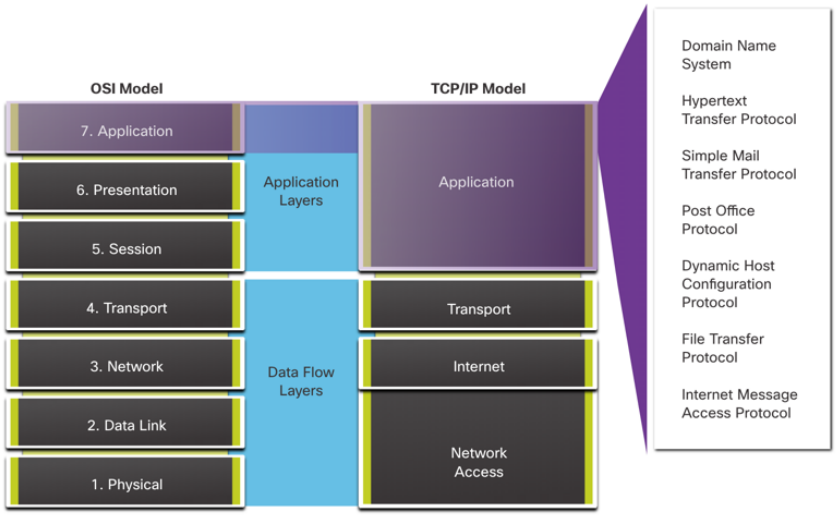
Introduction to Networks v7.0 (ITN) Module: 15

11.1.1 Functionalities

Application, Presentation, and Session

Application Layer

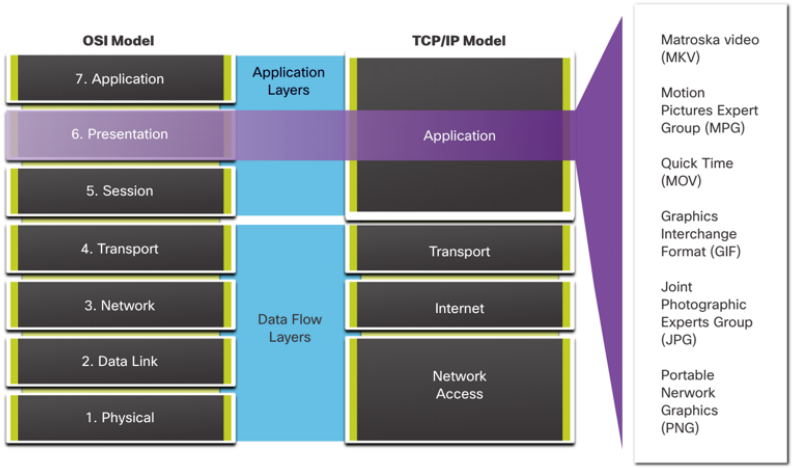
- The **upper three layers** of the OSI model (application, presentation, and session) define functions of the TCP/IP **application layer**.
- The application layer provides the **interface** between the applications used to communicate, and the underlying network over which messages are transmitted.
- Some of the most **widely known** application layer protocols include HTTP, FTP, TFTP, IMAP and DNS.



Presentation and Session Layer

The **presentation layer** has three primary functions:

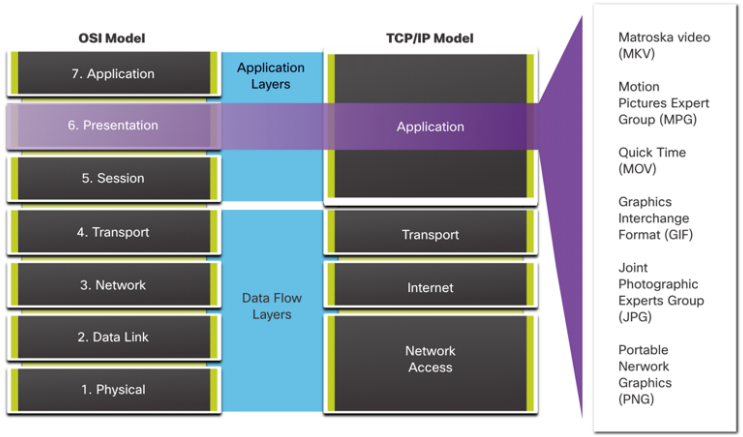
- **Formatting**, or **presenting**, data at the source device into a compatible format for receipt by the destination device.
- **Compressing** data in a way that can be decompressed by the destination device.
- **Encrypting** data for transmission and decrypting data upon receipt.



Presentation and Session Layer

The **session layer** has three primary functions:

- It creates and maintains **dialogs** between source and destination applications.
- It handles the **exchange of information** to initiate dialogs, keep them active, and to restart sessions that are disrupted or idle for a long period of time.



Application, Presentation, and Session

TCP/IP Application Layer Protocols

- The **TCP/IP application protocols** specify the format and control information necessary for many common internet communication functions.
  - Application layer protocols are used by both the source and destination devices during a communication session.
  - For the communications to be successful, the application layer protocols that are implemented on the source and destination host must be compatible.

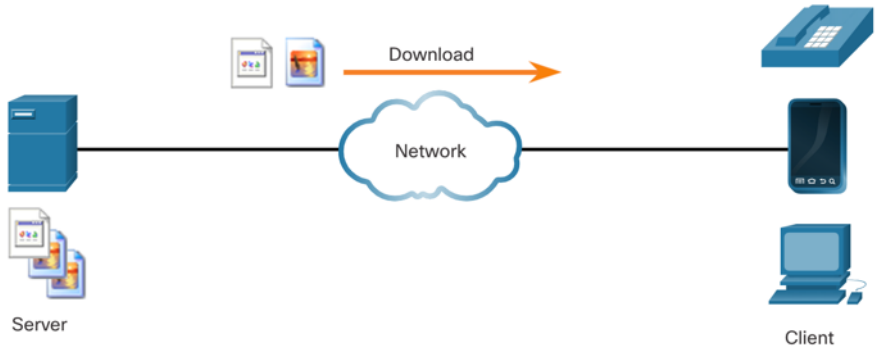
Name System - DNS	**Host Config - DHCP**	**Web - HTTP**
• TCP, UDP client 53 • Translates domain names, such as cisco.com, into IP addresses.	• UDP server/client 67/68 • Dynamically assigns IP addresses to be re-used when no longer needed	• TCP 80, 8080 • Exchanging text, graphic images, sound, video, and others on the WWW

11.1.2 Peer-to-Peer

Peer-to-Peer

Client-Server Model

- **Client** and **server** processes are considered to be in the application layer.
- 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.
- Application layer protocols describe the **format of** the **requests** and **responses** between clients and servers.

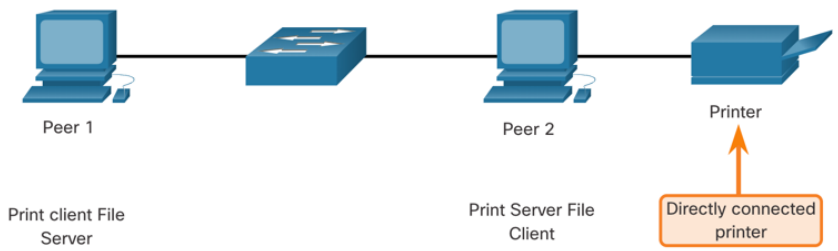


Peer-to-Peer Networks

- In a **peer-to-peer** (P2P) network, two or more computers are connected via a network and can share resources (such as printers and files) without having a dedicated server.

- Every connected end device (**peer**) can function as both a server and a client.

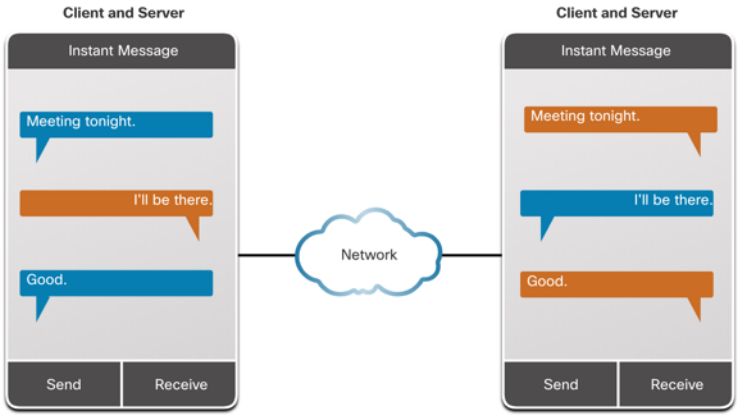
- One computer might assume the role of server for one transaction while simultaneously serving as a client for another. The roles of client and server are set on a per request basis.



Peer-to-Peer Applications

- A **P2P application** allows a device to act as **both** a client and a server within the same communication.

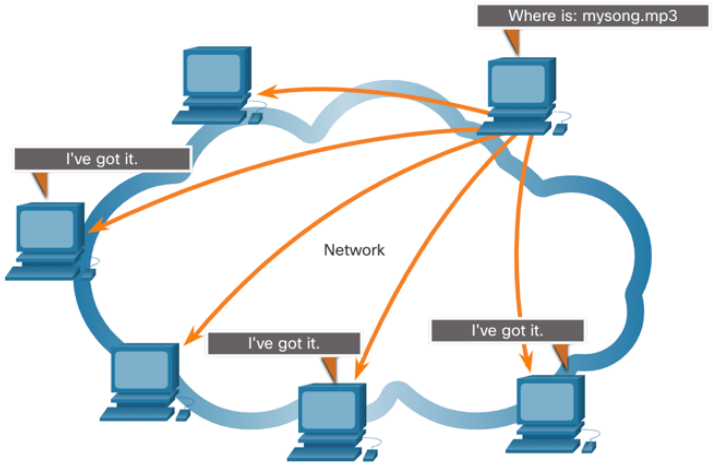
- Some P2P applications use a hybrid system where each peer accesses an index server to get the location of a resource stored on another peer.



Common Peer-to-Peer Applications

- With **P2P applications**, each computer in the network that is running the application can act as a client or a server for the other computers in the network that are also running the application.

- Common P2P networks include the following:
  - § BitTorrent
  - § Direct Connect
  - § eDonkey
  - § Freenet



Lecture#11: Application Layer

Web and Email Services

Introduction to Networks v7.0 (ITN) Module: 15

11.2.1 Web Protocols

Hypertext Transfer Protocol and Hypertext Markup Language

When a web address or Uniform Resource Locator (URL) is typed into a web browser, the web browser establishes a connection to the web service. The web service is running on the server that is using the HTTP protocol.

To better understand how the web browser and web server interact, examine how a web page is opened in a browser.

Step 1

The browser interprets the three parts of the URL:

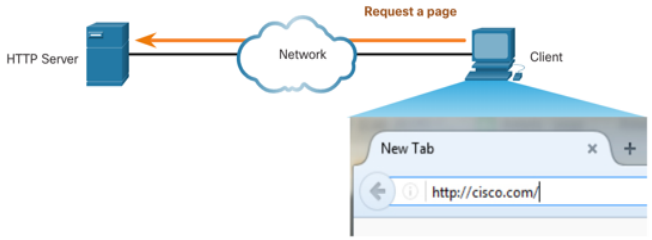
- http (the protocol or scheme)
- www.cisco.com (the server name)
- index.html (the specific filename requested)

Hypertext Transfer Protocol and Hypertext Markup Language (Cont.)

Step 2

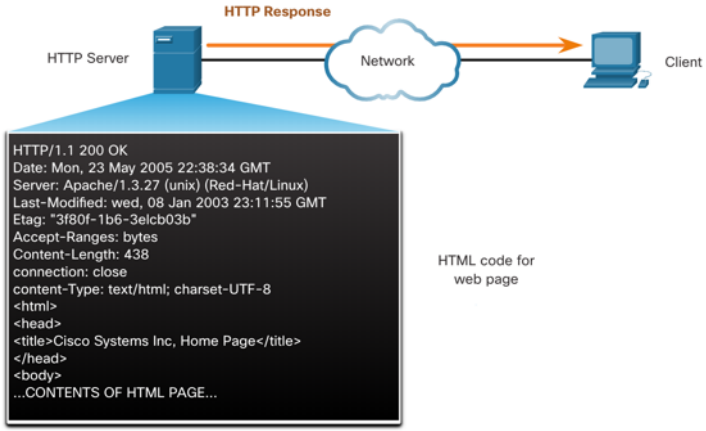
The browser then checks with a name server to convert www.cisco.com into a numeric IP address, which it uses to connect to the server.

The client initiates an HTTP request to a server by sending a GET request to the server and asks for the index.html file.



Step 3

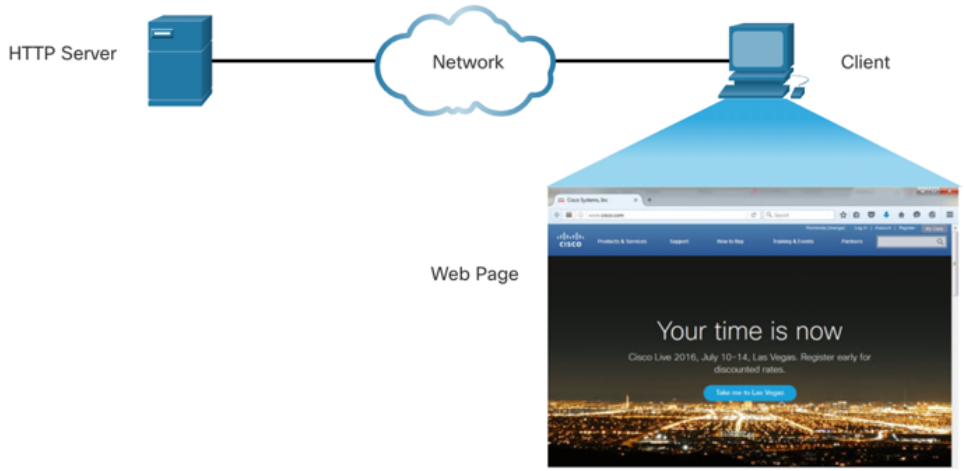
In response to the request, the server sends the HTML code for this web page to the browser.



Hypertext Transfer Protocol and Hypertext Markup Language (Cont.)

Step 4

The browser deciphers the HTML code and formats the page for the browser window.

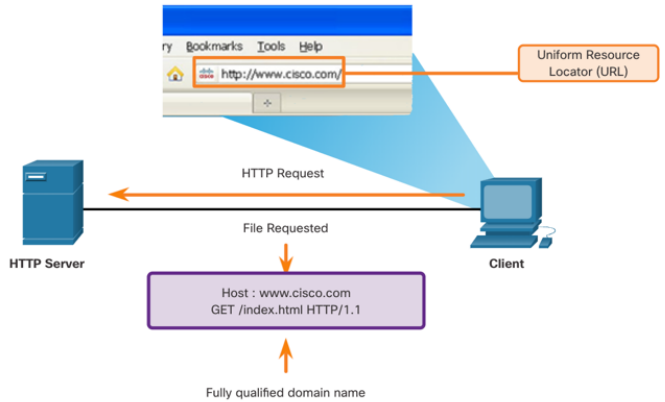


HTTP and HTTPS

HTTP is a request/response protocol that specifies the message types used for that communication.

The three common message types are GET, POST, and PUT:

- **GET** - This is a client request for data. A client (web browser) sends the GET message to the web server to request HTML pages.
- **POST** - This uploads data files to the web server, such as form data.
- **PUT** - This uploads resources or content to the web server, such as an image.



Note: For secure communications sent across the internet, HTTPS should be used.

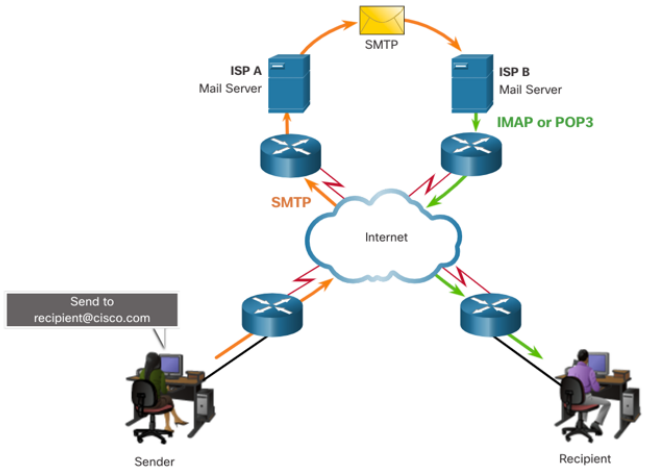
11.2.2 Email Protocols

Email Protocols

Email is a **store-and-forward** method of sending, storing, and retrieving electronic messages across a network. Email messages are stored in databases on mail servers. Email clients communicate with mail servers to send and receive email.

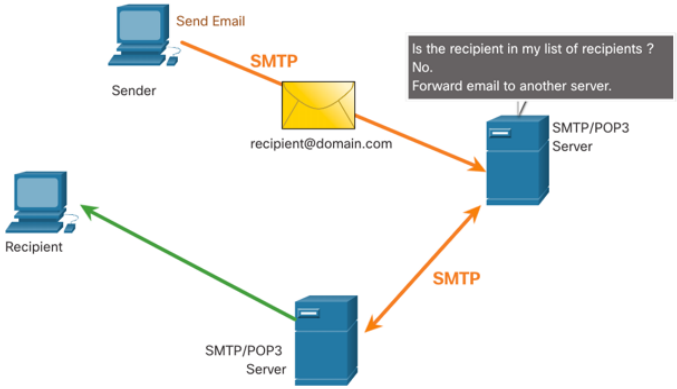
The email protocols used for operation are :

- **Simple Mail Transfer Protocol** (SMTP) – used to send mail.
- **Post Office Protocol** (POP) & IMAP – used for clients to receive mail.



SMTP, POP and IMAP

- When a client sends email, the **client SMTP** process connects with a **server SMTP** process on well-known port 25.
- After the connection is made, the client attempts to send the email to the server across the connection.
- When the server receives the message, it either places the message in a local account, if the recipient is local, or forwards the message to another mail server for delivery.
- The destination email server may not be online or may be busy. If so, SMTP spools messages to be sent at a later time.

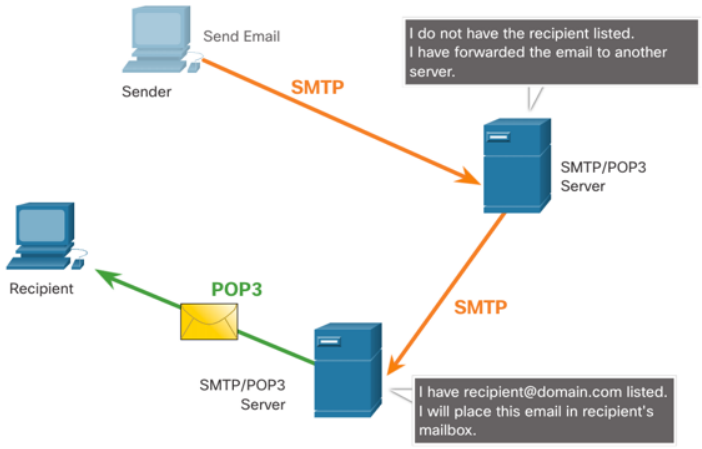


***Note**: SMTP message formats require a message header (recipient email address & sender email address) and a message body.*

SMTP, POP and IMAP (Cont.)

POP is used by an application to retrieve mail from a mail server. When mail is downloaded from the server to the client using POP the messages are then deleted on the server.

- The server starts the POP service by passively listening on TCP port 110 for client connection requests.
- When a client wants to make use of the service, it sends a request to establish a TCP connection with the server.
- When the connection is established, the POP server sends a greeting.
- The client and POP server then exchange commands and responses until the connection is closed or aborted

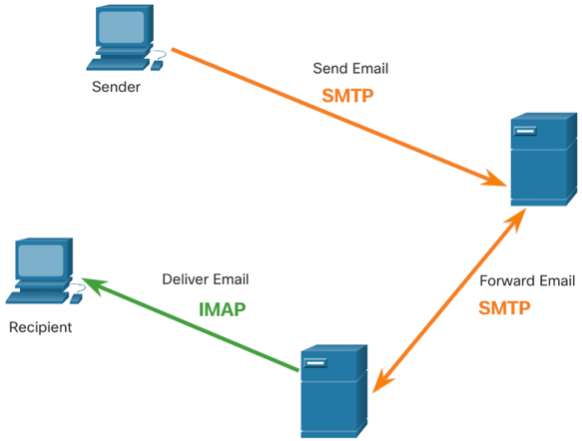


***Note**: Since POP does not store messages, it is not recommended for small businesses that need a centralized backup solution.*

SMTP, POP and IMAP (Cont.)

IMAP is another protocol that describes a method to retrieve email messages.

- Unlike POP, when a user connects to an IMAP server, copies of the messages are downloaded to the client application. The original messages are kept on the server until manually deleted.
- When a user decides to delete a message, the server synchronizes that action and deletes the message from the server.




CISCO