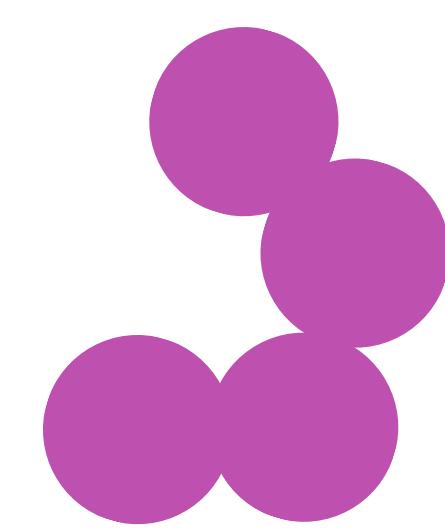


Module 1



NETWORK COMMUNICATION MODELS



History Leading to TCP/IP

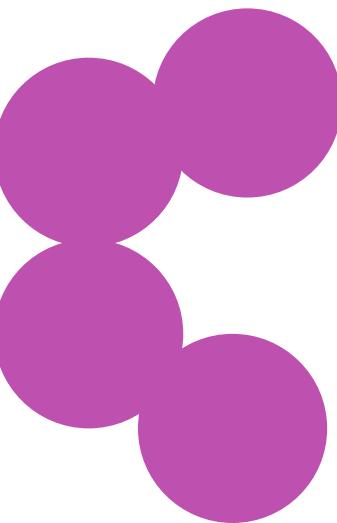


- Once upon a time, networking protocols didn't exist, including TCP/IP.
- Vendors created the first networking protocols; these protocols supported only that vendor's computers.



- There are two basic types of models that we use to describe the functions that must occur in order for network communications to be successful:
protocol models and reference models.

- Protocol model - TCP/IP suite
- Reference model - OSI 7 layer

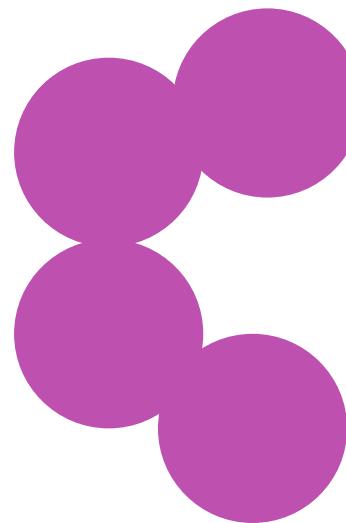
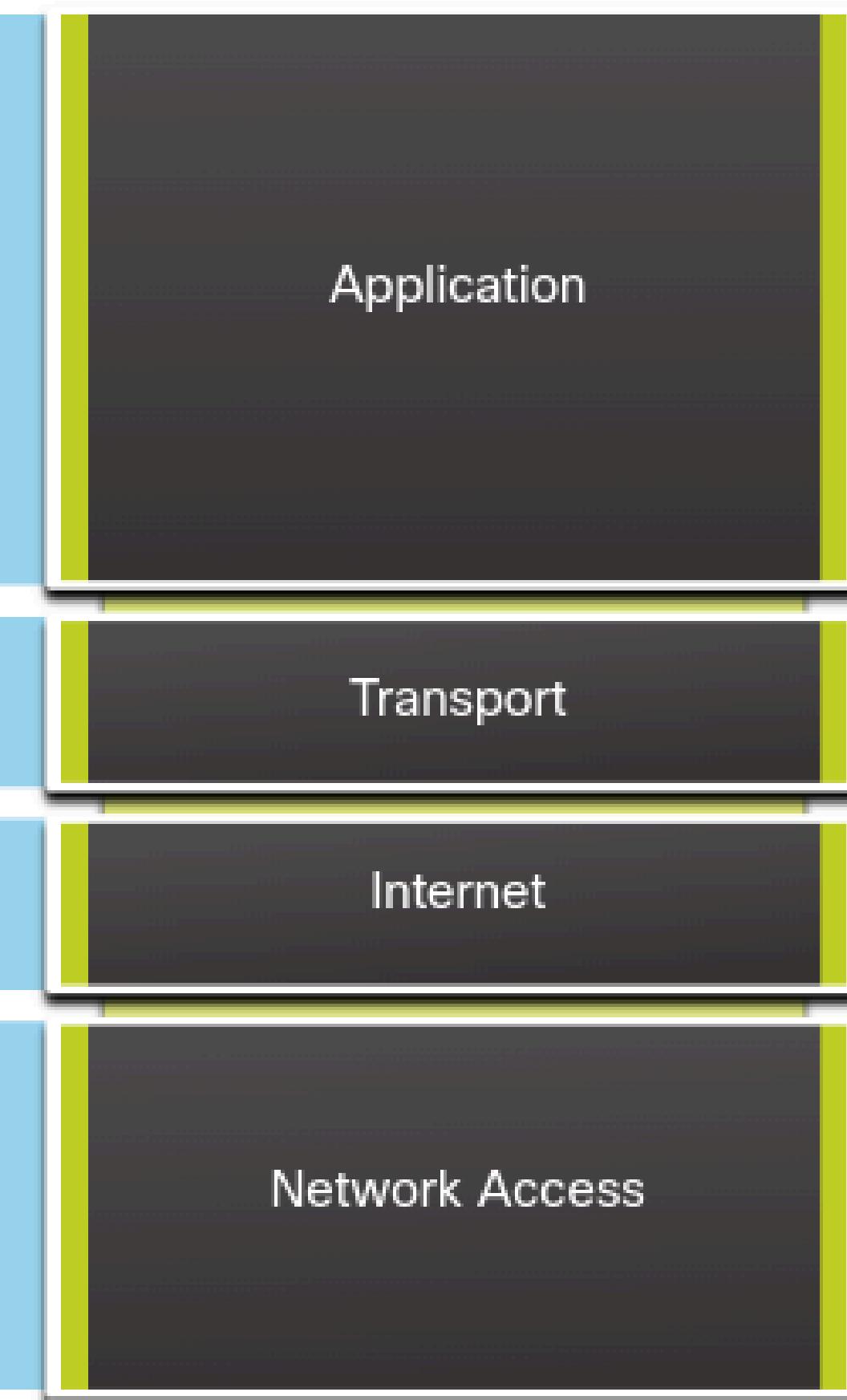




OSI Model



TCP/IP Model





DevKTOps

OSI 7 Layers

7. Application

6. Presentation

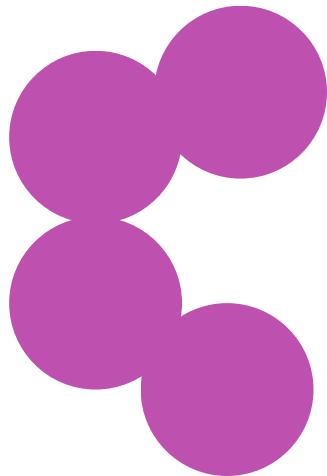
5. Session

4. Transport

3. Network

2. Data Link

1. Physical





DevKTOps

7. Application

Data

6. Presentation

Data

5. Session

Data

4. Transport

Segments

3. Network

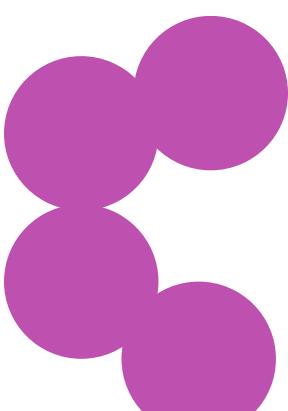
Packets

2. Data Link

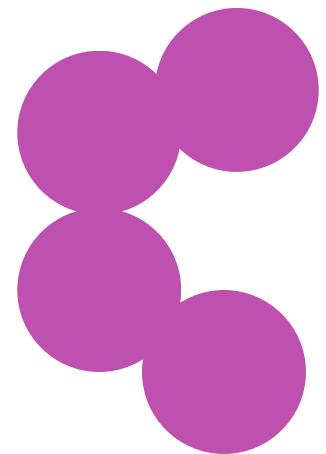
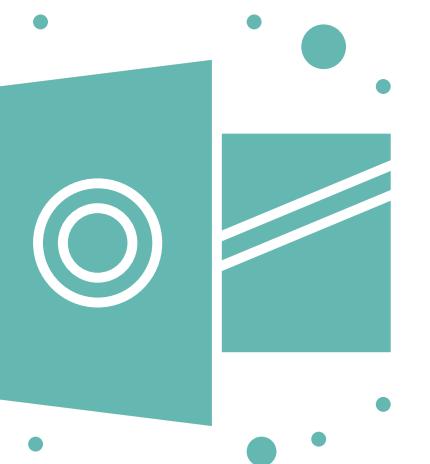
Frames

1. Physical

Bits



7. Application





Protocol	Description
DNS	Resolves internet name to IP address.
SSH	Used to provide remote access to servers and networking devices.
SMTP	Sends email messages and attachments from client to servers and from servers to other email servers.
IMAP	Used by email Client to retrieve email and attachments from a remote server.
POP	Used by email Client to retrieve email and attachments from a remote server.



DHCP	<p>Used to automatically configure devices with IP addressing and other necessary information to enable them to communicate over the internet.</p>
HTTP	<p>Used by web browsers to request web pages and web servers to transfers the files that make up web pages of the World Wide Web.</p>
FTP	<p>Used for interactive file transfer between systems.</p>

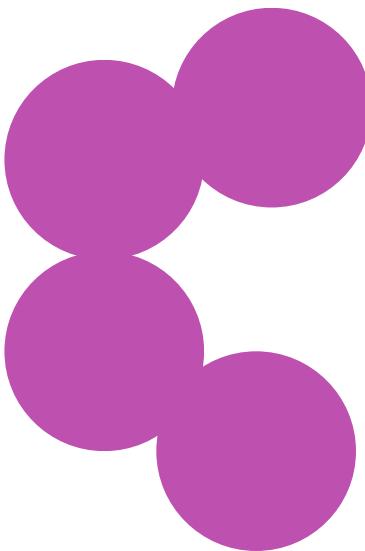
6. Presentation



Translation

Compression/Decompression

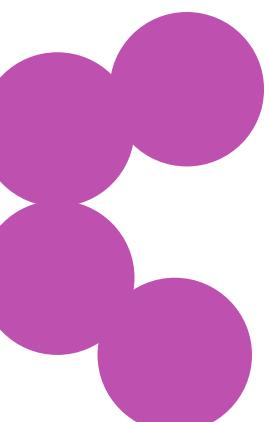
Encryption/Decryption





5. Session

- **Session Layer is the layer of the ISO Open Systems Interconnection (OSI) model that controls the dialogues (connections) between computers.**
- **It establishes, manages, and terminates the connections between the local and remote application.**





4. Transport

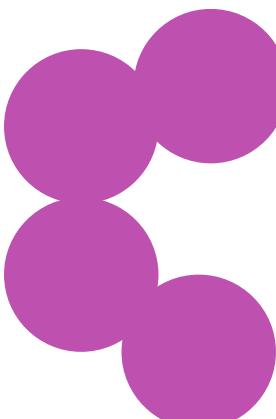
Segmentation

Flow Control

Error Control

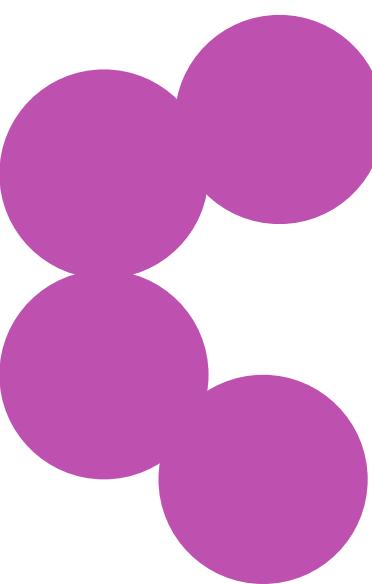
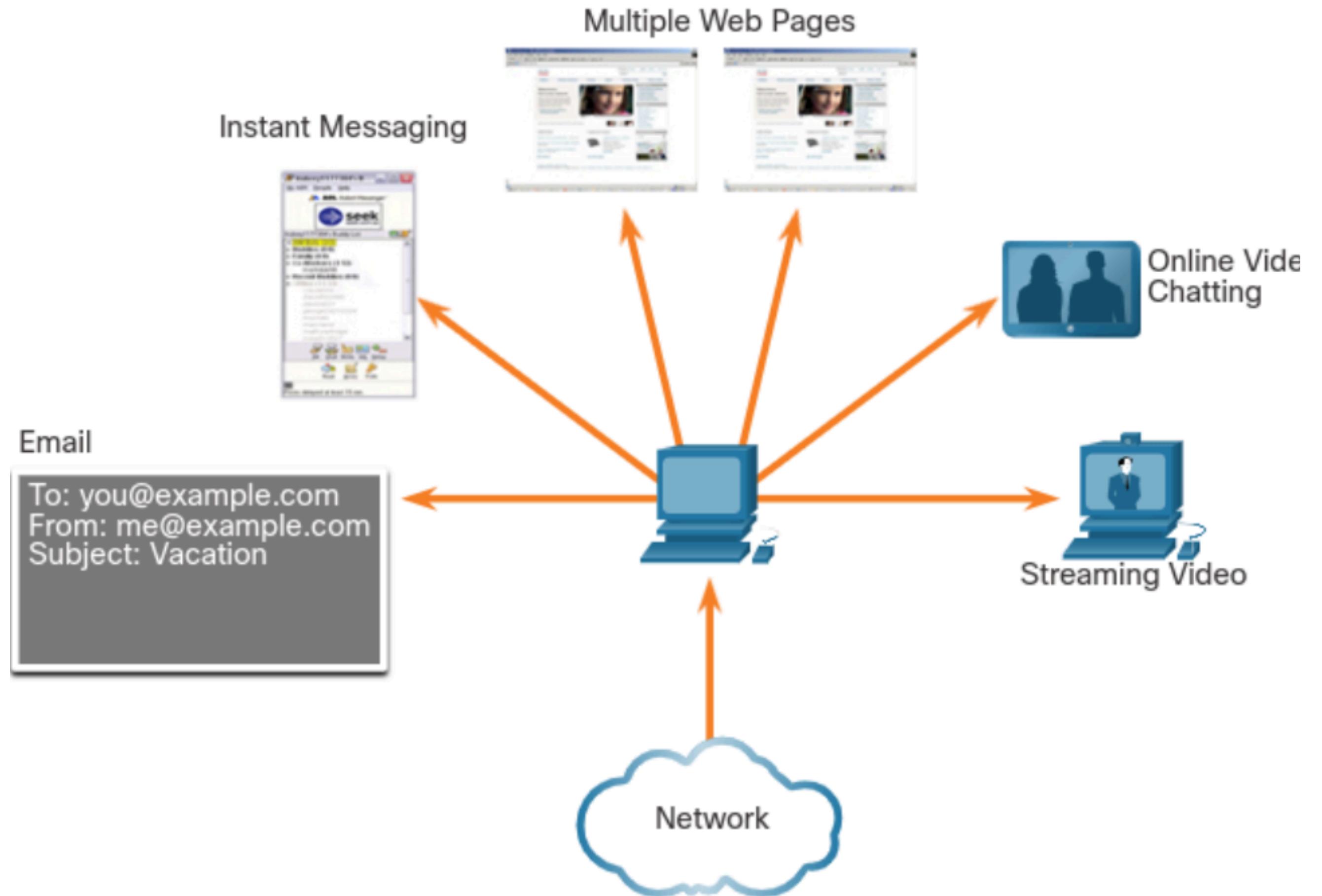
Seq 1

Seq 2



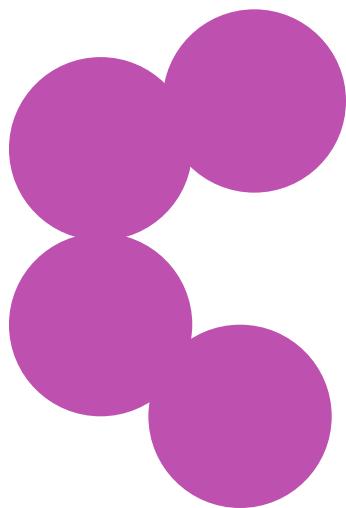
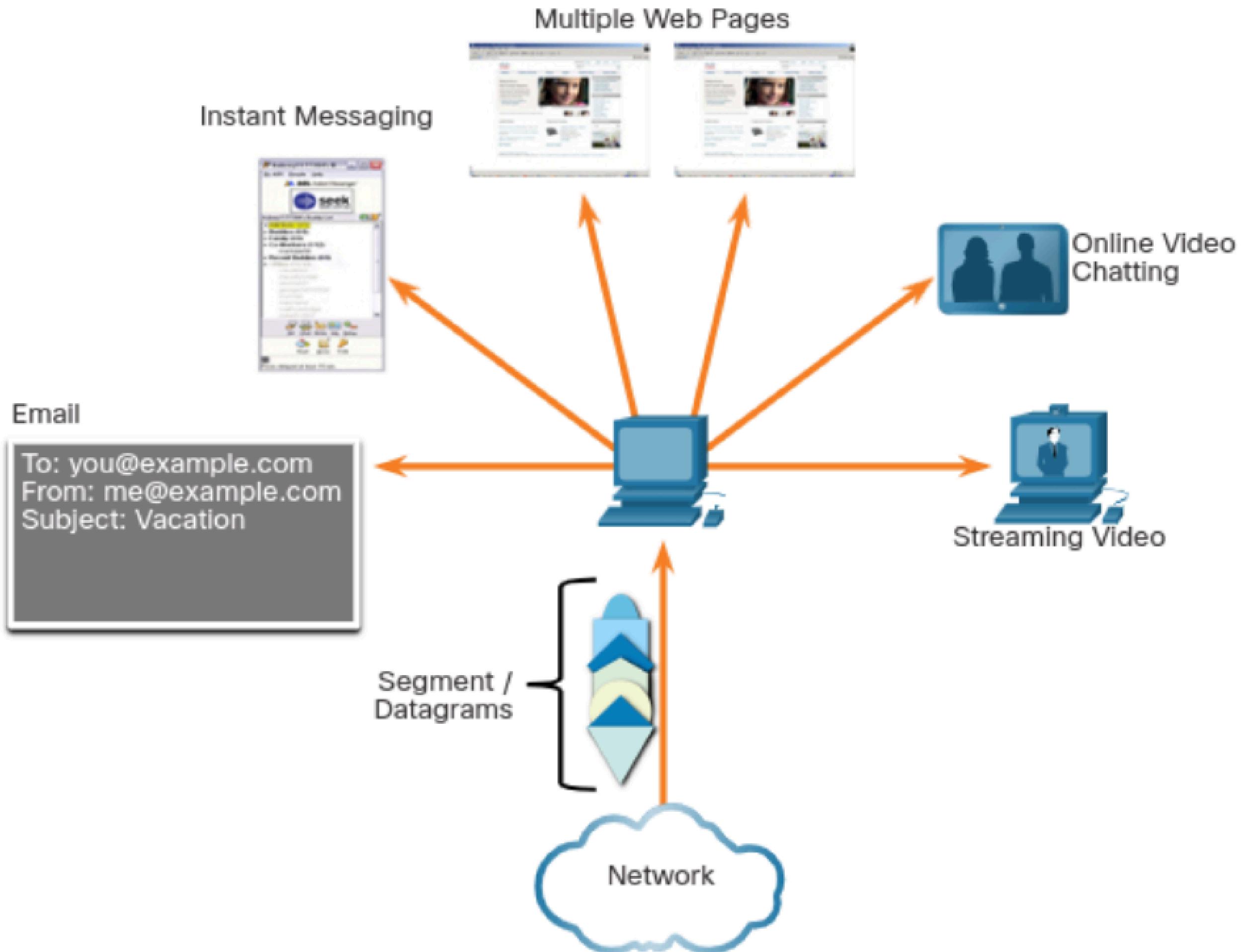


DevKTOps



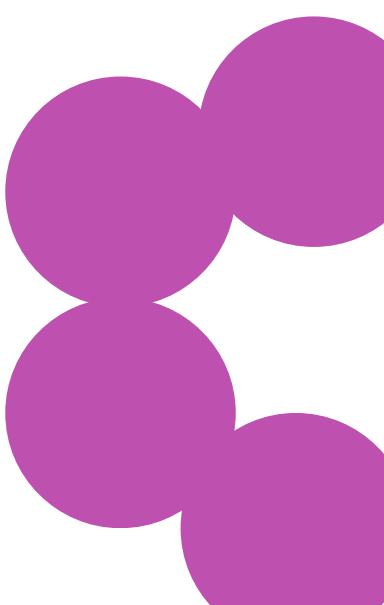
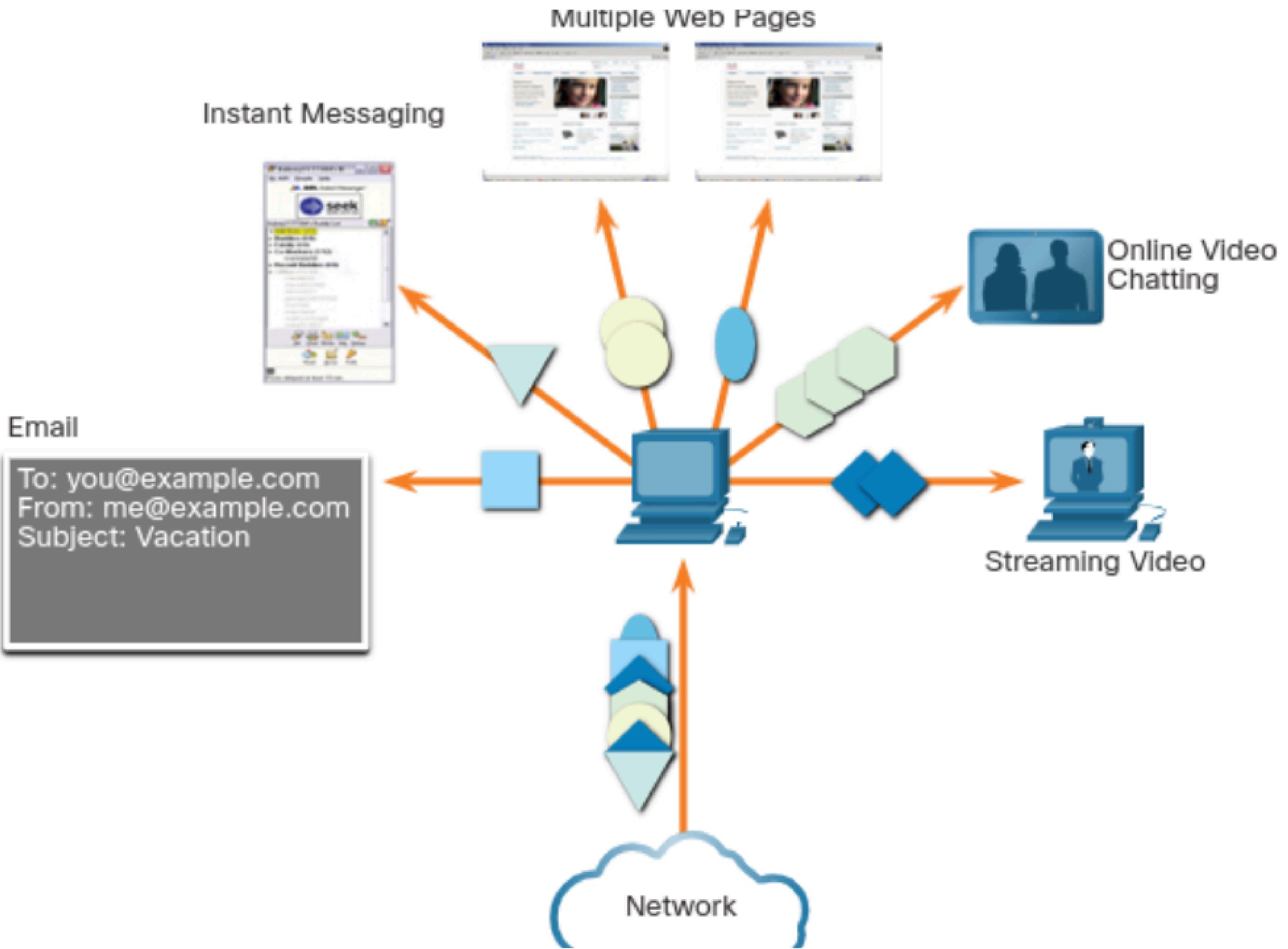


DevKTOps





DevKTOps





DevKTOps

Different Applications	Electronic Mail	HTML Page	Internet Chat
Port	110	80	531

The diagram features a light blue house with a steeper roofline. A computer monitor is positioned on top of the roof. Below the roofline, there are four rectangular windows divided by vertical and horizontal lines. The first window contains text representing different applications. The second window shows a screenshot of an email inbox. The third window shows a screenshot of a web browser displaying a news article. The fourth window is empty, represented by a light gray background.

To: you@example.com
From: me@example.com
Subject: Vacation

Electronic Mail

HTML Page

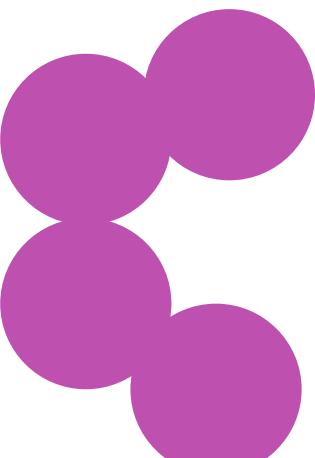
Internet Chat

Different Applications

110

80

531



Port Number Groups



- The Internet Assigned Numbers Authority (IANA) is the standards organization responsible for assigning various addressing standards, including the 16-bit port numbers.
- The 16 bits used to identify the source and destination port numbers provides a range of ports from 0 through 65535.
- The IANA has divided the range of numbers into the following three port groups.

Port Number Groups



Port Group

Number Range

Well-known Ports

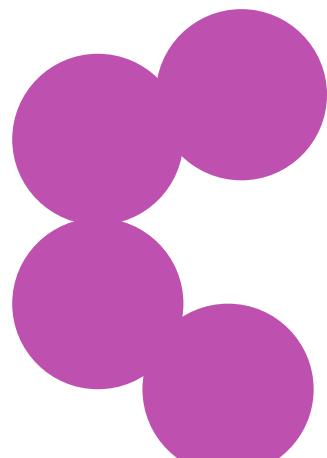
0 to 1,023

Registered Ports

1,024 to 49,151

Private and/or Dynamic Ports

49,152 to 65,535

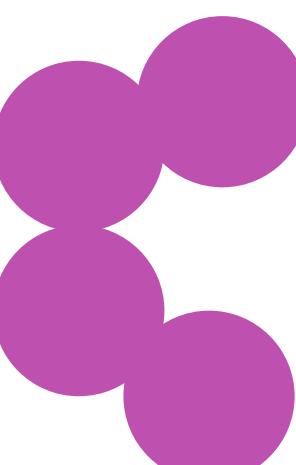


Well-known Ports



DevKTOps

- 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



DevKTOps

- 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

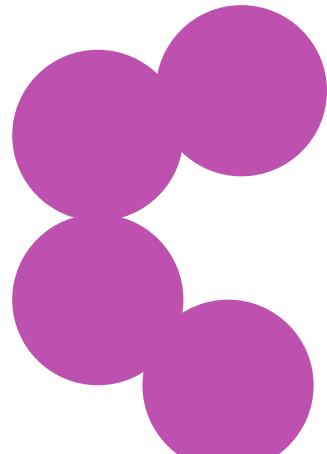


- These ports are also known ephemeral ports.
- The client's OS usually assign port numbers dynamically when a connection to a service is initiated.
- The dynamic port is then used to identify the client application during communication.



DevKTOps

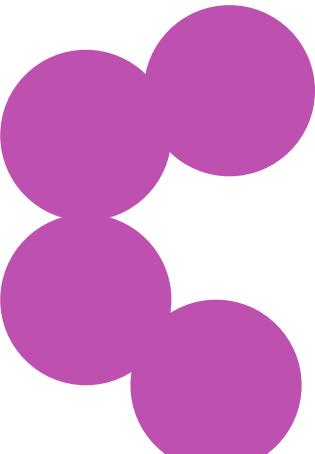
Port number	Protocol	Application
20	TCP	File Transfer Protocol (FTP) - Data
21	TCP	File Transfer Protocol (FTP) - Control
22	TCP	SSH
23	TCP	Telnet
25	TCP	SMTP
53	UDP, TCP	DNS
67	UDP	DHCP - Server





DevKTOps

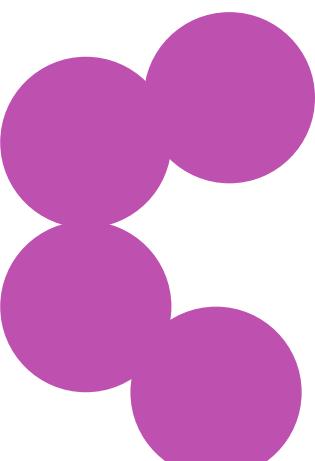
Port number	Protocol	Application
68	UDP	DHCP - Client
69	UDP	TFTP
80	TCP	HTTP
110	TCP	POP3
143	TCP	IMAP
161	UDP	SMTP
443	TCP	HTTPS





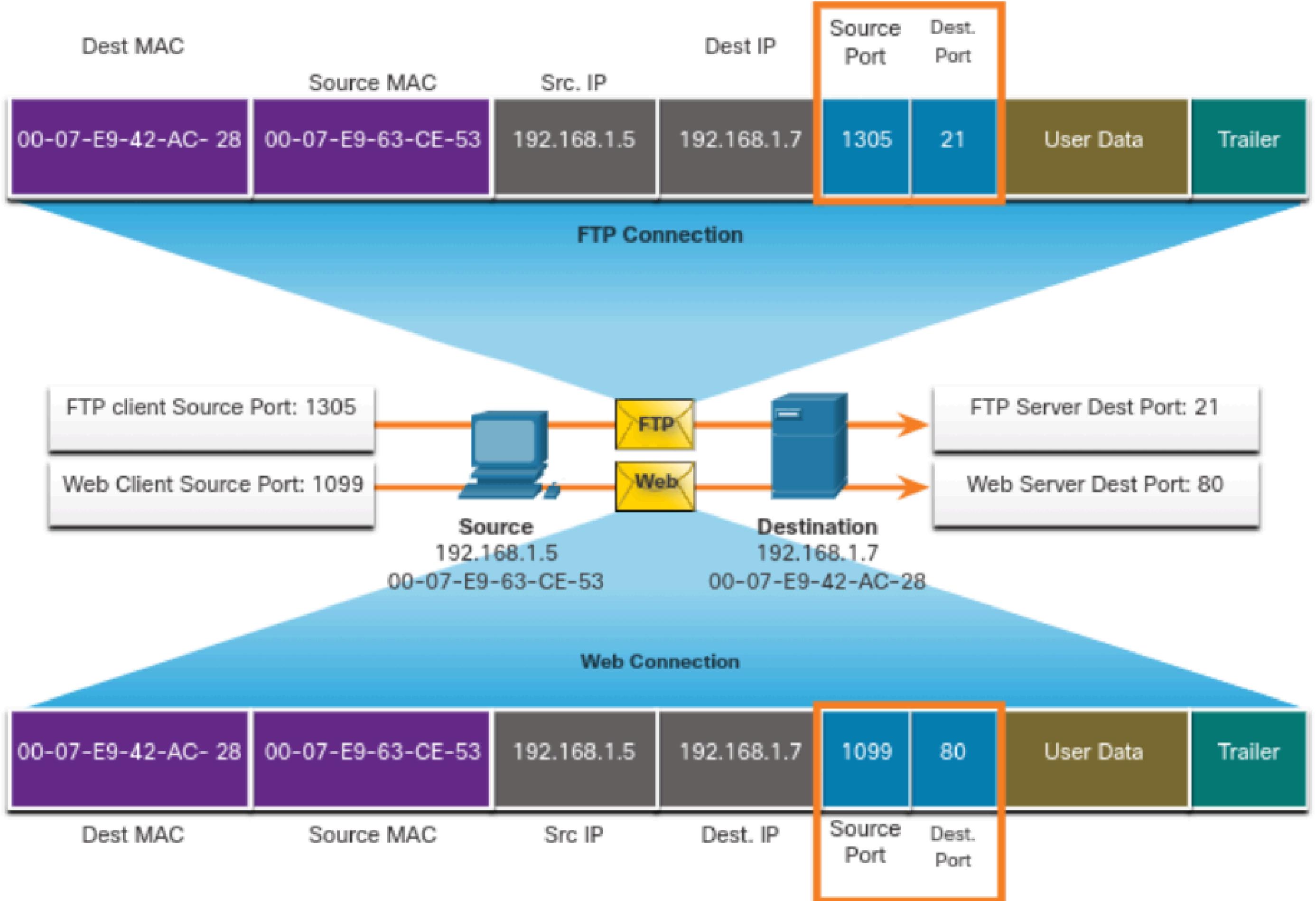
Some applications may use both TCP and UDP. For example, DNS uses UDP when clients send requests to a DNS server. However, communication between two DNS servers always uses TCP.

Search the IANA website for port registry to view the full list of port numbers and associated applications.



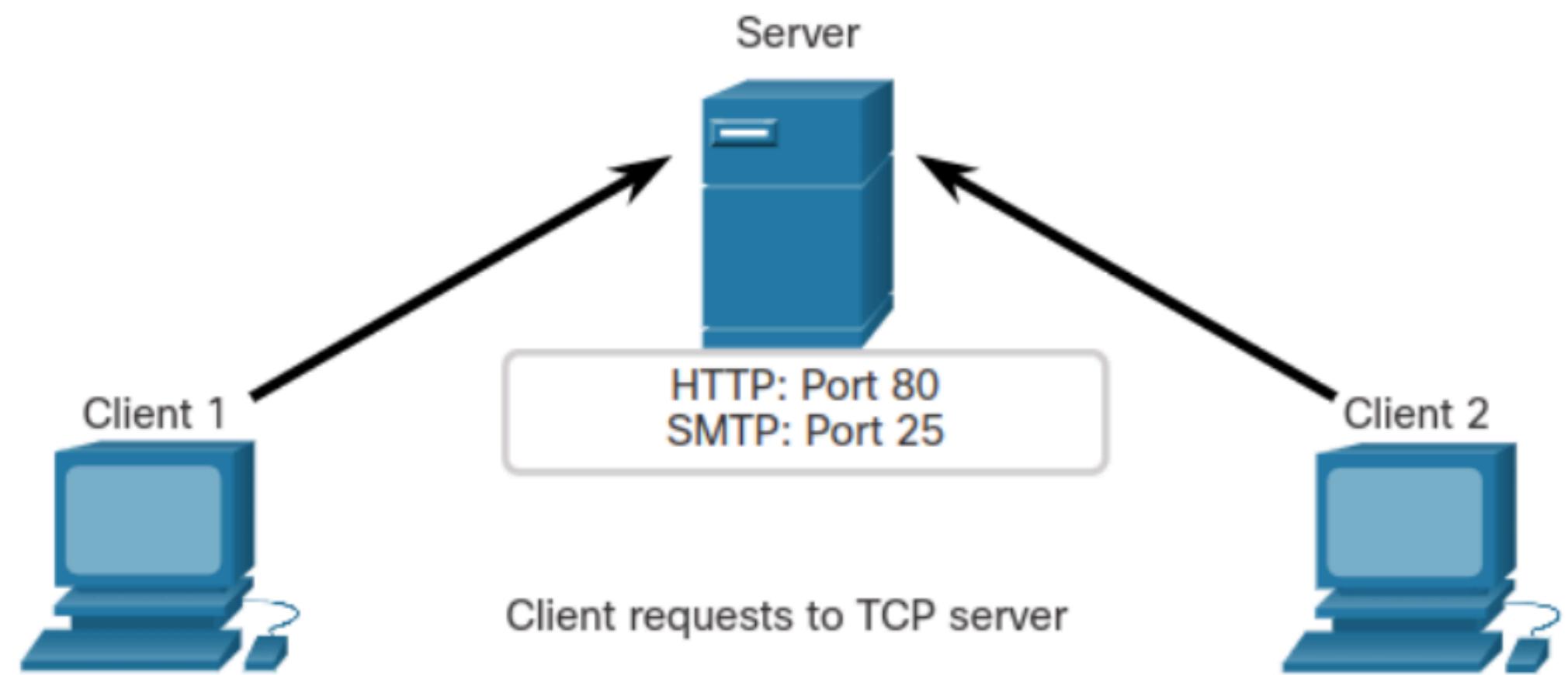


DevKTOps





DevKTOps



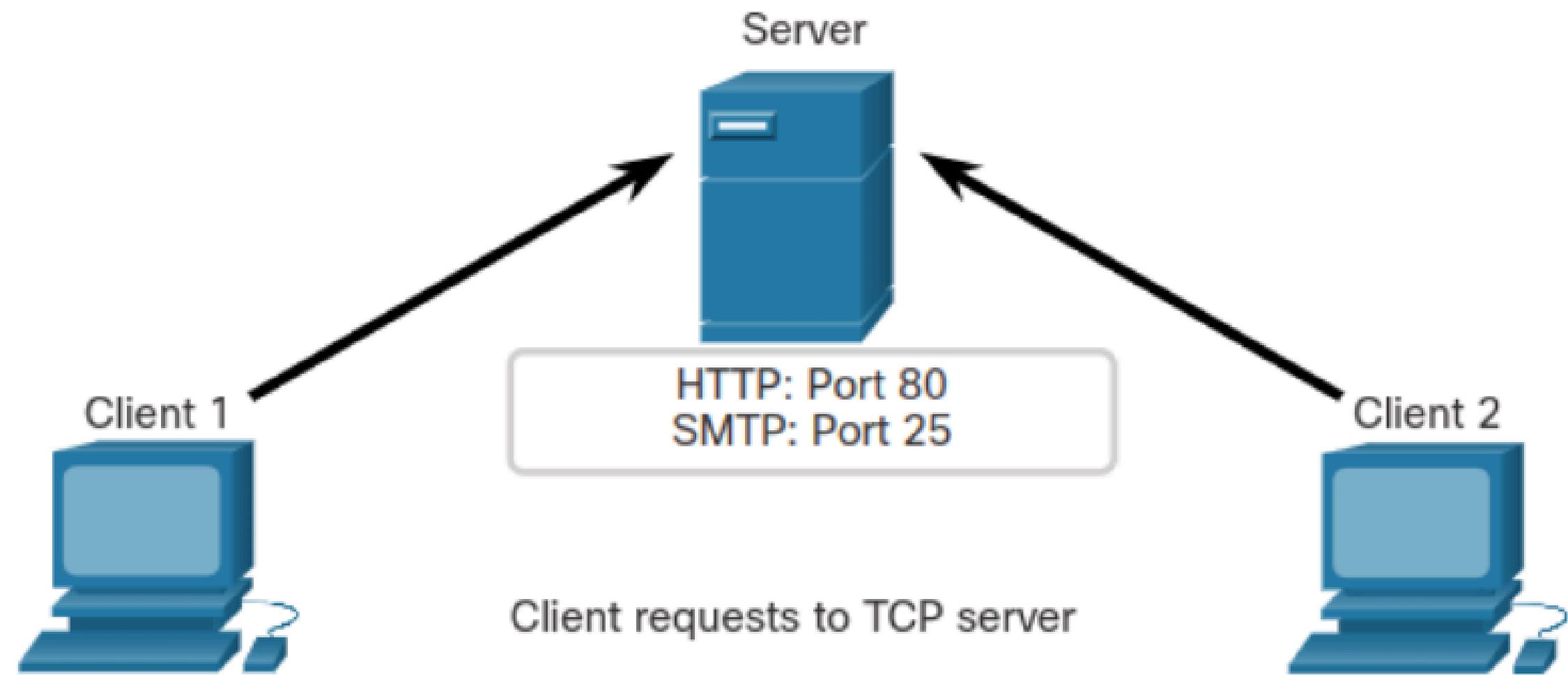
HTTP Request:
Source Port: 49152
Destination Port: 80

SMTP Request:
Source Port: 51152
Destination Port: 25

Use well known port numbers as the destination port.



DevKTOps



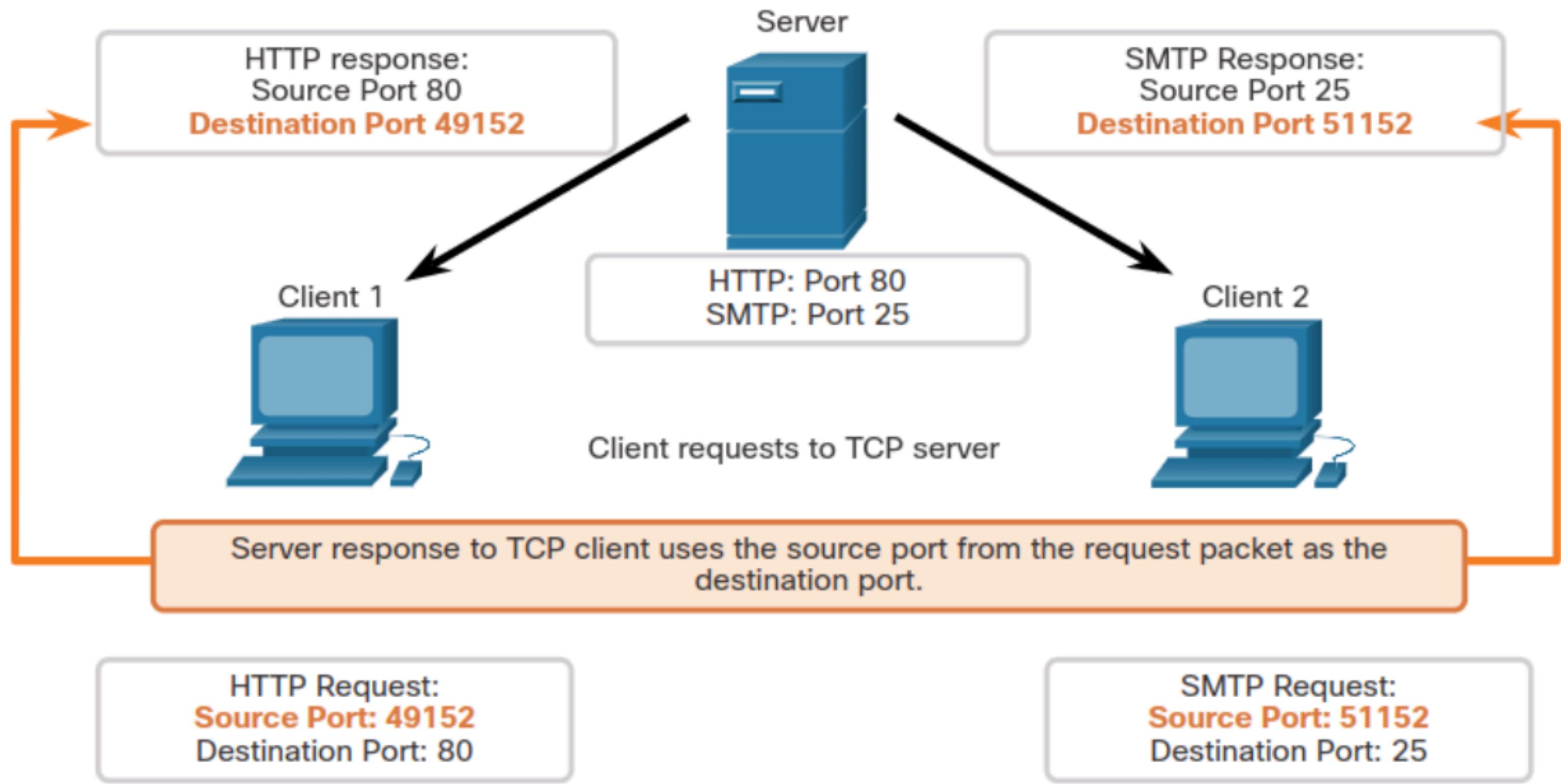
HTTP Request:
Source Port: 49152
Destination Port: 80

SMTP Request:
Source Port: 51152
Destination Port: 25

Use dynamic port numbers as the source port.



DevKTOps

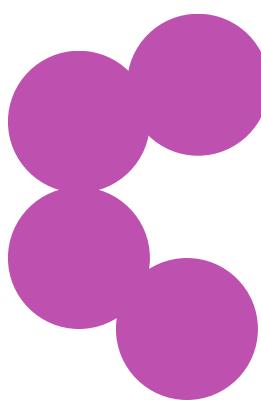




Transport Layer Protocols

TCP

UDP



TCP



UDP



TCP



UDP



DevKTOps



TCP



DevKTOps

- 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

In order to maintain the state of a conversation and track the information, TCP must first establish a connection between the sender and the receiver. This is why TCP is known as a connection-oriented protocol.

UDP



DevKTOps

UDP is a connectionless protocol.

Because UDP does not provide reliability or flow control, it does not require an established connection.

Because UDP does not track information sent or received between the client and server,



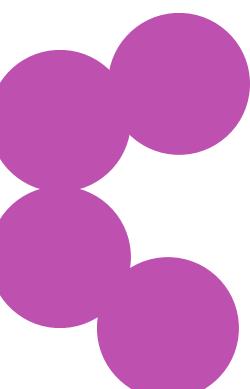
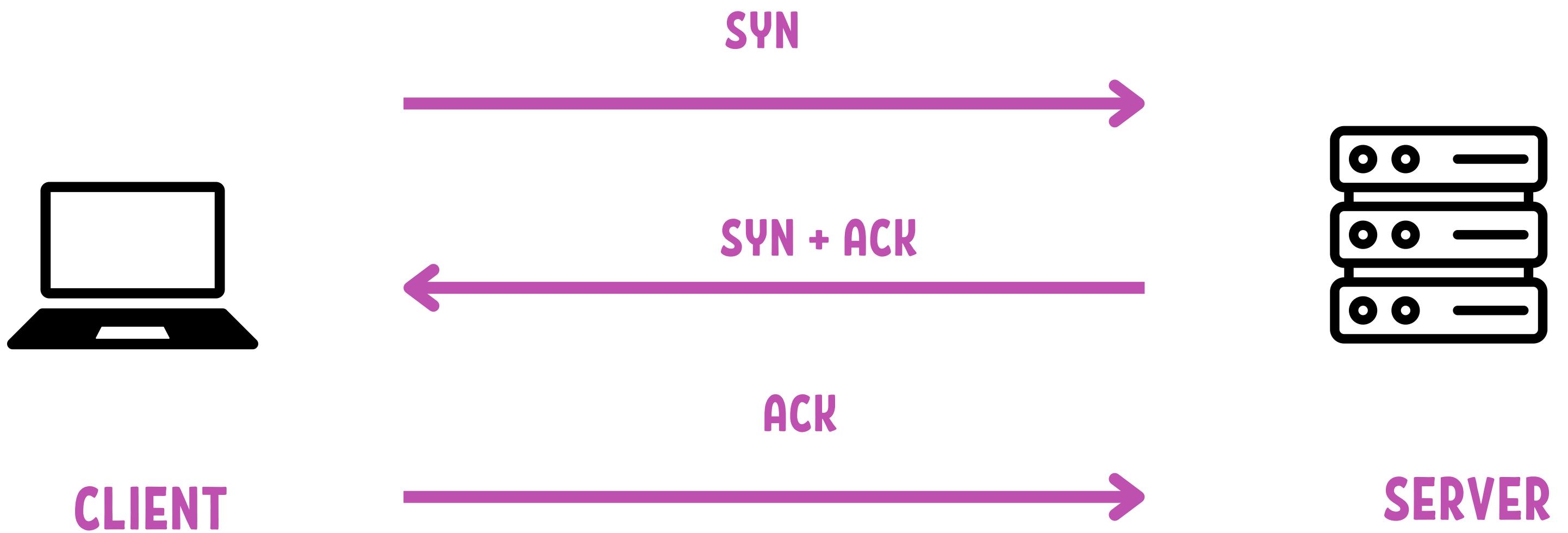
TCP

- Connection oriented protocol
- Reliable
- Acknowledges data
- Resends lost data

UDP

- Connectionless
- Fast
- Does not require acknowledgements
- Does not resend lost data

TCP 3-way Handshakes





THANK
YOU!

