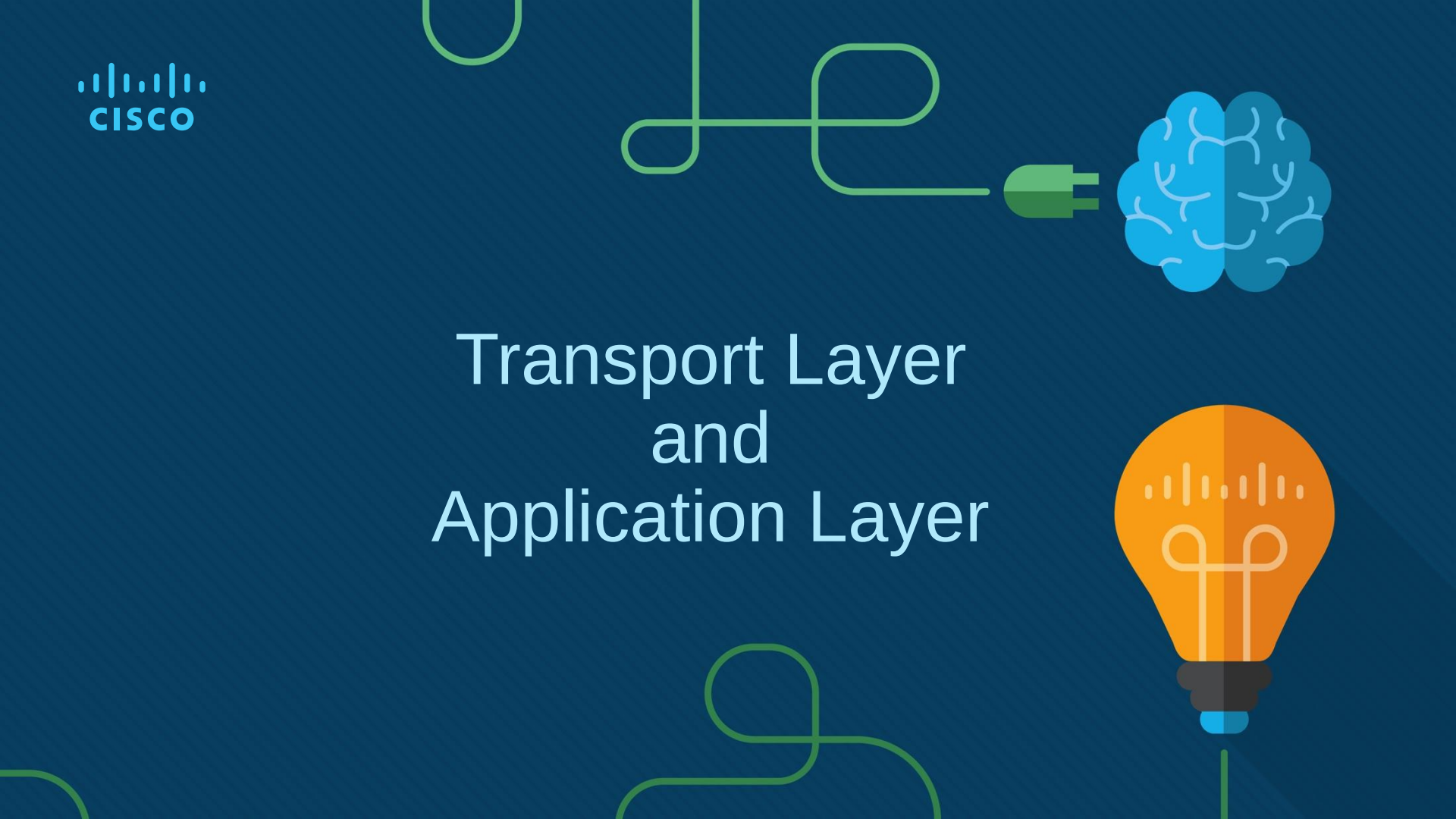
Introduction to Networking

CTO43-3-1 Version VD1

Transport Layer and Application Layer



Transport Layer and Application Layer

Topics and Structure of the lesson

Topic Title	Topic Objective
Transportation of Data	Explain the purpose of the transport layer in managing the transportation of data in end-to-end communication.
TCP Overview	Explain characteristics of TCP.
UDP Overview	Explain characteristics of UDP.
Port Numbers	Explain how TCP and UDP use port numbers.
TCP Communication Process	Explain how TCP session establishment and termination processes facilitate reliable communication.
Reliability and Flow Control	Explain how TCP protocol data units are transmitted and acknowledged to guarantee delivery.
UDP Communication	Compare the operations of transport layer protocols in supporting end-to-end communication.
Application, Presentation, and Session	Explain how the functions of the application layer, presentation layer, and session layer work together to provide network services to end user applications.
Peer-to-Peer	Explain how end user applications operate in a peer-to-peer network.
Web and Email Protocols	Explain how web and email protocols operate.
IP Addressing Services	Explain how DNS and DHCP operate.
File Sharing Services	Explain how file transfer protocols operate.

Key Terms You Must Be Able To Use

If you have mastered this topic, you should be able to use the following terms correctly in your exams:

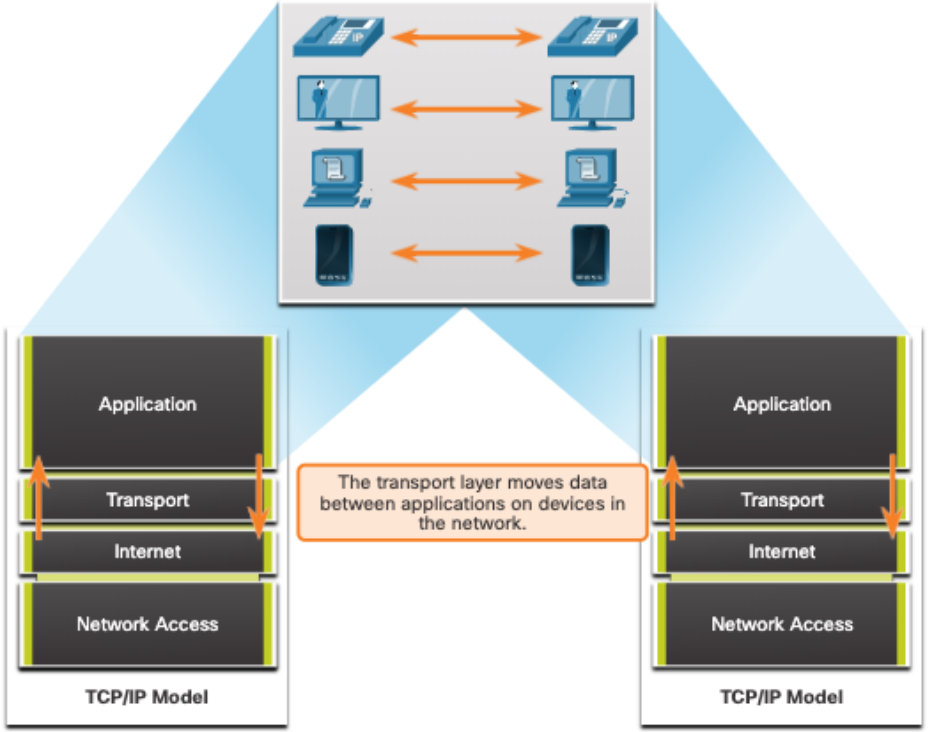
- **Transportation of Data**
- **TCP/UDP Overview/TCP Features/Header/UDP Features/Header**
- **Port Numbers and Group**
- **Multiple Separate Communications**
- **TCP Communication Process/Reliability and Flow Control**
- **UDP Communication Process**
- **Application, Presentation and Session**
- **TCP/IP Application Layer Protocols**
- **Peer-to-Peer**
- **Web and Email Protocols: HTTP and HTTPs**
- **Email Protocols: SMTP, POP, IMAP**
- **Addressing Services: Domain Name Service (DNS),DNS Message Format/ Hierarchy Dynamic Host Configuration Protocol (DHCP)**
- **File Sharing Services**
- **Server Message Block (SMB)**

Transportation of Data

Role of the Transport Layer

The transport layer is:

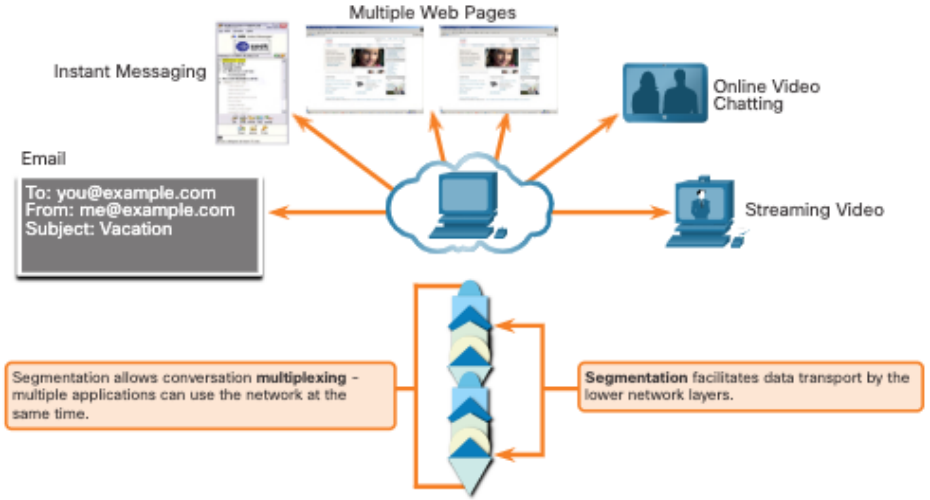
- Responsible for logical communications between applications running on different hosts.
- The link between the application layer and the lower layers that are responsible for network transmission.



Transport Layer Responsibilities

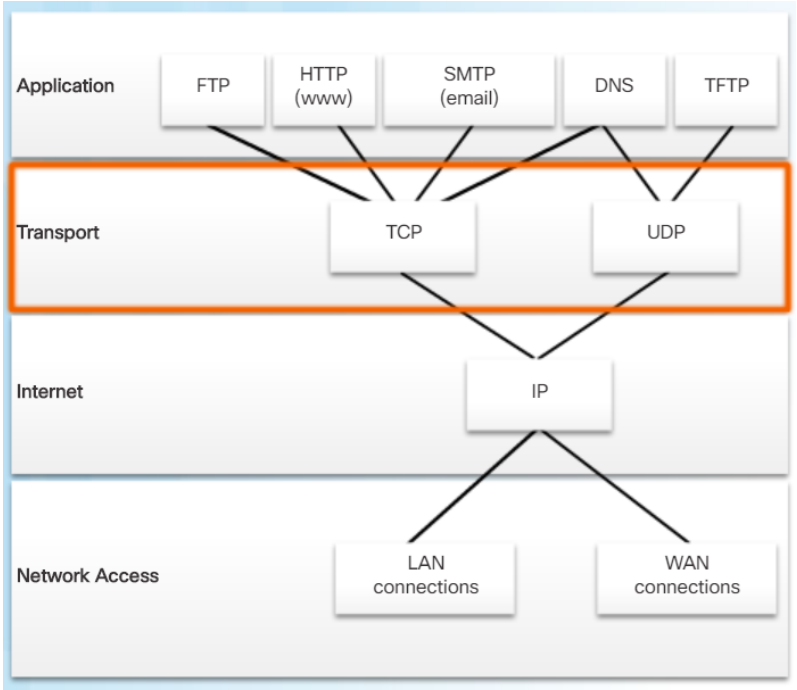
The transport layer has the following responsibilities:

- Tracking individual conversations
- Segmenting data and reassembling segments
- Adds header information
- Identify, separate, and manage multiple conversations
- Uses segmentation and multiplexing to enable different communication conversations to be interleaved on the same network



Transport Layer Protocols

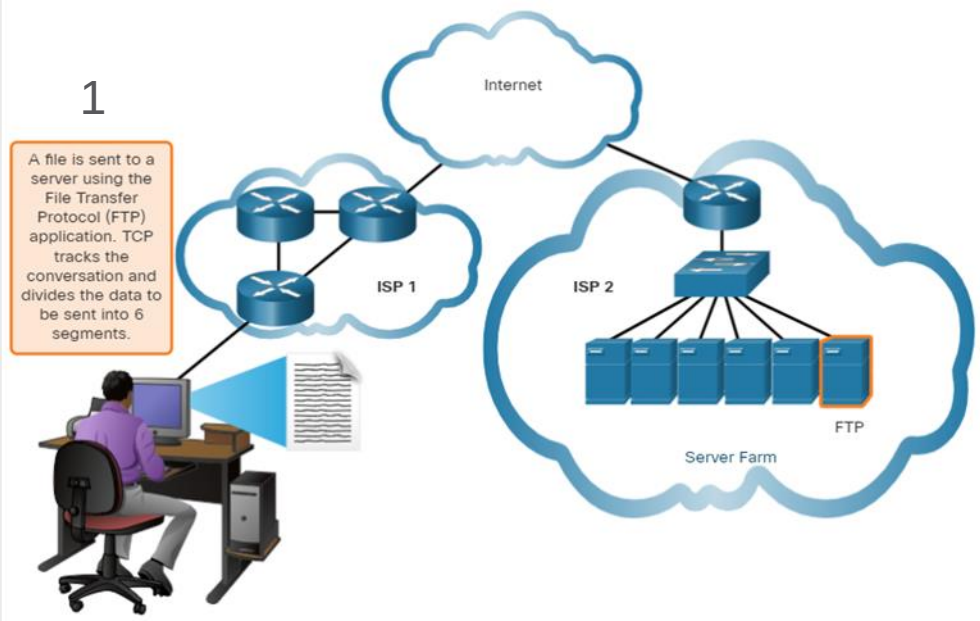
- IP does not specify how the delivery or transportation of the packets takes place.
- Transport layer protocols specify how to transfer messages between hosts, and are responsible for managing reliability requirements of a conversation.
- The transport layer includes the TCP and UDP protocols.



Transmission Control Protocol

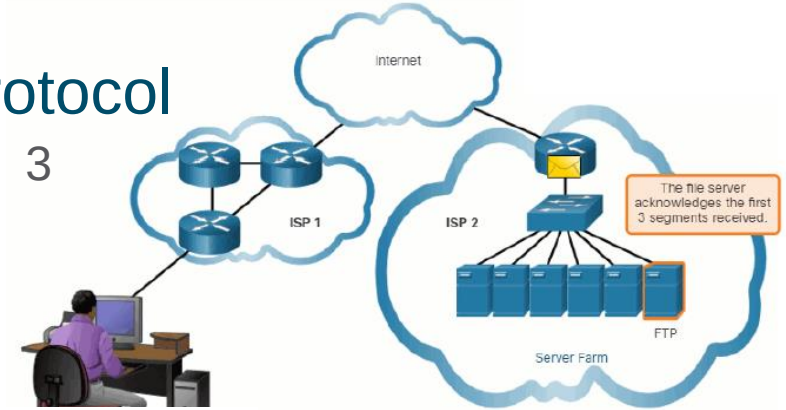
TCP provides reliability and flow control. TCP basic operations:

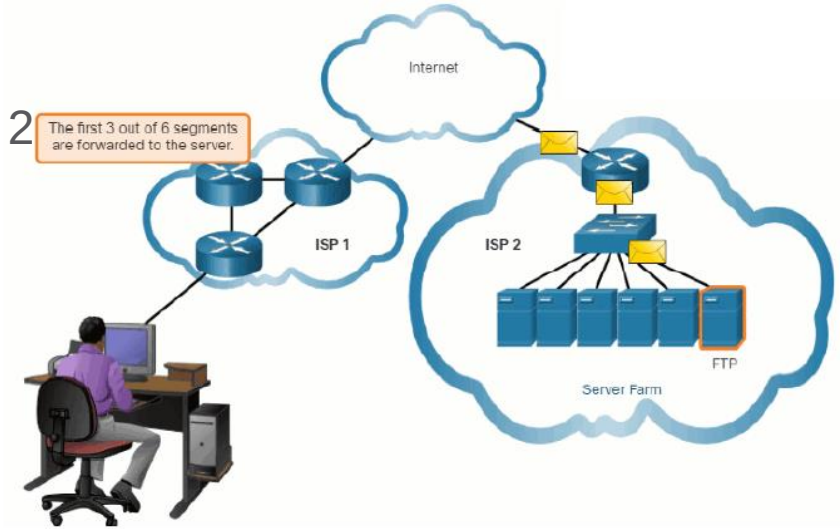
- Number and track data segments transmitted to a specific host from a specific application
- Acknowledge received data
- Retransmit any unacknowledged data after a certain amount of time
- Sequence data that might arrive in wrong order
- Send data at an efficient rate that is acceptable by the receiver

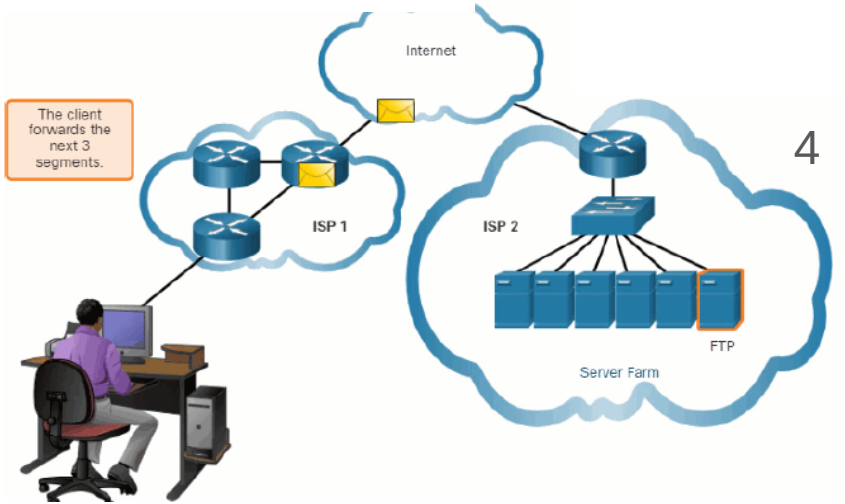


Transportation of Data

Transmission Control Protocol

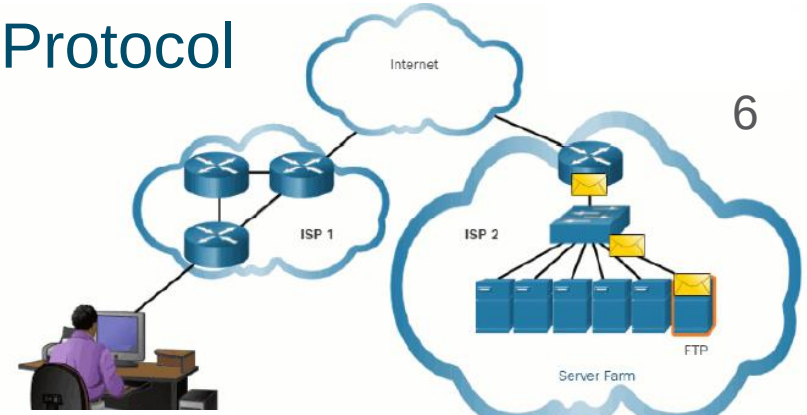


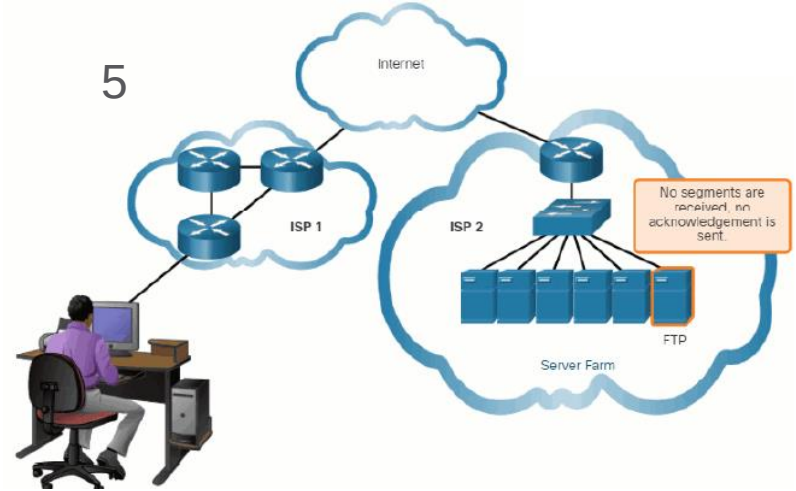


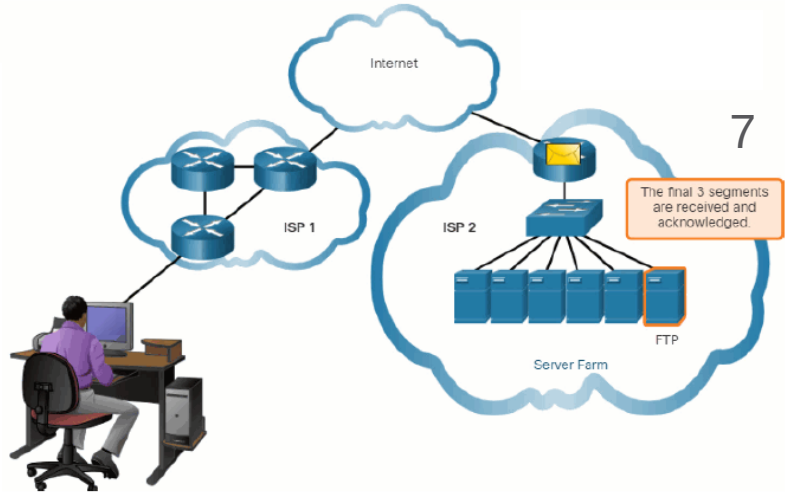


Transportation of Data

Transmission Control Protocol



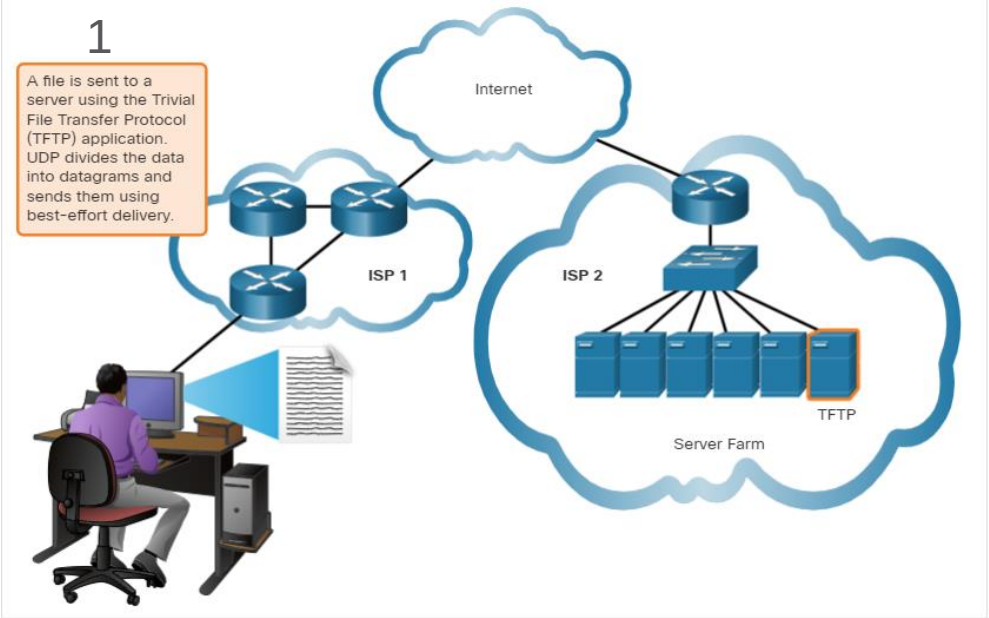




User Datagram Protocol (UDP)

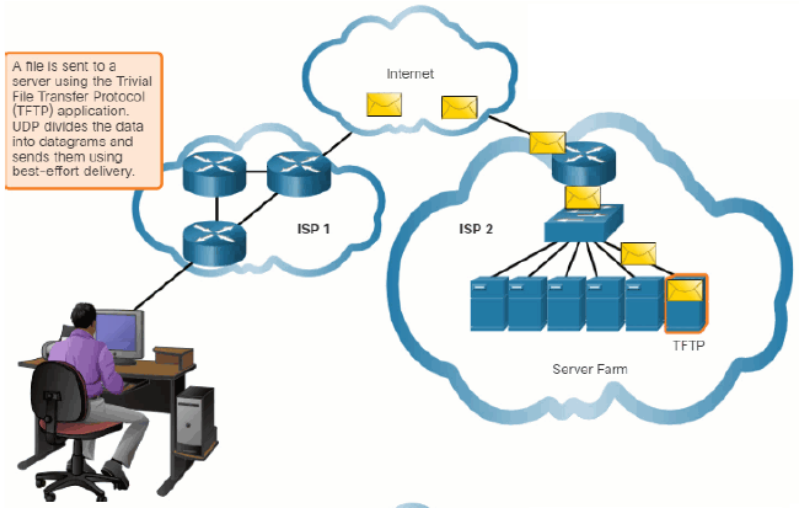
UDP provides the basic functions for delivering datagrams between the appropriate applications, with very little overhead and data checking.

- UDP is a connectionless protocol.
- UDP is known as a best-effort delivery protocol because there is no acknowledgment that the data is received at the destination.

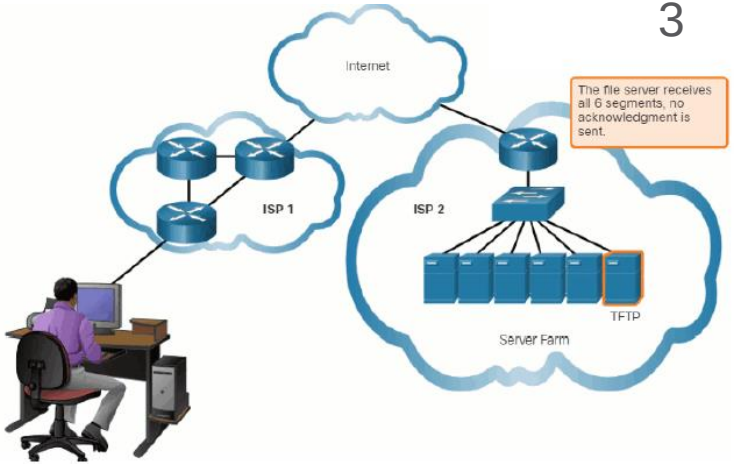


User Datagram Protocol (UDP)

2



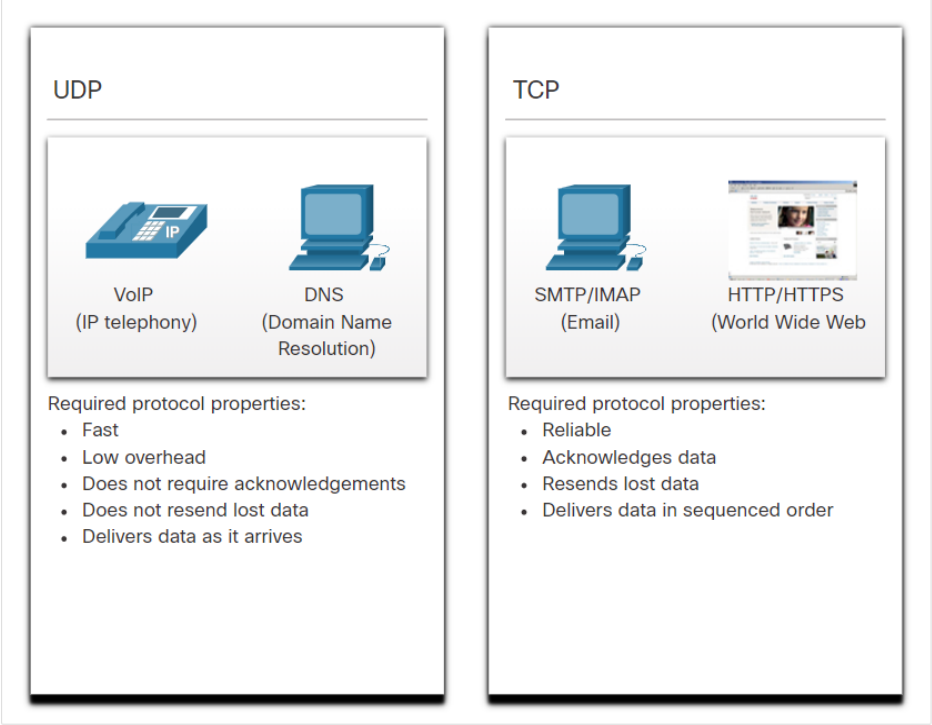
3



The Right Transport Layer Protocol for the Right Application

UDP is also used by request-and-reply applications where the data is minimal, and retransmission can be done quickly.

If it is important that all the data arrives and that it can be processed in its proper sequence, TCP is used as the transport protocol.



TCP Overview

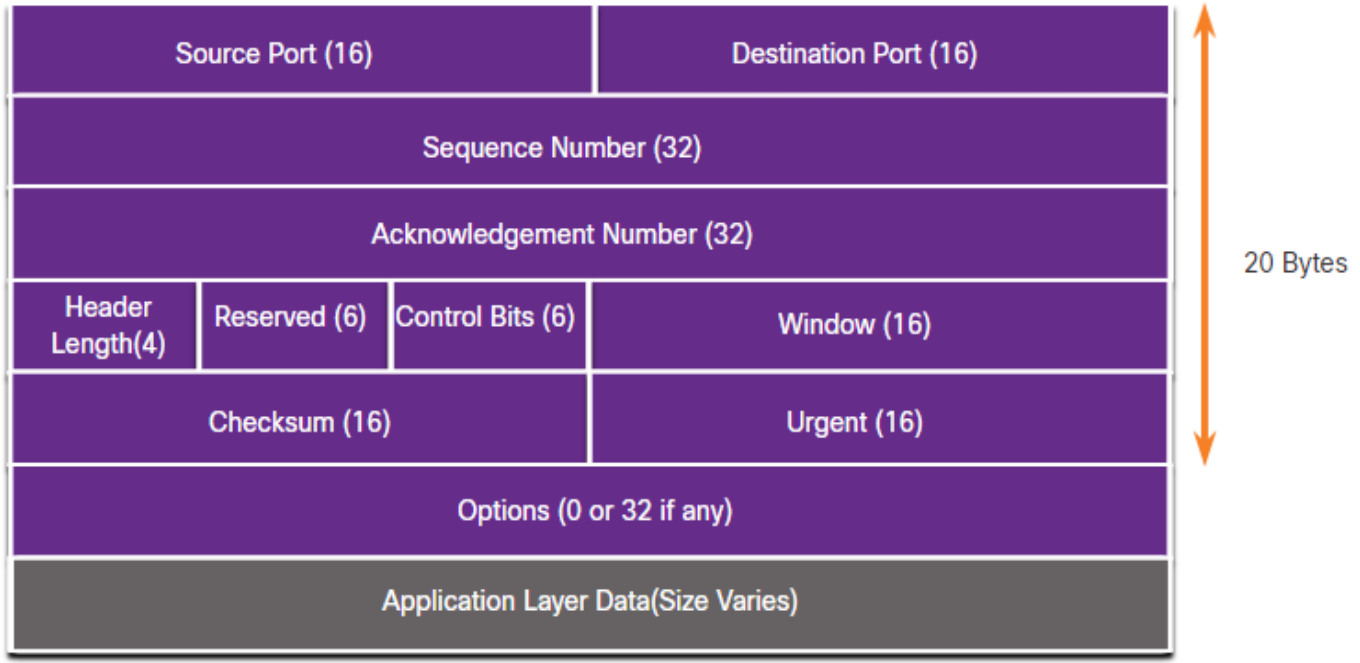
TCP Features

- **Establishes a Session** - TCP is a connection-oriented protocol that negotiates and establishes a permanent connection (or session) between source and destination devices prior to forwarding any traffic.
- **Ensures Reliable Delivery** - For many reasons, it is possible for a segment to become corrupted or lost completely, as it is transmitted over the network. TCP ensures that each segment that is sent by the source arrives at the destination.
- **Provides Same-Order Delivery** - Because networks may provide multiple routes that can have different transmission rates, data can arrive in the wrong order.
- **Supports Flow Control** - Network hosts have limited resources (i.e., memory and processing power). When TCP is aware that these resources are overtaxed, it can request that the sending application reduce the rate of data flow.

TCP Header

TCP is a stateful protocol which means it keeps track of the state of the communication session.

TCP records which information it has sent, and which information has been acknowledged.

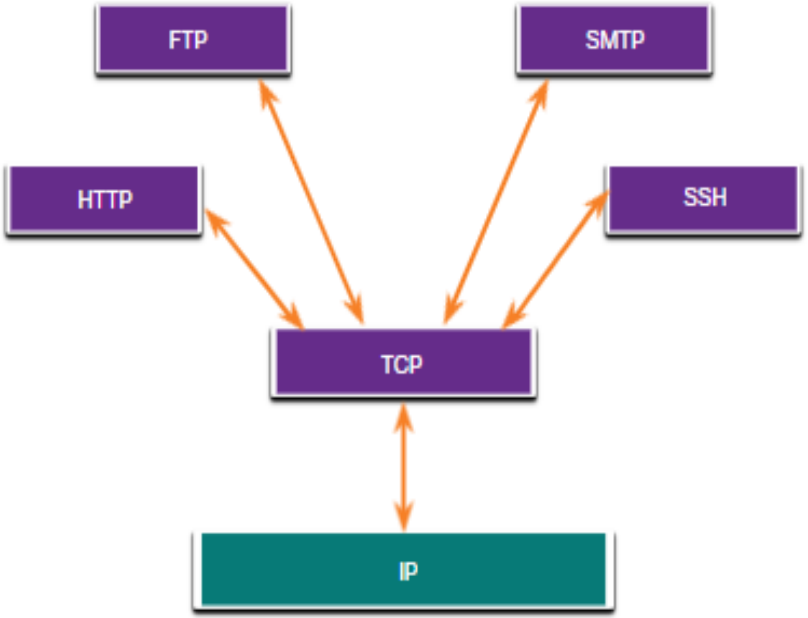


TCP Header Fields

TCP Header Field	Description
Source Port	A 16-bit field used to identify the source application by port number.
Destination Port	A 16-bit field used to identify the destination application by port number.
Sequence Number	A 32-bit field used for data reassembly purposes.
Acknowledgment Number	A 32-bit field used to indicate that data has been received and the next byte expected from the source.
Header Length	A 4-bit field known as "data offset" that indicates the length of the TCP segment header.
Reserved	A 6-bit field that is reserved for future use.
Control bits	A 6-bit field used that includes bit codes, or flags, which indicate the purpose and function of the TCP segment.
Window size	A 16-bit field used to indicate the number of bytes that can be accepted at one time.
Checksum	A 16-bit field used for error checking of the segment header and data.
Urgent	A 16-bit field used to indicate if the contained data is urgent.

Applications that use TCP

TCP handles all tasks associated with dividing the data stream into segments, providing reliability, controlling data flow, and reordering segments.



UDP Overview

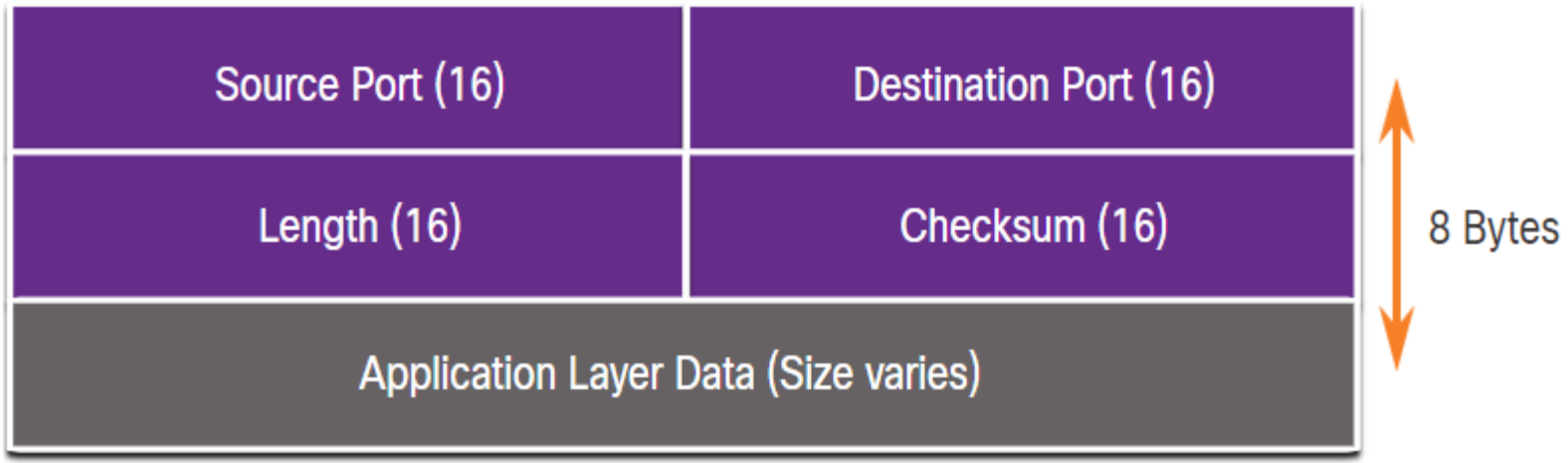
UDP Features

UDP features include the following:

- Data is reconstructed in the order that it is received.
- Any segments that are lost are not resent.
- There is no session establishment.
- The sender is not informed about resource availability.

UDP Header

The UDP header is far simpler than the TCP header because it only has four fields and requires 8 bytes (i.e. 64 bits).



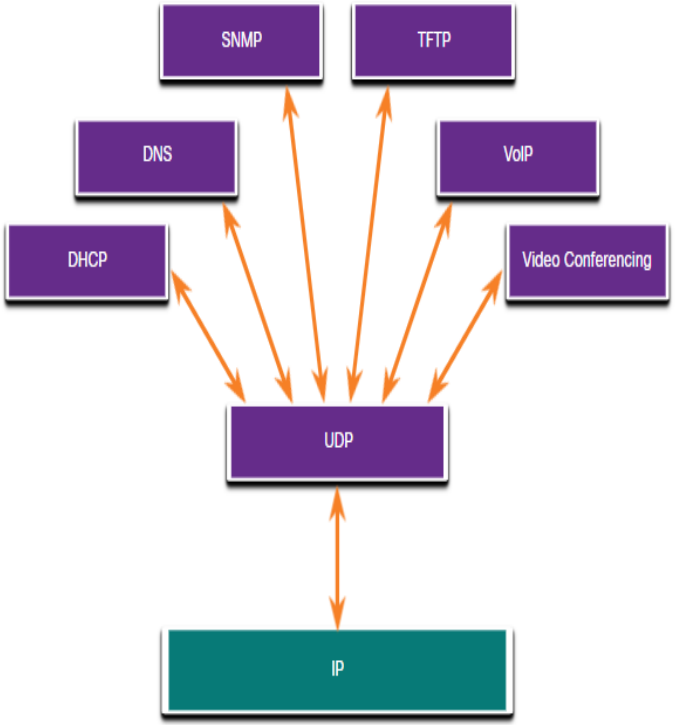
UDP Header Fields

The table identifies and describes the four fields in a UDP header.

UDP Header Field	Description
Source Port	A 16-bit field used to identify the source application by port number.
Destination Port	A 16-bit field used to identify the destination application by port number.
Length	A 16-bit field that indicates the length of the UDP datagram header.
Checksum	A 16-bit field used for error checking of the datagram header and data.

Applications that use UDP

- Live video and multimedia applications - These applications can tolerate some data loss but require little or no delay. Examples include VoIP and live streaming video.
- Simple request and reply applications - Applications with simple transactions where a host sends a request and may or may not receive a reply. Examples include DNS and DHCP.
- Applications that handle reliability themselves - Unidirectional communications where flow control, error detection, acknowledgments, and error recovery is not required, or can be handled by the application. Examples include SNMP and TFTP.



Port Numbers

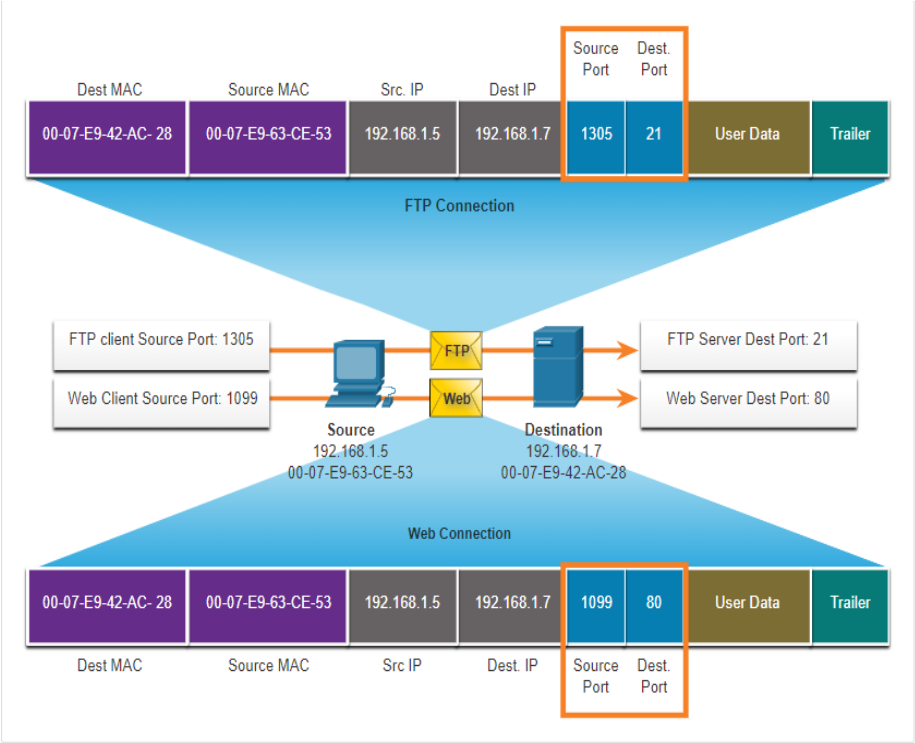
Multiple Separate Communications

TCP and UDP transport layer protocols use port numbers to manage multiple, simultaneous conversations.

The source port number is associated with the originating application on the local host whereas the destination port number is associated with the destination application on the remote host.

Socket Pairs

- The source and destination ports are placed within the segment.
- The segments are then encapsulated within an IP packet.
- The combination of the source IP address and source port number, or the destination IP address and destination port number is known as a socket.
- Sockets enable multiple processes, running on a client, to distinguish themselves from each other, and multiple connections to a server process to be distinguished from each other.



Port Number Groups

Port Group	Number Range	Description
Well-known Ports	**0 to 1,023**	•These port numbers are reserved for common or popular services and applications such as web browsers, email clients, and remote access clients. •Defined well-known ports for common server applications enables clients to easily identify the associated service required.
Registered Ports	**1,024 to 49,151**	•These port numbers are assigned by IANA to a requesting entity to use with specific processes or applications. •These processes are primarily individual applications that a user has chosen to install, rather than common applications that would receive a well-known port number. •For example, Cisco has registered port 1812 for its RADIUS server authentication process.
Private and/or **Dynamic Ports**	**49,152 to 65,535**	•These ports are also known as *ephemeral ports*. •The client's OS usually assign port numbers dynamically when a connection to a service is initiated. •The dynamic port is then used to identify the client application during communication.

Port Number Groups (Cont.)

Well-Known Port Numbers

Port Number	Protocol	Application
20	TCP	File Transfer Protocol (FTP) - Data
21	TCP	File Transfer Protocol (FTP) - Control
22	TCP	Secure Shell (SSH)
23	TCP	Telnet
25	TCP	Simple Mail Transfer Protocol (SMTP)
53	UDP, TCP	Domain Name Service (DNS)
67	UDP	Dynamic Host Configuration Protocol (DHCP) - Server
68	UDP	Dynamic Host Configuration Protocol - Client
69	UDP	Trivial File Transfer Protocol (TFTP)
80	TCP	Hypertext Transfer Protocol (HTTP)
110	TCP	Post Office Protocol version 3 (POP3)
143	TCP	Internet Message Access Protocol (IMAP)
161	UDP	Simple Network Management Protocol (SNMP)
443	TCP	Hypertext Transfer Protocol Secure (HTTPS)

Port Numbers

The netstat Command

Unexplained TCP connections can pose a major security threat. Netstat is an important tool to verify connections.

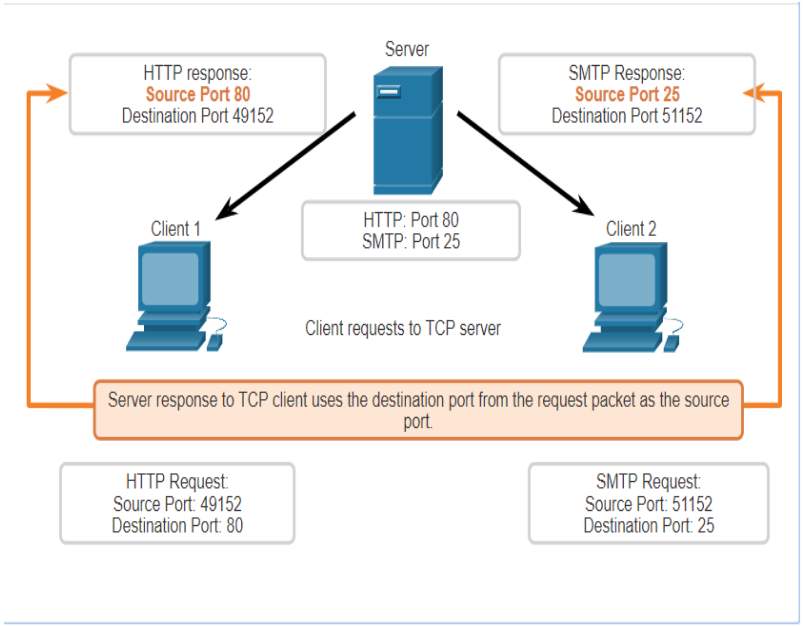
```
C:\> netstat
Active Connections
Proto Local Address          Foreign Address                State
TCP   192.168.1.124:3126     192.168.0.2:netbios-ssn        ESTABLISHED
TCP   192.168.1.124:3158     207.138.126.152:http           ESTABLISHED
TCP   192.168.1.124:3159     207.138.126.169:http           ESTABLISHED
TCP   192.168.1.124:3160     207.138.126.169:http           ESTABLISHED
TCP   192.168.1.124:3161     sc.msn.com:http                ESTABLISHED
TCP   192.168.1.124:3166     www.cisco.com:http             ESTABLISHED
```

TCP Communication Process

TCP Server Processes

Each application process running on a server is configured to use a port number.

- An individual server cannot have two services assigned to the same port number within the same transport layer services.
- An active server application assigned to a specific port is considered open, which means that the transport layer accepts, and processes segments addressed to that port.
- Any incoming client request addressed to the correct socket is accepted, and the data is passed to the server application.



TCP Connection Establishment

The three way handshake validates that the destination host is available for communication.

Step 1: SYN

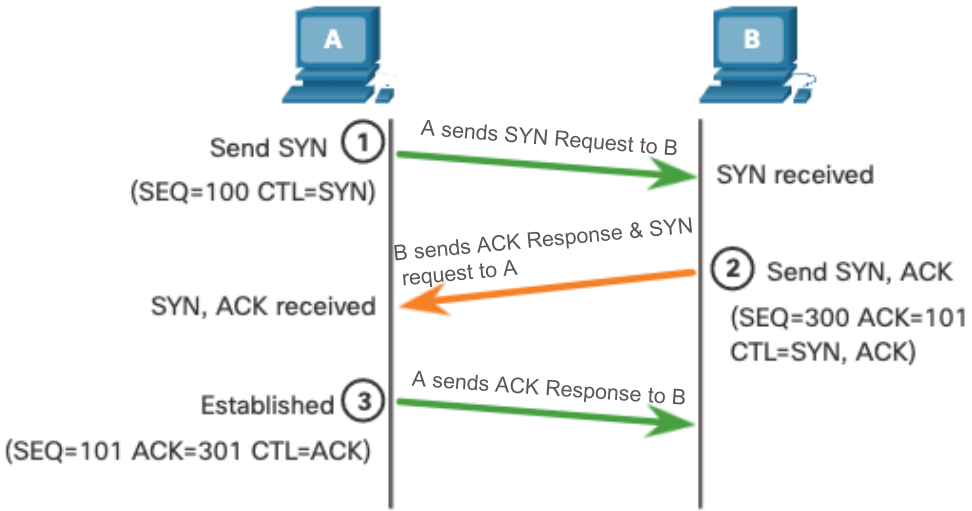
- The initiating client requests a client-to-server communication session with the server.

Step 2: ACK & SYN

- The server acknowledges the client-to-server communication session and requests a server-to-client communication session.

Step 3: ACK

- The initiating client acknowledges the server-to-client communication session.



Session Termination

Step 1: FIN

- When the client has no more data to send in the stream, it sends a segment with the FIN flag set.

Step 2: ACK

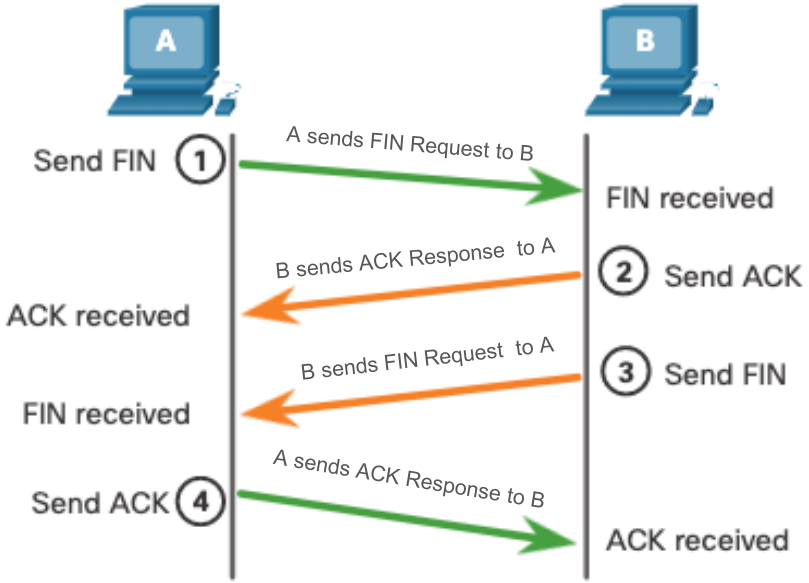
- The server sends an ACK to acknowledge the receipt of the FIN to terminate the session from client to server.

Step 3: FIN

- The server sends a FIN to the client to terminate the server-to-client session.

Step 4: ACK

- The client responds with an ACK to acknowledge the FIN from the server.



TCP Three-Way Handshake Analysis

Functions of the Three-Way Handshake:

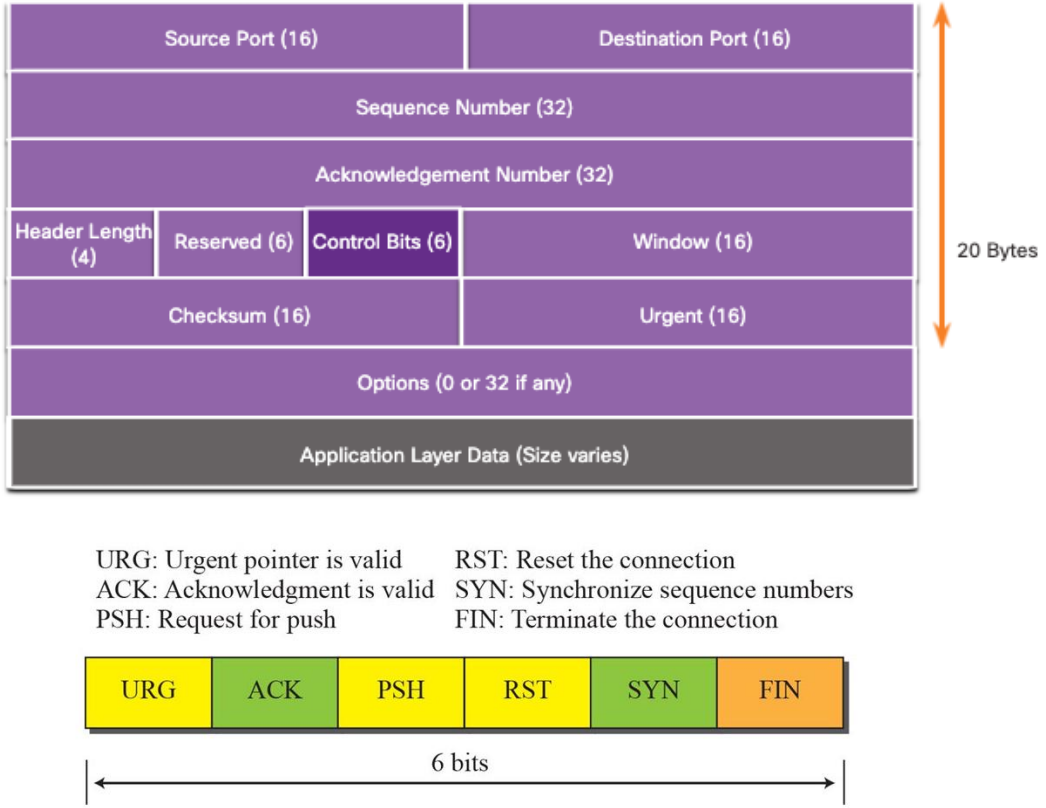
- It establishes that the destination device is present on the network.
- It verifies that the destination device has an active service and is accepting requests on the destination port number that the initiating client intends to use.
- It informs the destination device that the source client intends to establish a communication session on that port number.

After the communication is completed the sessions are closed, and the connection is terminated. The connection and session mechanisms enable TCP reliability function.

TCP Three-Way Handshake Analysis (Cont.)

The six control bit flags are as follows:

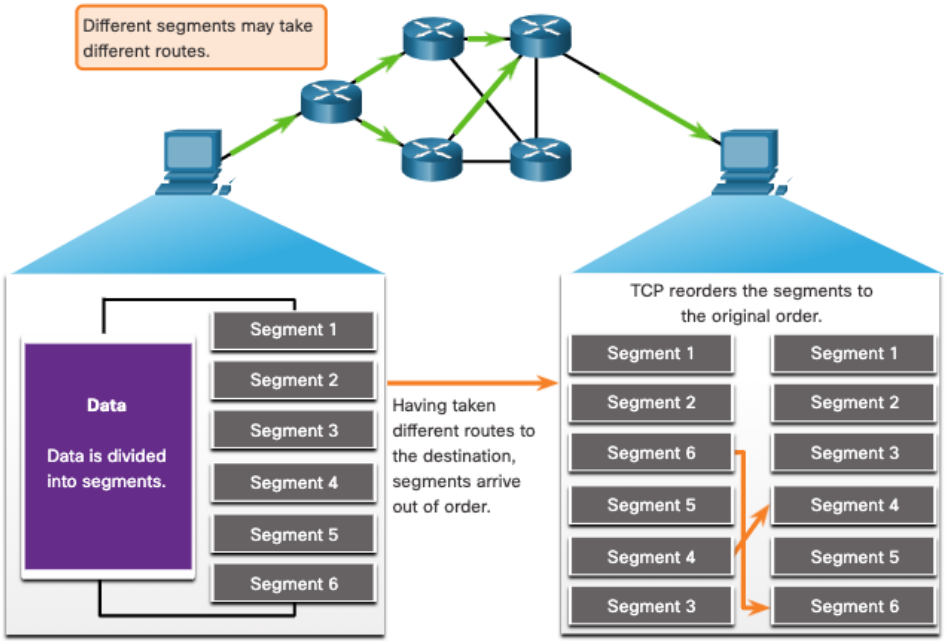
- **URG** - Urgent pointer field indicates that there is data in the segment that the sending-side upper-layer has marked “urgent”.
- **ACK** - Acknowledgment flag used in connection establishment and session termination
- **PSH** - Push function, where the receiver should pass the data to the upper layer immediately
- **RST** - Reset the connection when an error or timeout occurs
- **SYN** - Synchronize sequence numbers used in connection establishment
- **FIN** - No more data from sender and used in session termination



Reliability and Flow Control

TCP Reliability- Guaranteed and Ordered Delivery

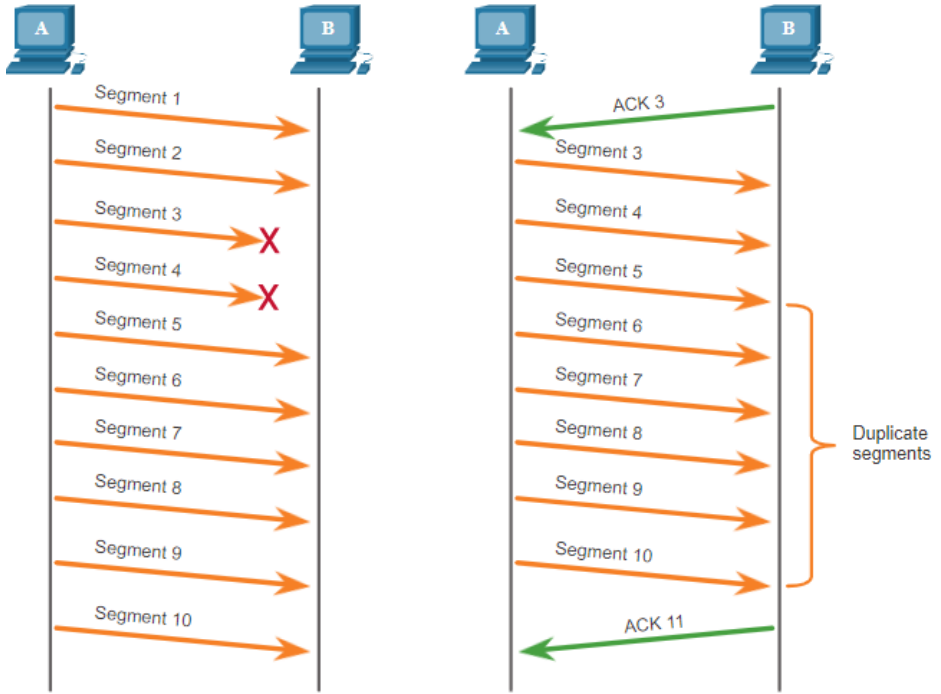
- TCP can also help maintain the flow of packets so that devices do not become overloaded.
- There may be times when TCP segments do not arrive at their destination or arrive out of order.
- All the data must be received and the data in these segments must be reassembled into the original order.
- Sequence numbers are assigned in the header of each packet to achieve this goal.



TCP Reliability – Data Loss and Retransmission

No matter how well designed a network is, data loss occasionally occurs.

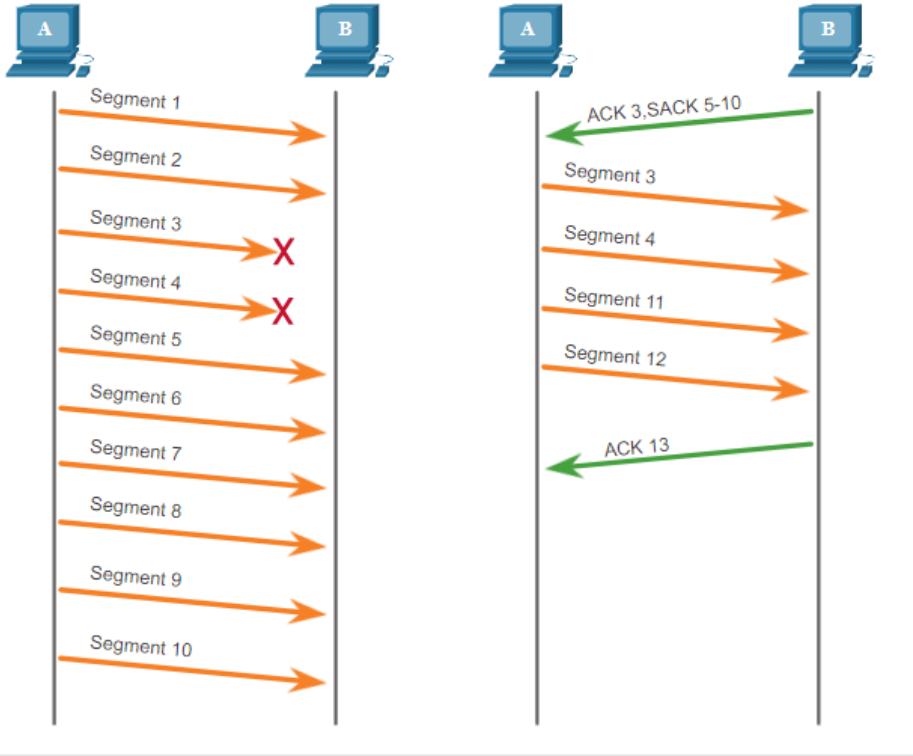
TCP provides methods of managing these segment losses. Among these is a mechanism to retransmit segments for unacknowledged data.



TCP Reliability – Data Loss and Retransmission (Cont.)

Host operating systems today typically employ an optional TCP feature called selective acknowledgment (SACK), negotiated during the three-way handshake.

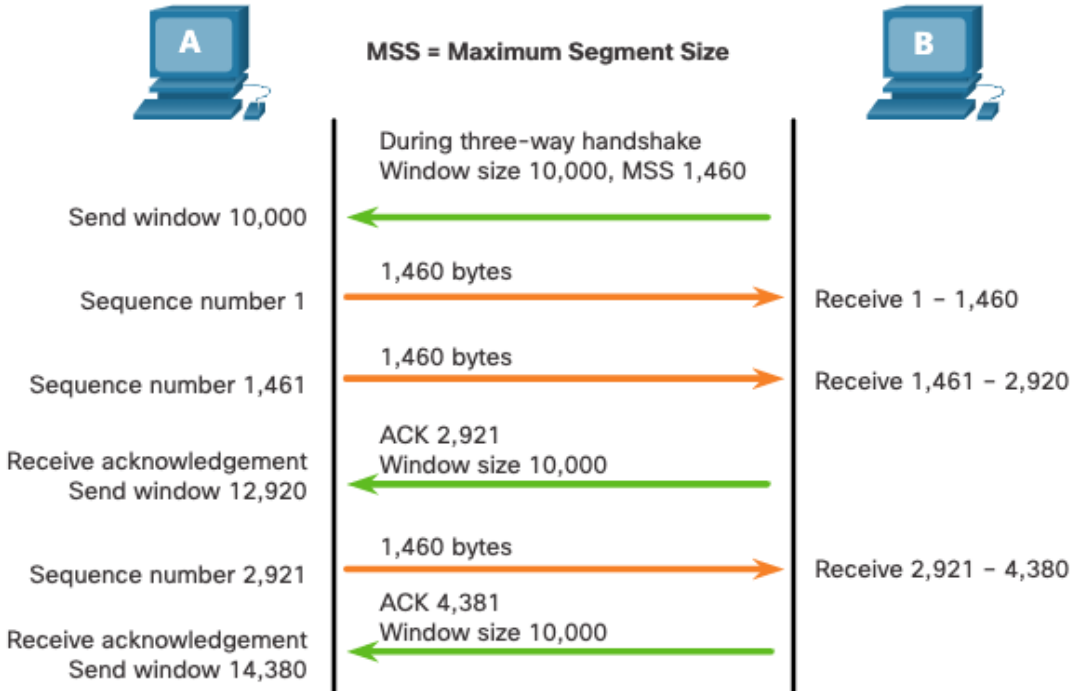
If both hosts support SACK, the receiver can explicitly acknowledge which segments (bytes) were received including any discontinuous segments.



TCP Flow Control – Window Size and Acknowledgments

TCP also provides mechanisms for flow control as follows:

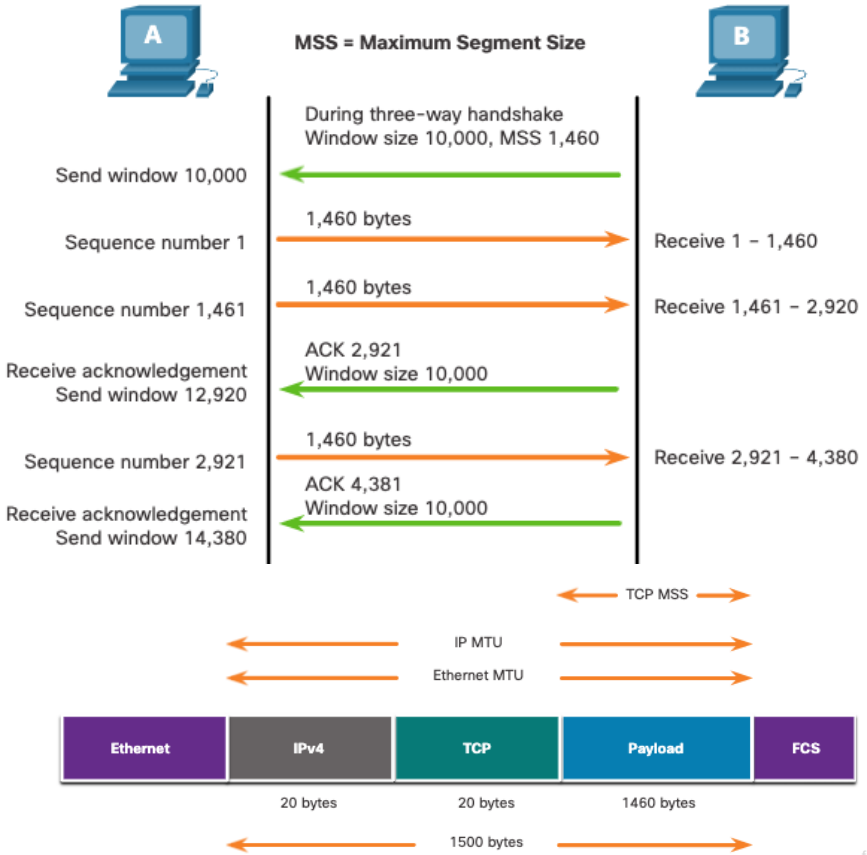
- Flow control is the amount of data that the destination can receive and process reliably.
- Flow control helps maintain the reliability of TCP transmission by adjusting the rate of data flow between source and destination for a given session.



TCP Flow Control – Maximum Segment Size

Maximum Segment Size (MSS) is the maximum amount of data that the destination device can receive.

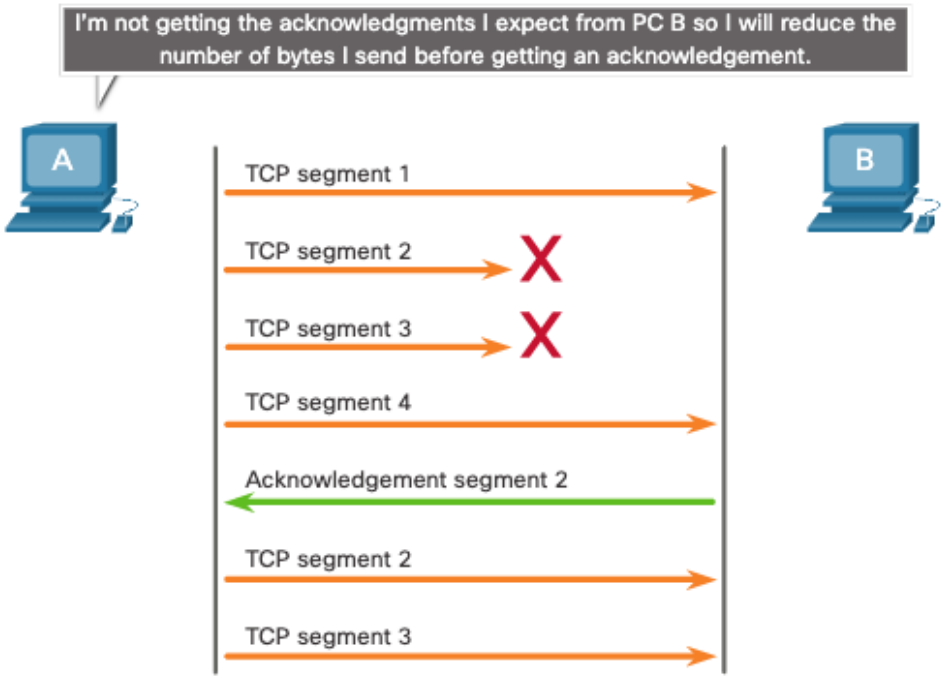
- A common MSS is 1,460 bytes when using IPv4.
- A host determines the value of its MSS field by subtracting the IP and TCP headers from the Ethernet maximum transmission unit (MTU), which is 1500 bytes be default.
- 1500 minus 40 (20 bytes for the IPv4 header and 20 bytes for the TCP header) leaves 1460 bytes.



TCP Flow Control – Congestion Avoidance

When congestion occurs on a network, it results in packets being discarded by the overloaded router.

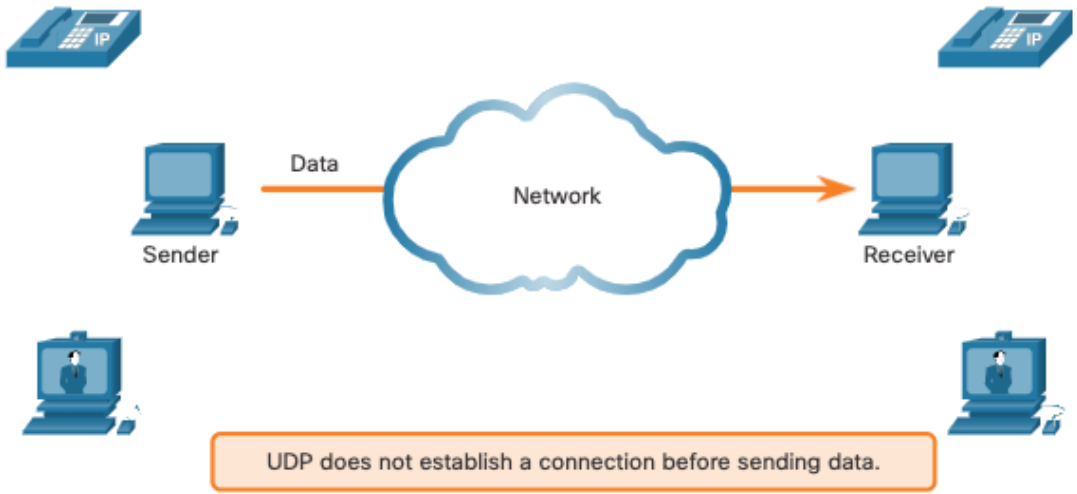
To avoid and control congestion, TCP employs several congestion handling mechanisms, timers, and algorithms.



UDP Communication

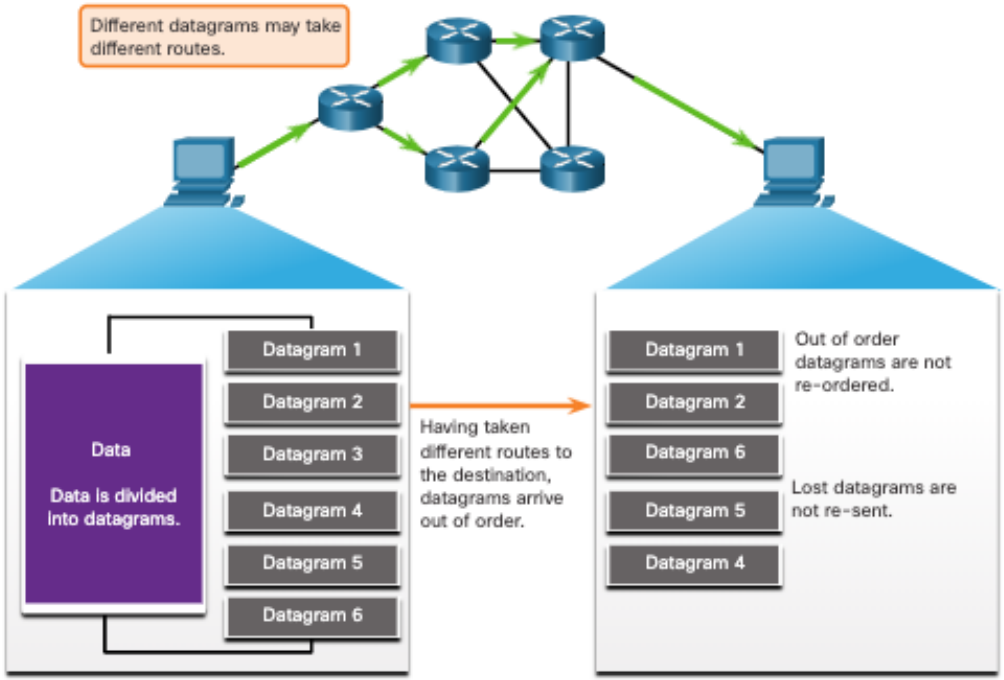
UDP Low Overhead versus Reliability

UDP does not establish a connection. UDP provides low overhead data transport because it has a small datagram header and no network management traffic.



UDP Datagram Reassembly

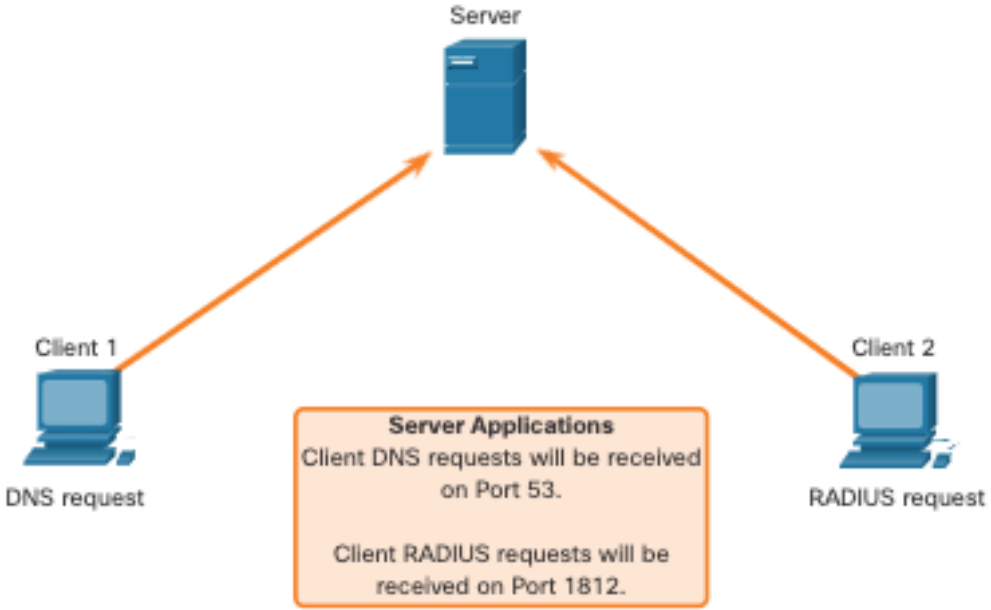
- UDP does not track sequence numbers the way TCP does.
- UDP has no way to reorder the datagrams into their transmission order.
- UDP simply reassembles the data in the order that it was received and forwards it to the application.



UDP Server Processes and Requests

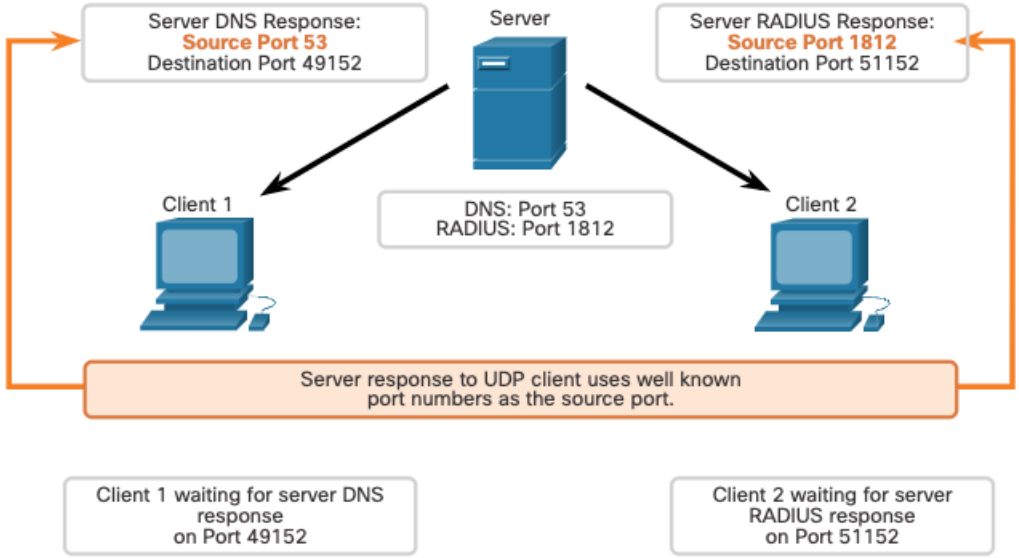
UDP-based server applications are assigned well-known or registered port numbers.

UDP receives a datagram destined for one of these ports, it forwards the application data to the appropriate application based on its port number.



UDP Client Processes

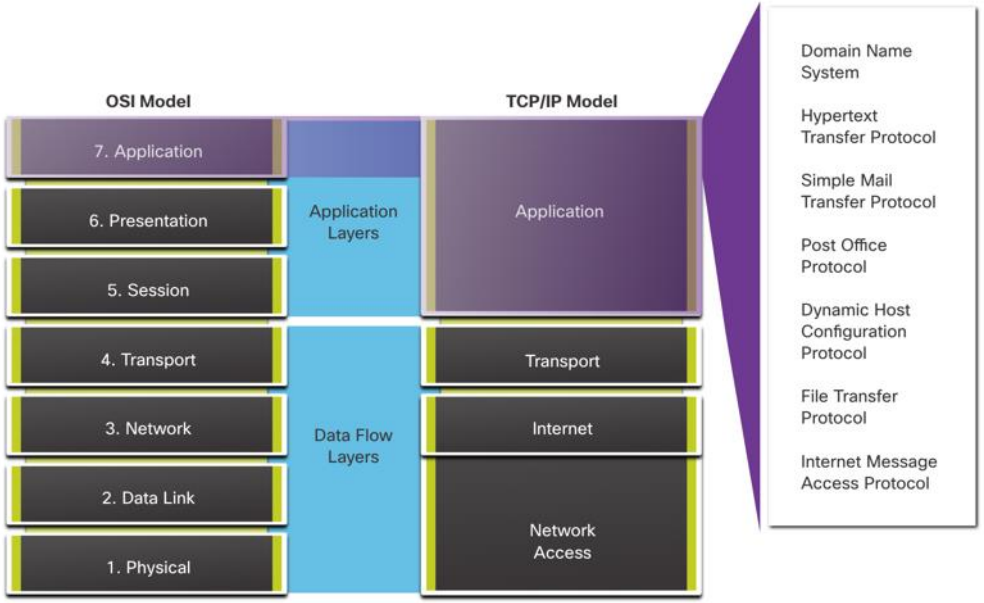
- The UDP client process dynamically selects a port number from the range of port numbers and uses this as the source port for the conversation.
- The destination port is usually the well-known or registered port number assigned to the server process.
- After a client has selected the source and destination ports, the same pair of ports are used in the header of all datagrams in the transaction.



Application, Presentation, and Session

Application Layer

- Application Layer Protocols are used to exchange data between programs running on the source and destination hosts
- 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.



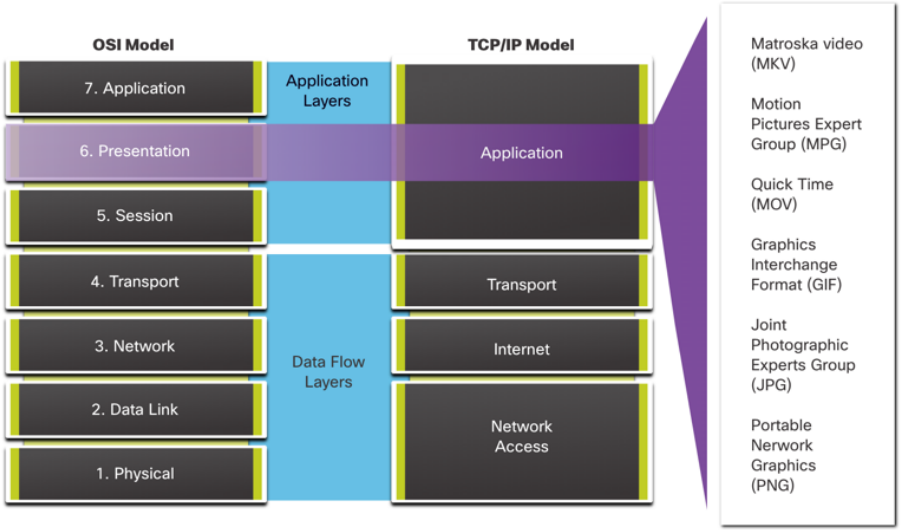
Presentation and Session Layer

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

The session layer functions:

- It establishes, maintains, synchronizes and terminates the connection between communicating systems.
- It creates and maintains dialogs in half duplex or full duplex between source and destination applications.



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 - Domain Name System (or Service)
- TCP, UDP client 53
- Translates domain names, such as cisco.com, into IP addresses.

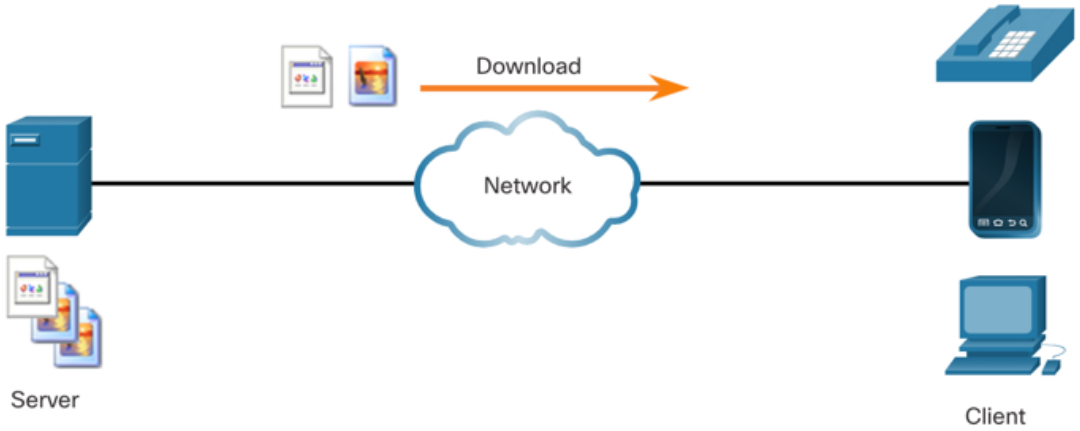
Host Config
DHCP - Dynamic Host Configuration Protocol
- UDP client 68, server 67
- Dynamically assigns IP addresses to be re-used when no longer needed

Web
HTTP - Hypertext Transfer Protocol
- TCP 80, 8080
- A set of rules for exchanging text, graphic images, sound, video, and other multimedia files on the World Wide Web

Peer-to-Peer

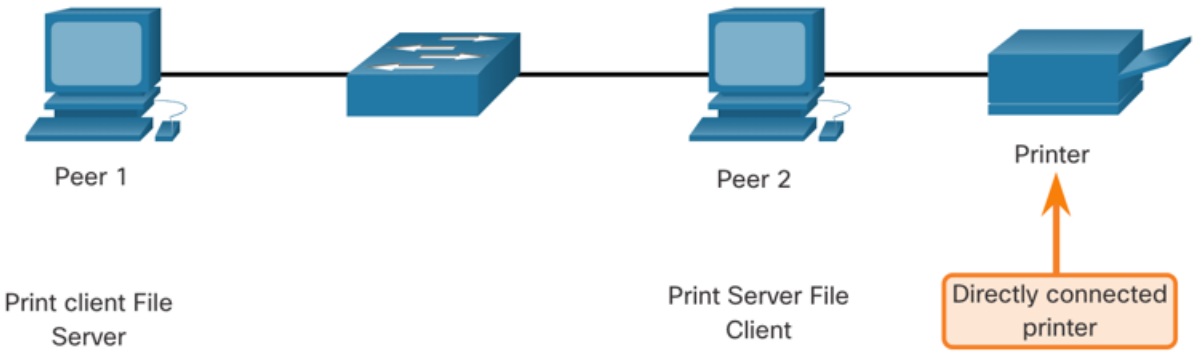
Client-Server Model

- 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.



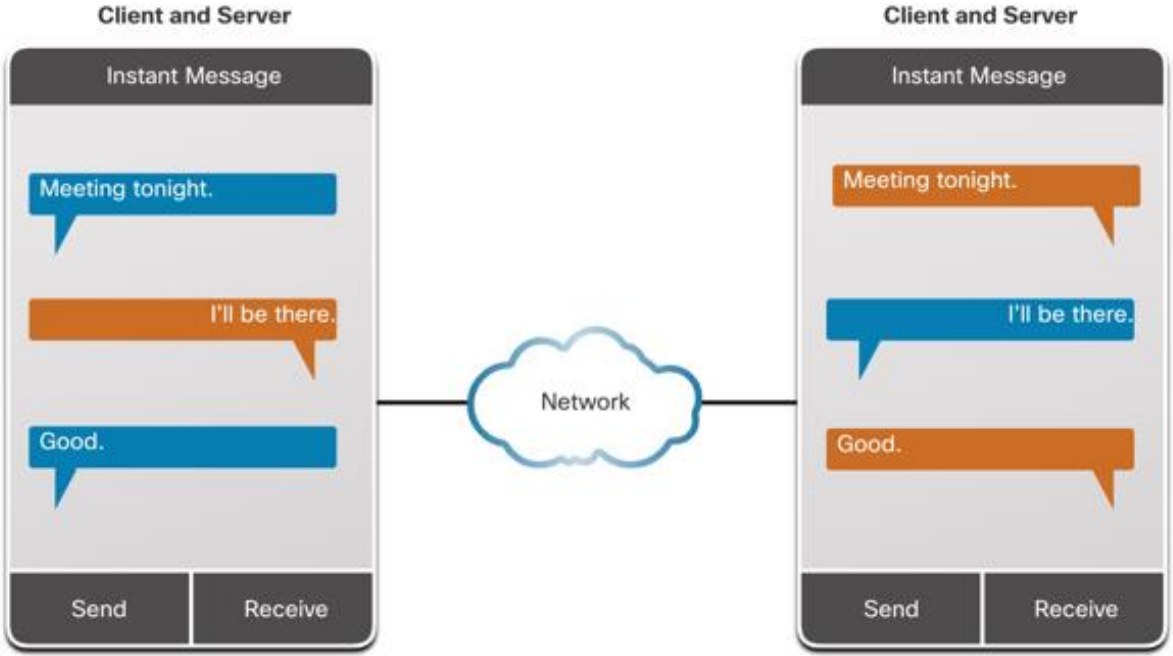
Peer-to-Peer Networks

- 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 (known as a 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.
- Both clients can simultaneously send and receive messages



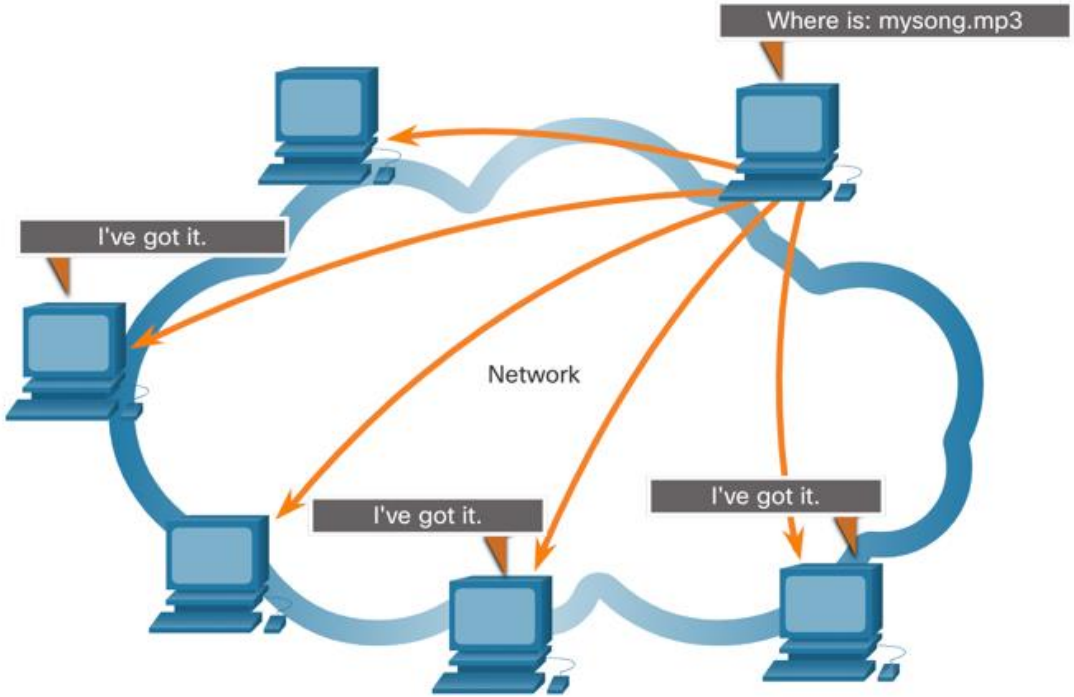
Common P2P Applications

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

- Many Gnutella client applications are available including utorrent, bitcomet, DC++, deluge and emule



Web and Email 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:

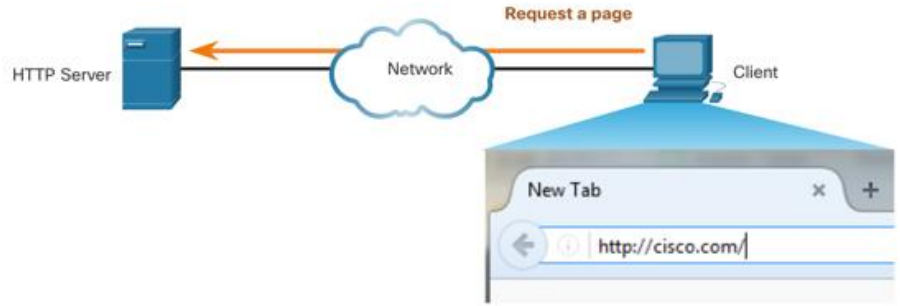
- http (the protocol or scheme)
- www.cisco.com (the server name)
- index.html (the specific filename requested)

Hypertext Transfer Protocol and Hypertext Markup Language (Cont.)

Step 2

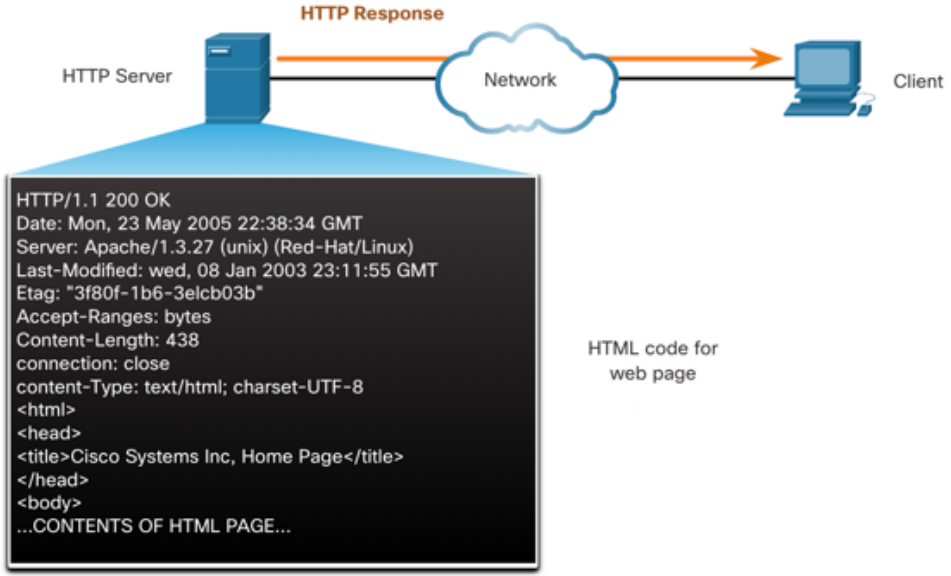
The browser then checks with a name server to convert www.cisco.com into a numeric IP address, which it uses to connect to the server.

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

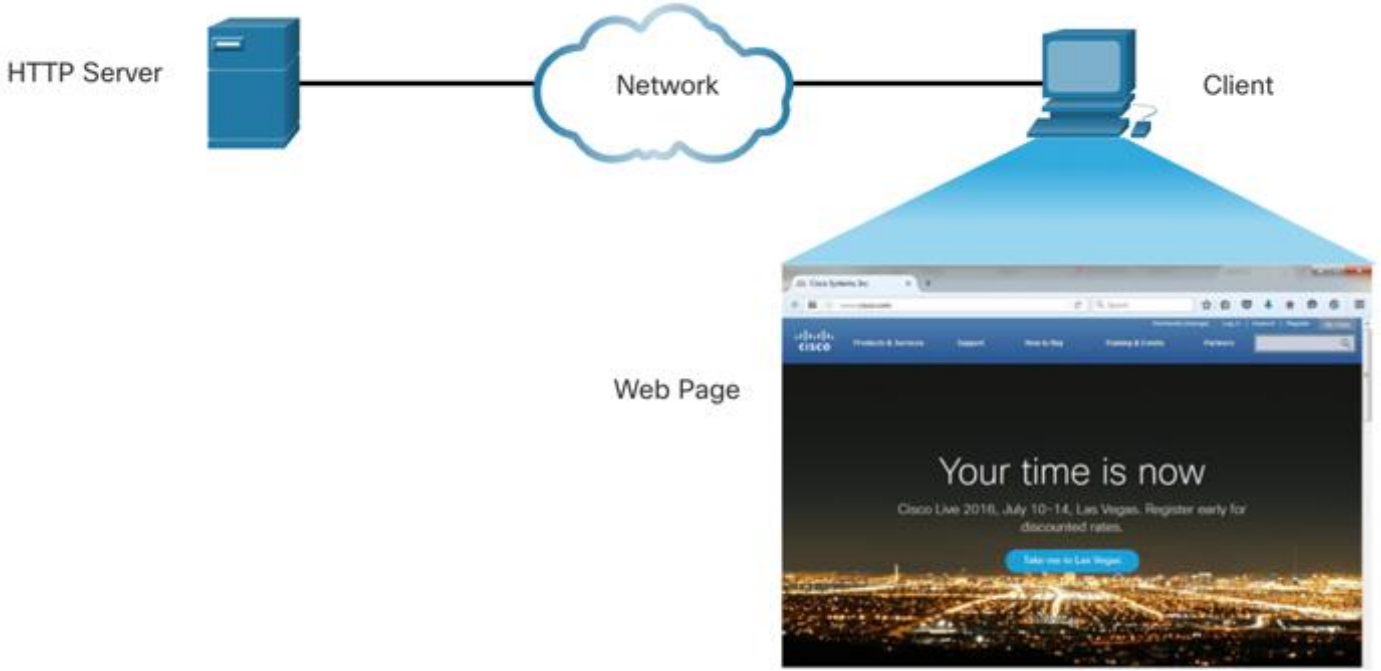
In response to the request, the server sends the HTML code for this web page to the browser.



Hypertext Transfer Protocol and Hypertext Markup Language (Cont.)

Step 4

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

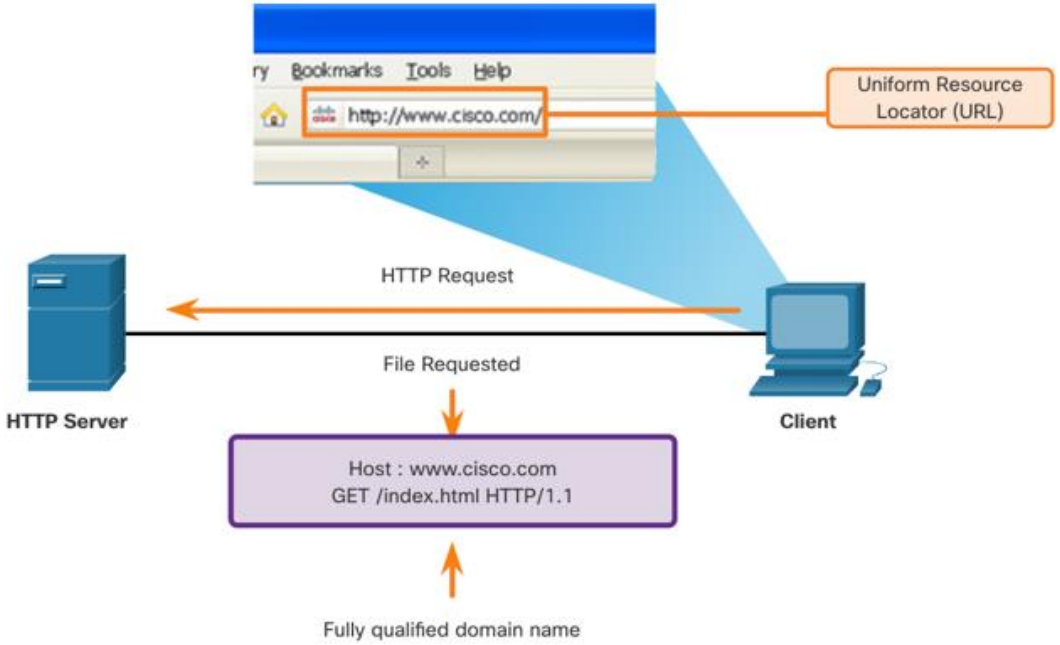


HTTP and HTTPS

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.



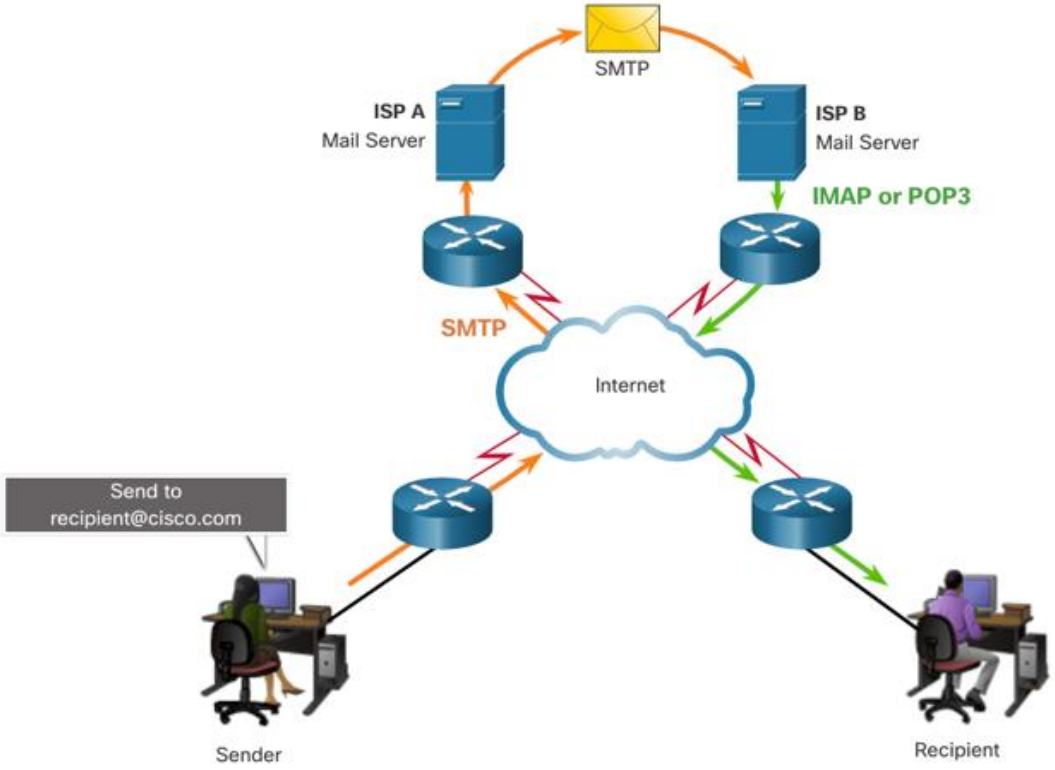
Note: HTTP is not a secure protocol. For secure communications sent across the internet, HTTPS should be used.

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.

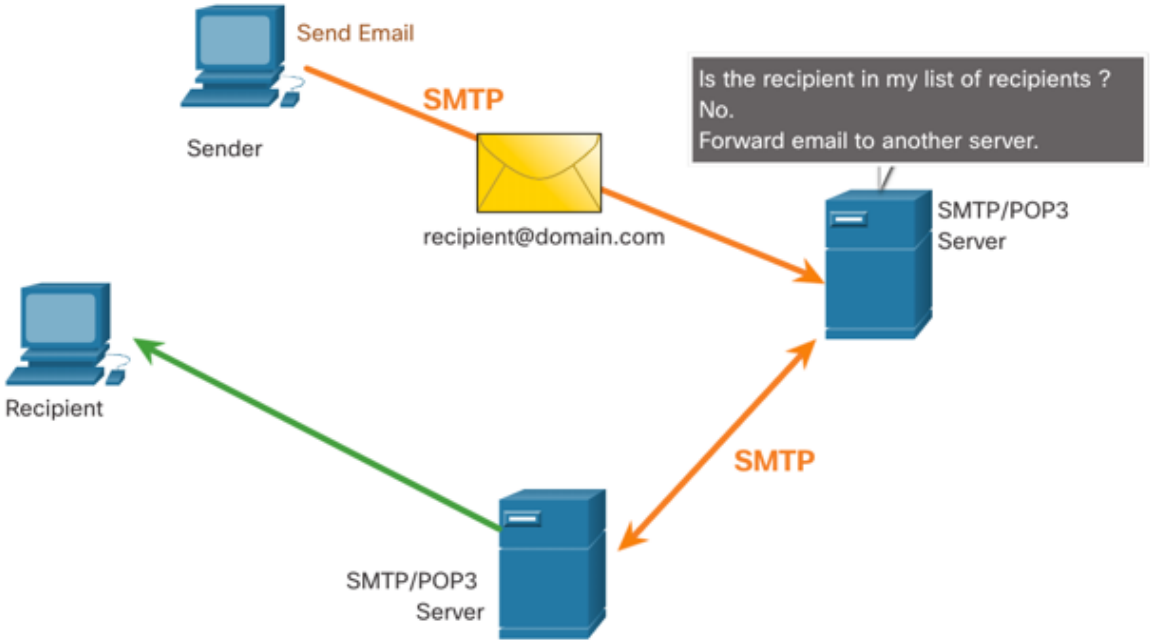
The email protocols used for operation are:

- 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.

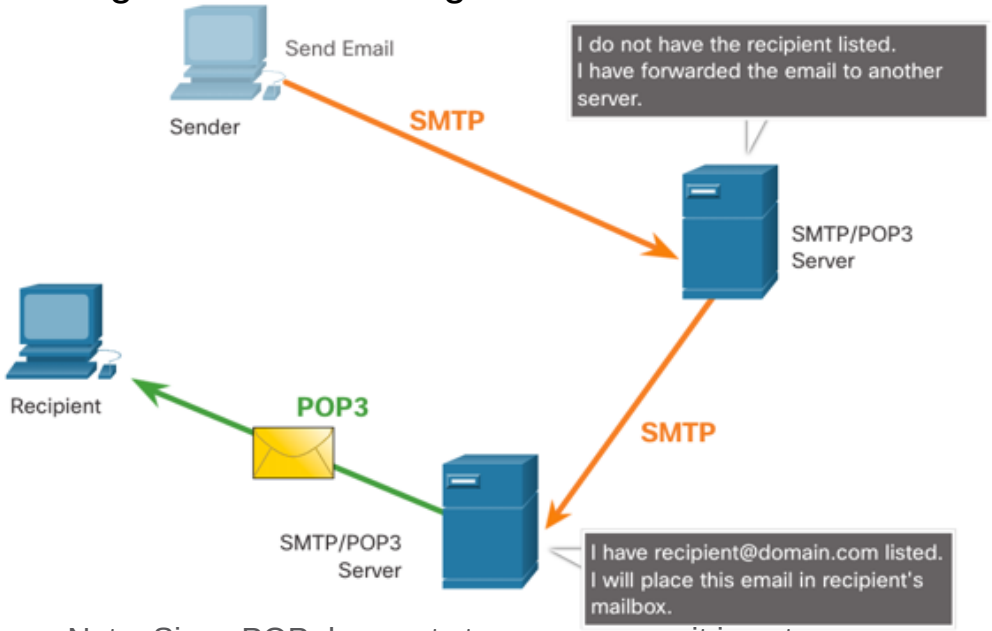


Note: SMTP message formats require a message header (recipient email address & sender email address) and a message body.

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.

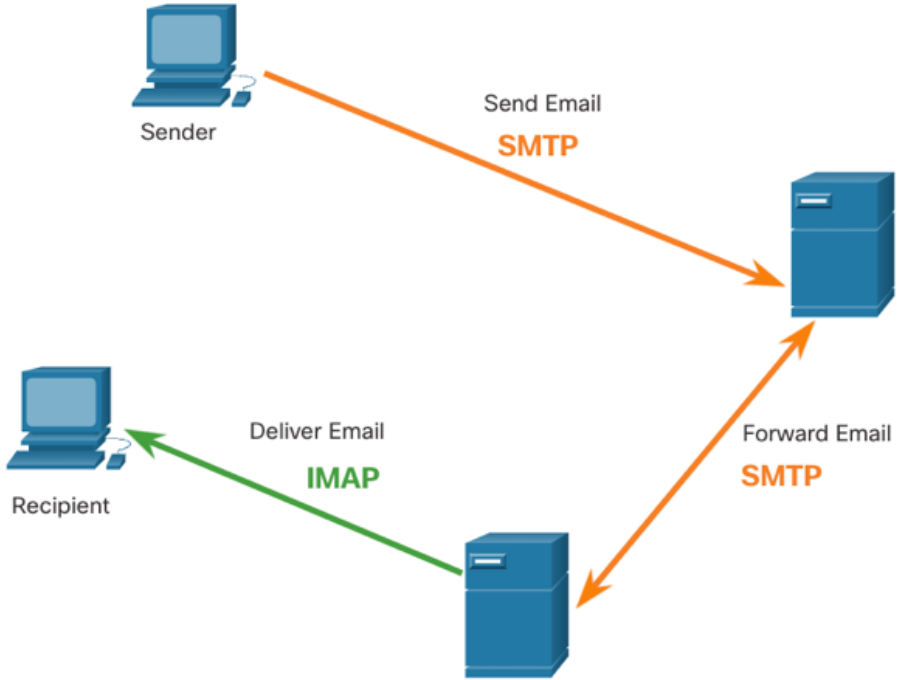


Note: Since POP does not store messages, it is not recommended for small businesses that need a centralized backup solution.

SMTP, POP and IMAP (Cont.)

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

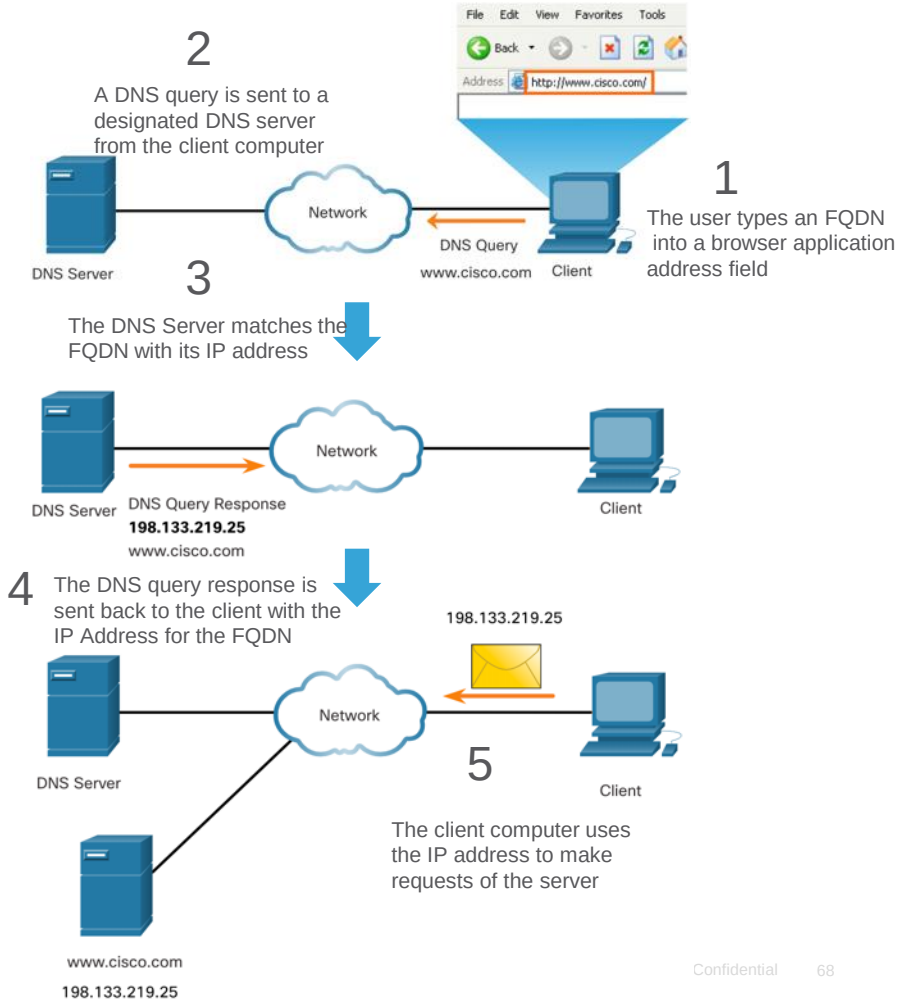
- Unlike POP, when a user connects to an IMAP server, copies of the messages are downloaded to the client application. The original messages are kept on the server until manually deleted.
- When a user decides to delete a message, the server synchronizes that action and deletes the message from the server.



Addressing Services

Domain Name Service

- Domain names were created to convert the numeric IP addresses into a simple, recognizable name.
- Fully-qualified domain names (FQDNs), such as http://www.cisco.com, are much easier for people to remember than 198.133.219.25.
- The DNS protocol defines an automated service that matches resource names with the required numeric network address. It includes the format for queries, responses, and data.



DNS Message Format

The DNS server stores different types of resource records that are used to resolve names. These records contain the name, address, and type of record.

Some of these record types are as follows:

- **A** - An end device IPv4 address
- **NS** - An Authoritative Name Server
- **AAAA** - An end device IPv6 address (pronounced quad-A)
- **MX** - A Mail Exchange Record
- **SOA –** Start of Authority
- **CNAME –** Canonical Name
- **TXT –** Descriptive ASCII Text

When a client makes a query, the server DNS process first looks at its own records to resolve the name. If it is unable to resolve the name by using its stored records, it contacts other servers to resolve the name.

After a match is found and returned to the original requesting server, the server temporarily stores the numbered address in the event that the same name is requested again.

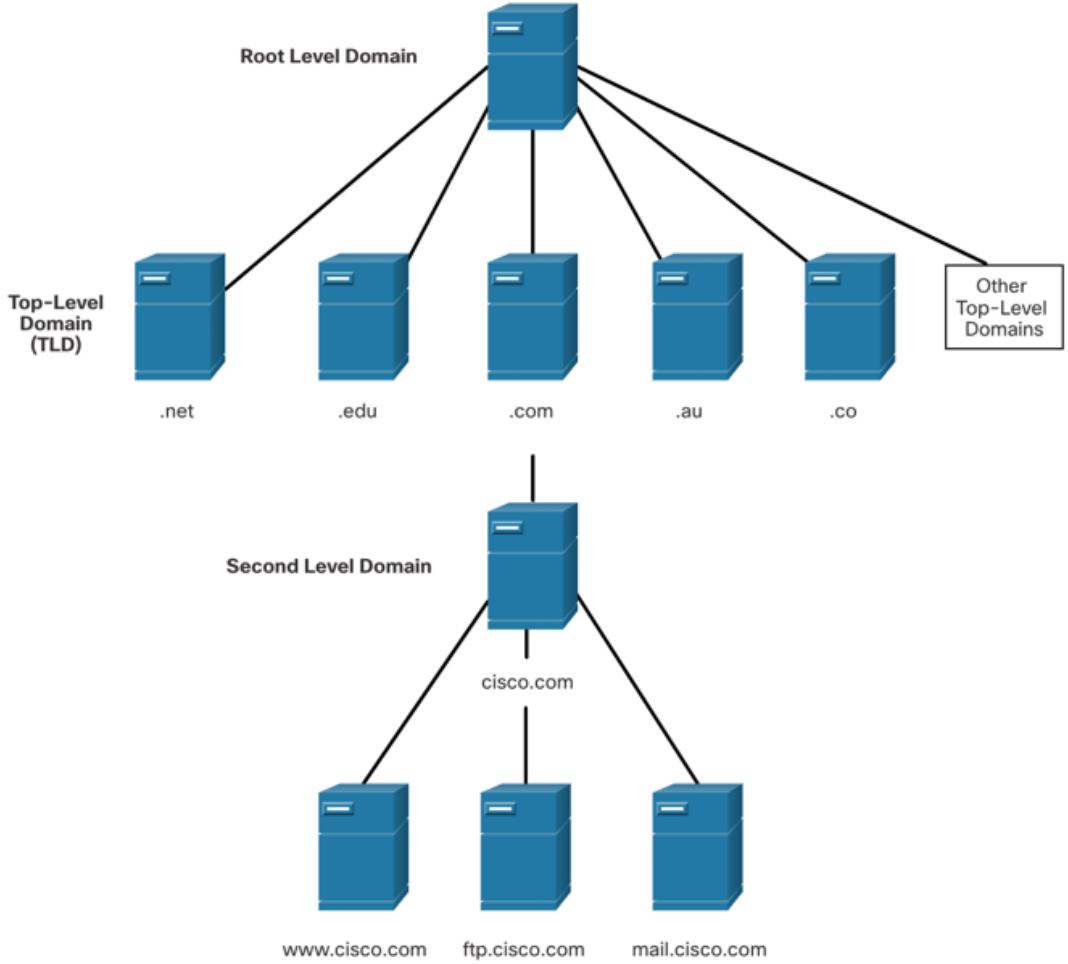
DNS Message Format (Cont.)

DNS uses the same message format between servers, consisting of a question, answer, authority, and additional information for all types of client queries and server responses, error messages, and transfer of resource record information.

DNS message section	Description
Question	The question for the name server
Answer	Resource Records answering the question
Authority	Resource Records pointing toward an authority
Additional	Resource Records holding additional information

DNS Hierarchy

- DNS uses a hierarchical system to create a database to provide name resolution.
- Each DNS server maintains a specific database file and is only responsible for managing name-to-IP mappings for that small portion of the entire DNS structure.
- When a DNS server receives a request for a name translation that is not within its DNS zone, the DNS server forwards the request to another DNS server within the proper zone for translation.
- Examples of top-level domains:
  - **.com** - a business or industry
  - **.org** - a non-profit organization
  - **.au** - Australia



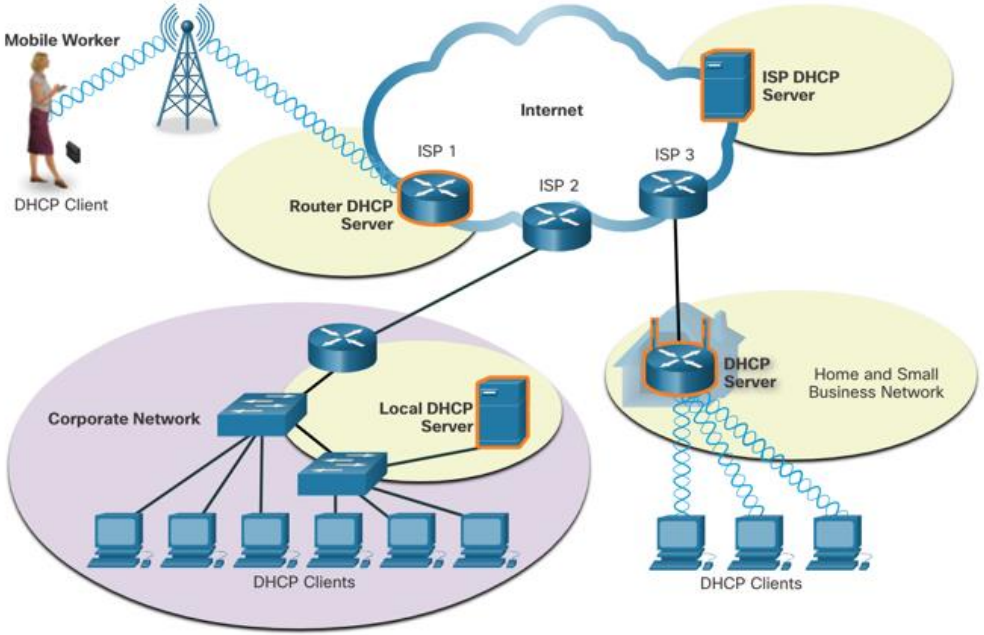
The nslookup Command

- Nslookup is a computer operating system utility that allows a user to manually query the DNS servers configured on the device to resolve a given host name.
- This utility can also be used to troubleshoot name resolution issues and to verify the current status of the name servers.
- When the **nslookup** command is issued, the default DNS server configured for your host is displayed.
- The name of a host or domain can be entered at the **nslookup** prompt.

```
C:\Users> nslookup
Default Server:  dns-sj.cisco.com
Address:  171.70.168.183
> www.cisco.com
Server:  dns-sj.cisco.com
Address:  171.70.168.183
Name:    origin-www.cisco.com
Addresses:  2001:420:1101:1::a
          173.37.145.84
Aliases:  www.cisco.com
> cisco.netacad.net
Server:  dns-sj.cisco.com
Address:  171.70.168.183
Name:    cisco.netacad.net
Address:  72.163.6.223
>
```

Dynamic Host Configuration Protocol

- The Dynamic Host Configuration Protocol (DHCP) for IPv4 service automates the assignment of IPv4 addresses, subnet masks, gateways, and other IPv4 networking parameters.
- DHCP is considered dynamic addressing compared to static addressing. Static addressing is manually entering IP address information.
- When a host connects to the network, the DHCP server is contacted, and an address is requested. The DHCP server chooses an address from a configured range of addresses called a pool and assigns (leases) it to the host.
- Many networks use both DHCP and static addressing. DHCP is used for general purpose hosts, such as end user devices. Static addressing is used for network devices, such as gateway routers, switches, servers, and printers.

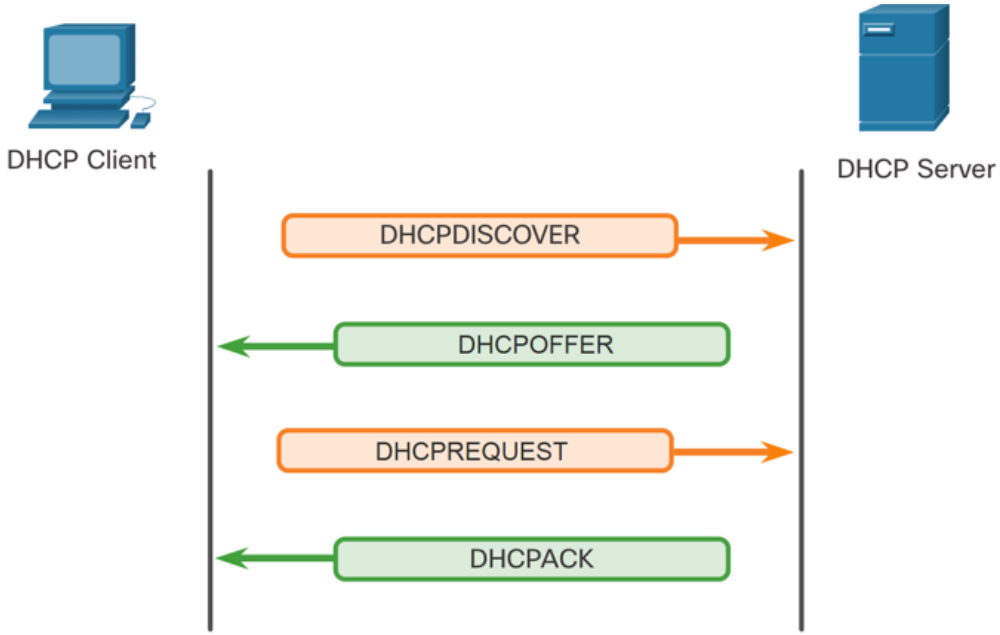


Note: DHCP for IPv6 (DHCPv6) provides similar services for IPv6 clients. However, DHCPv6 does not provide a default gateway address. This can only be obtained dynamically from the Router Advertisement message of the router.

DHCP Operation

The DHCP Process (DORA Process):

- When an IPv4, DHCP-configured device boots up or connects to the network, the client broadcasts a DHCP discover (DHCPDISCOVER) message to identify any available DHCP servers on the network.
- A DHCP server replies with a DHCP offer (DHCPOFFER) message, which offers a lease to the client. (If a client receives more than one offer due to multiple DHCP servers on the network, it must choose one.)
- The client sends a DHCP request (DHCPREQUEST) message that identifies the explicit server and lease offer that the client is accepting.
- The server then returns a DHCP acknowledgment (DHCPACK) message that acknowledges to the client that the lease has been finalized.
- If the offer is no longer valid, then the selected server responds with a DHCP negative acknowledgment (DHCPNAK) message and the process must begin with a new DHCPDISCOVER message.

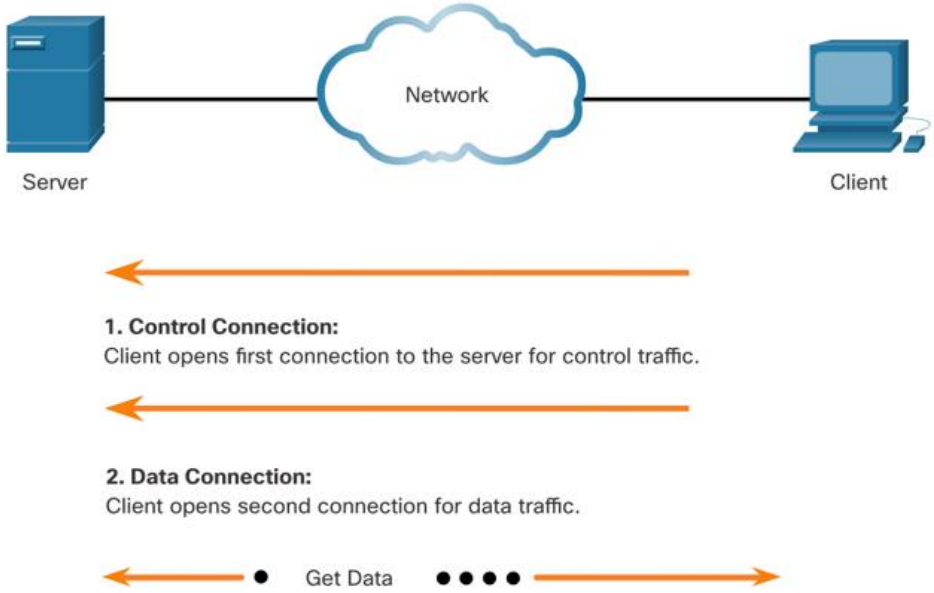


Note: DHCPv6 has a set of messages that is similar to those for DHCPv4. The DHCPv6 messages are SOLICIT, ADVERTISE, INFORMATION REQUEST, and REPLY.

File Sharing Services

File Transfer Protocol

FTP was developed to allow for data transfers between a client and a server. An FTP client is an application which runs on a computer that is being used to push and pull data from an FTP server.



Step 1 - The client establishes the first connection to the server for control traffic using TCP port 21. The traffic consists of client commands and server replies.

Step 2 - The client establishes the second connection to the server for the actual data transfer using TCP port 20. This connection is created every time there is data to be transferred.

Step 3 - The data transfer can happen in either direction. The client can download (pull) data from the server, or the client can upload (push) data to the server.

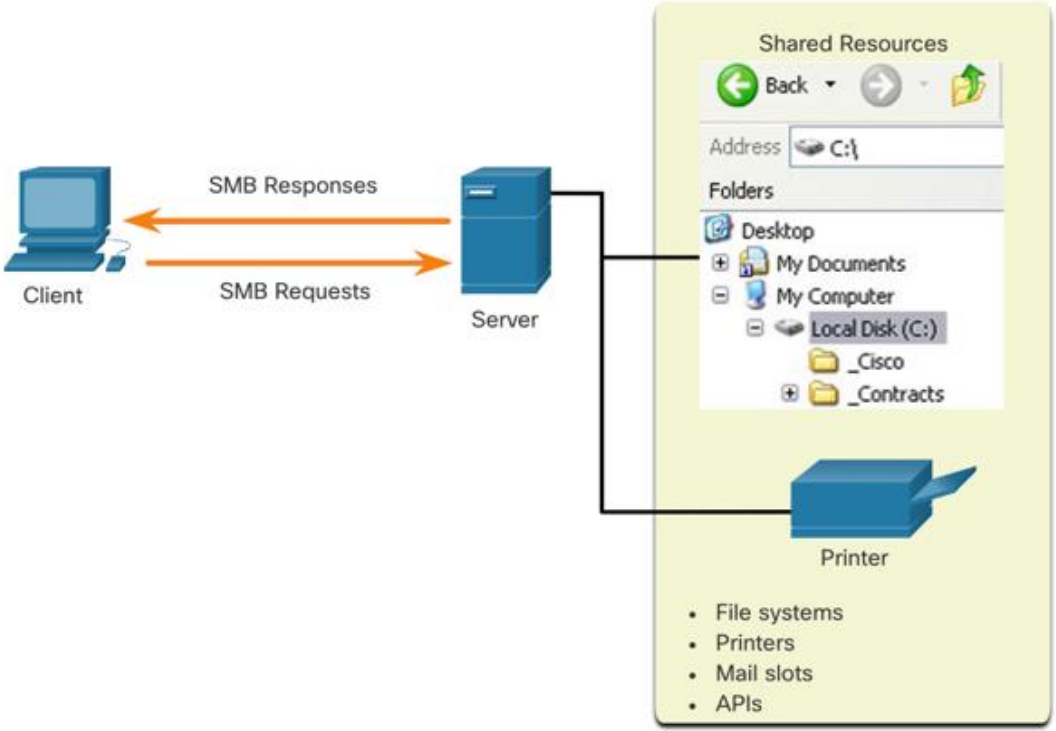
Server Message Block

The Server Message Block (SMB) is a client/server, request-response file sharing protocol. Servers can make their own resources available to clients on the network.

Three functions of SMB messages:

- Start, authenticate, and terminate sessions
- Control file and printer access
  - (Eg. Copying file from one desktop to another)
- Allow an application to send or receive messages to or from another device

Unlike the file sharing supported by FTP, clients establish a long-term connection to servers. After the connection is established, the user of the client can access the resources on the server as though the resource is local to the client host.



Summary

Summary of Main Teaching Points

- Advantages of UDP over TCP
- Difference between a socket and a port
- How does three-way handshake work ?
- What would happen if TCP were used instead of UDP for some applications that use UDP?
- Advantages/Disadvantages of using a Peer-to-Peer Network?
- Why is HTTPS recommended over HTTP on websites such as banks or online stores?
- Three common HTTP Message Types.
- Difference between POP3 and IMAP.
- Difference between FTP and Server Message Block (SMB)

Question and Answer Session

A·P·U
ASIA PACIFIC UNIVERSITY
OF TECHNOLOGY & INNOVATION

Q & A

A · P · U
ASIA PACIFIC UNIVERSITY
OF TECHNOLOGY & INNOVATION

What We Will Cover Next:

Application Layer


CISCO