CSCI 466: Networks

Review

Reese Pearsall

Fall 2023

*All images are stolen from the internet

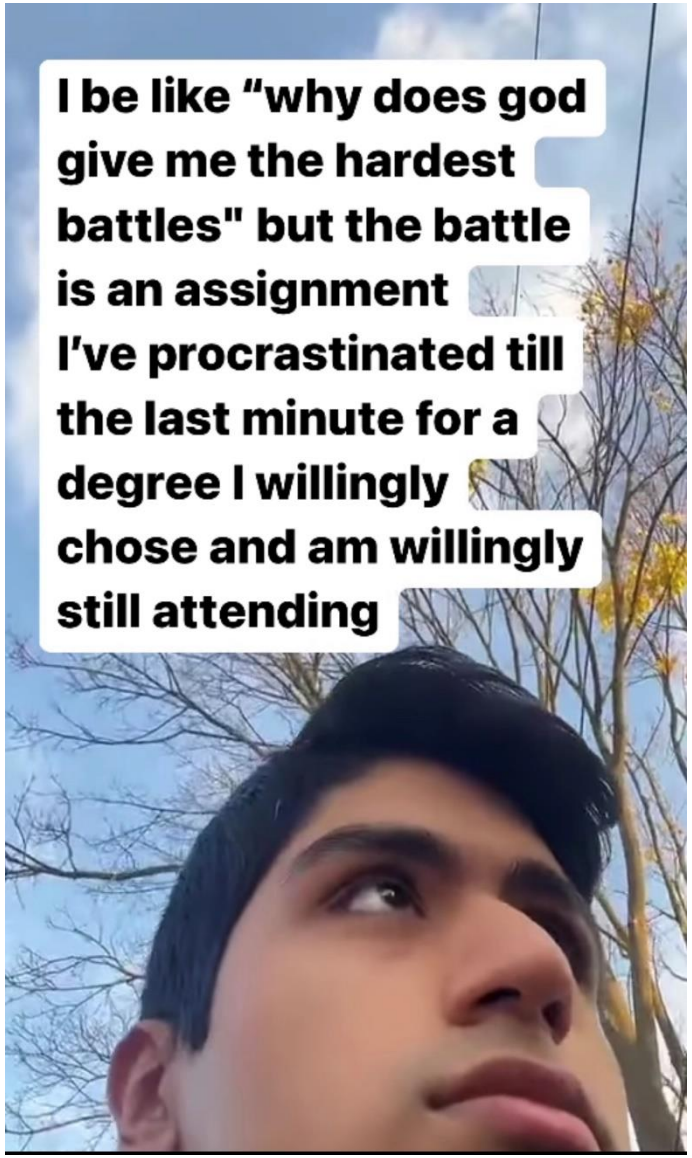
Announcements

Wireshark Lab 4 due **tonight**

Final Exam on Friday

Wireshark Lab 5 (course evaluation) and PA5 due next Wednesday (12/13)

- Remember to submit a repo link to D2L



Final Exam Structure

No notes allowed
10% of your grade
Please show up

Part I. OSI Model
For each layer

- Name the layer (Ex. Network Layer)
- Provide a primary responsibility/functionality (Ex. Forwarding and Routing)
- Provide the unit of data that is being transmitted (Ex. Datagram)

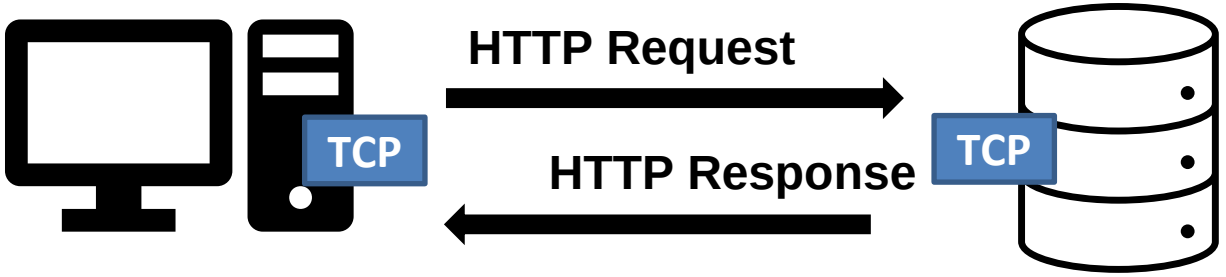
Part II. Question 2

- Won’t tell you the exact question, but it will require your knowledge of how the internet works and how two hosts communicate with each other

Application Layer

The layer which **interacts directly with applications** and provide necessary protocols and services for web applications. Specifies the shared commination protocol(s) that will be used by hosts in a communication protocol

HyperText Transfer Protocol (HTTP)- protocol that dictates the transmitting of hypermedia documents, such as HTML and other webpage objects



Uniform Resource Locator (**URL**)- Addressing scheme for web objects

```
scheme://domain:port/path_to_object?query_string

http://cs.montana.edu/pearsall/classes/fall2022/466/main.html
```

Object

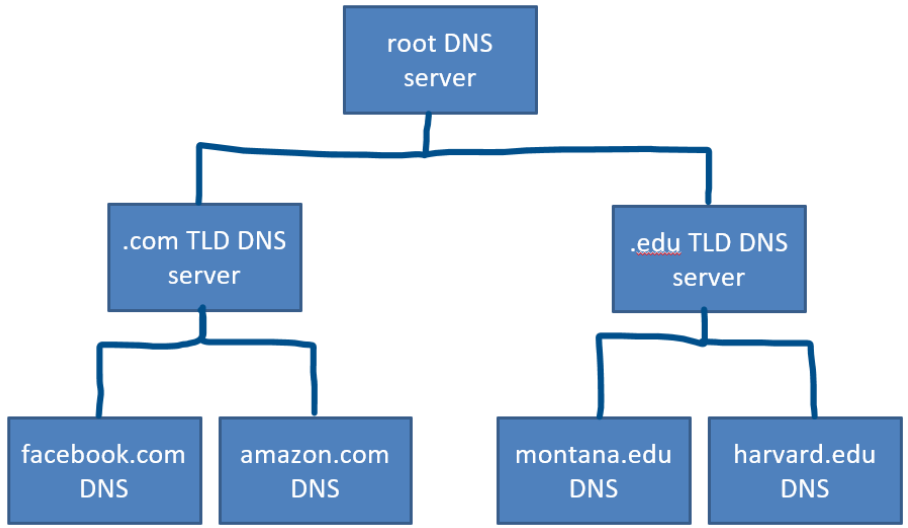
GET: Download resource
HEAD: Get resource metadata
POST: Upload form contents
PUT: Upload object to URL
DELETE: Delete object from URL

- ❑ Informational Responses (100s)
- ❑ Successful Responses (200s)
- ❑ Redirection messages (300s)
- ❑ Client error response (400s)
- ❑ Server error response (500s)

- **Domain Name System (DNS)** is a **distributed**, **hierarchical** database used for mapping hostnames to IP address

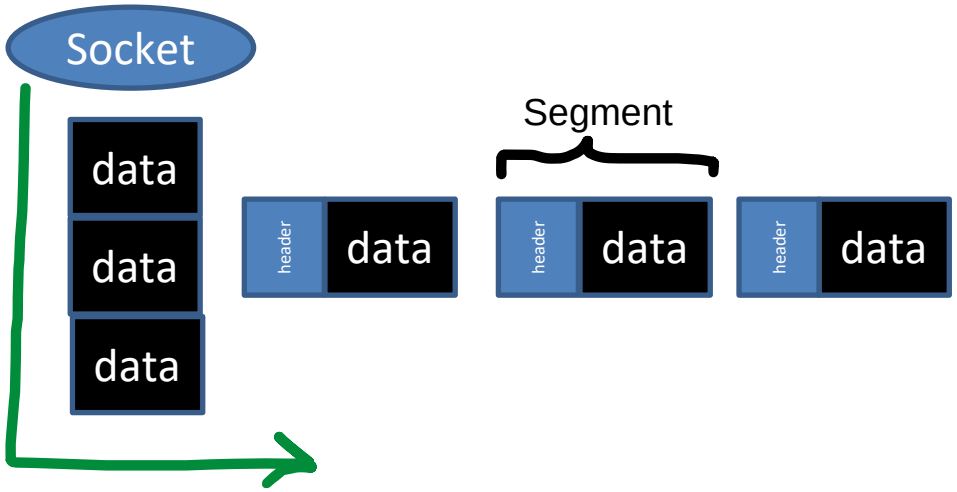
Prior to creating a TCP connection and sending an HTTP request, we first need to issue a DNS request!

(Built on UDP, lookups happen on port 53)

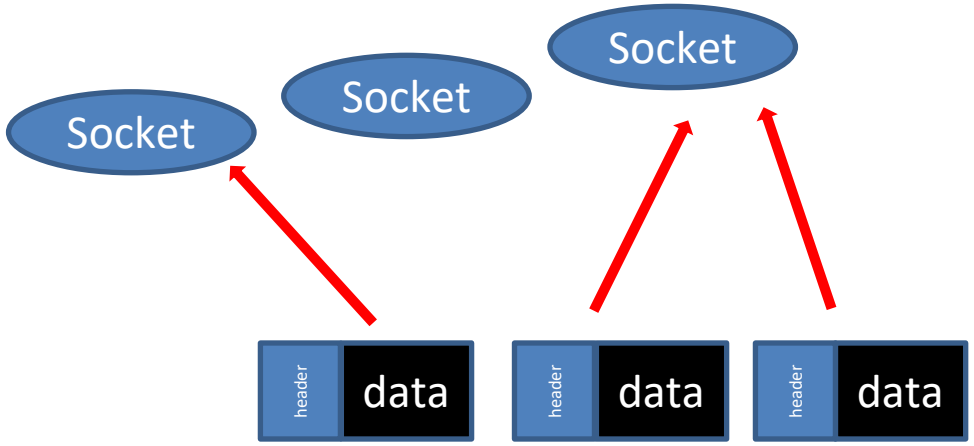


Transport Layer - Provides host-to-host, **reliable data transfer**, and dictates the flow of data

Multiplexing is the process of gathering chunks from sockets, encapsulating chunks with header information, and passing the segment into the network layer



Demultiplexing is the receiving segments from the transport layer and delivering the segment to the correct socket.



Transport Layer

Provides host-to-host, **reliable data transfer**, and dictates the flow of data

User Datagram Prot. (UDP)

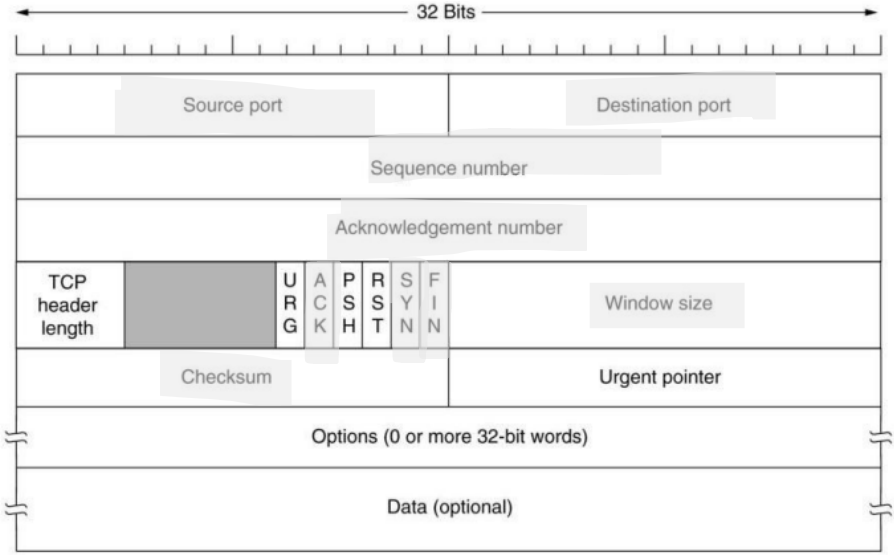
Unreliable data transfer

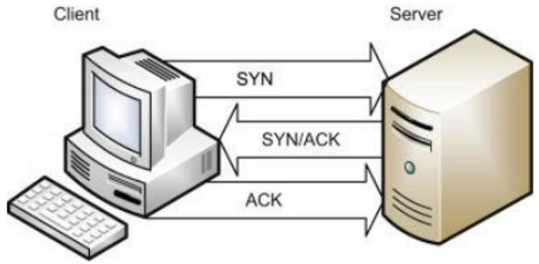
- Connection-less
  - Don't know if receiver is present
- No flow control
  - Overflow at receiver possible
- No congestion control
  - Sender can overload the network
- No guarantees on
  - End-to-end delay
  - Throughput
  - Security

Transmission Control Prot. (TCP)

Reliable stream transport

- Connection-oriented
  - Establishes receiver presence
- Flow control
  - Sender won't overwhelm receiver
- Congestion control
  - Senders won't overload network
- No guarantees on
  - End-to-end delay
  - Throughput
  - Security





TCP Handshake

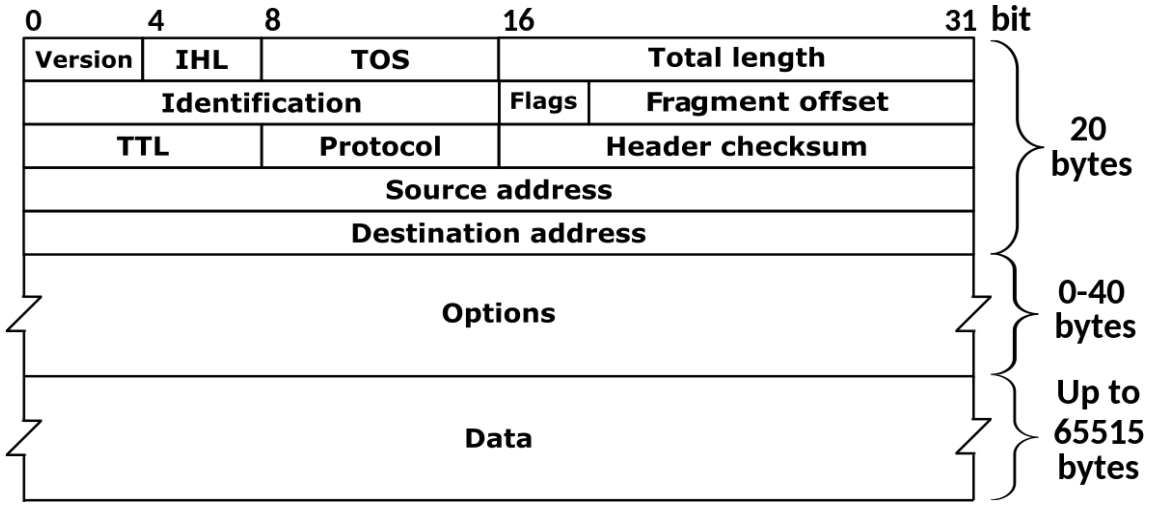
"Self-Clocking"

Network Layer | Provides Routing and Forwarding Functionality

Our segments from the transport layer are now encapsulated into network layer datagrams

IP Addresses are assigned here

IPv4



IPv4: 32-bit addresses (decimal)

192.149.252.76

Data Plane

Forwarding: move packets from router's input to appropriate router output

Address range	Interface (output link)
128.11.52.0 – 128.11.52.255	1
153.90.2.0 – 153.90.2.255	2
153.90.2.87 – 153.90.2.89	3

Address range	Interface (output link)
11001000 00010111 00010*** ********	1
11001000 00010111 00011000 ********	2
11001000 00010111 00011*** *******	3
otherwise	4

Network Layer | Provides Routing and Forwarding Functionality

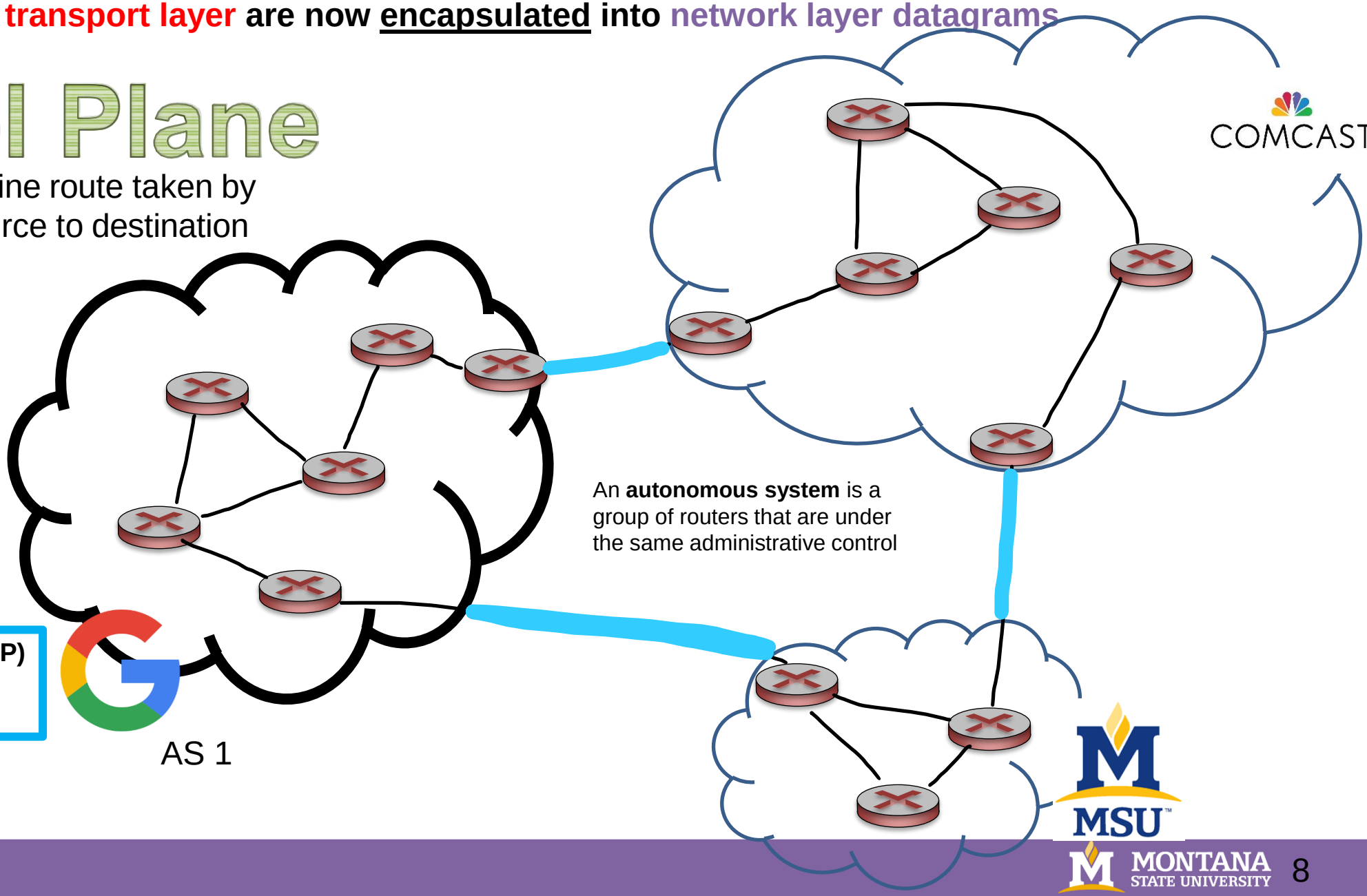
Our segments from the transport layer are now encapsulated into network layer datagrams

Control Plane

Routing: determine route taken by packets from source to destination

OSPF is a link-state protocol that uses flooding of link-state information and Dijkstra's least-cost algorithm
→ Used for routing *within* an AS

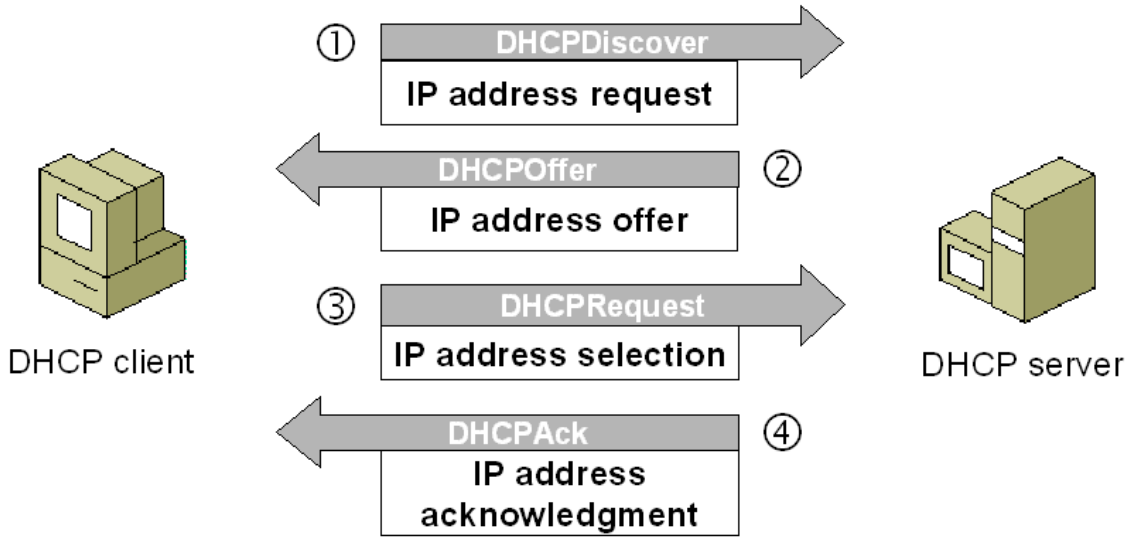
Border Gateway Protocol (BGP) is used for exchanging routing information *between* AS



Dynamic Host Configuration Protocol (DHCP) is a **plug-and-play**, client-server protocol that allows a host to obtain an IP address automatically

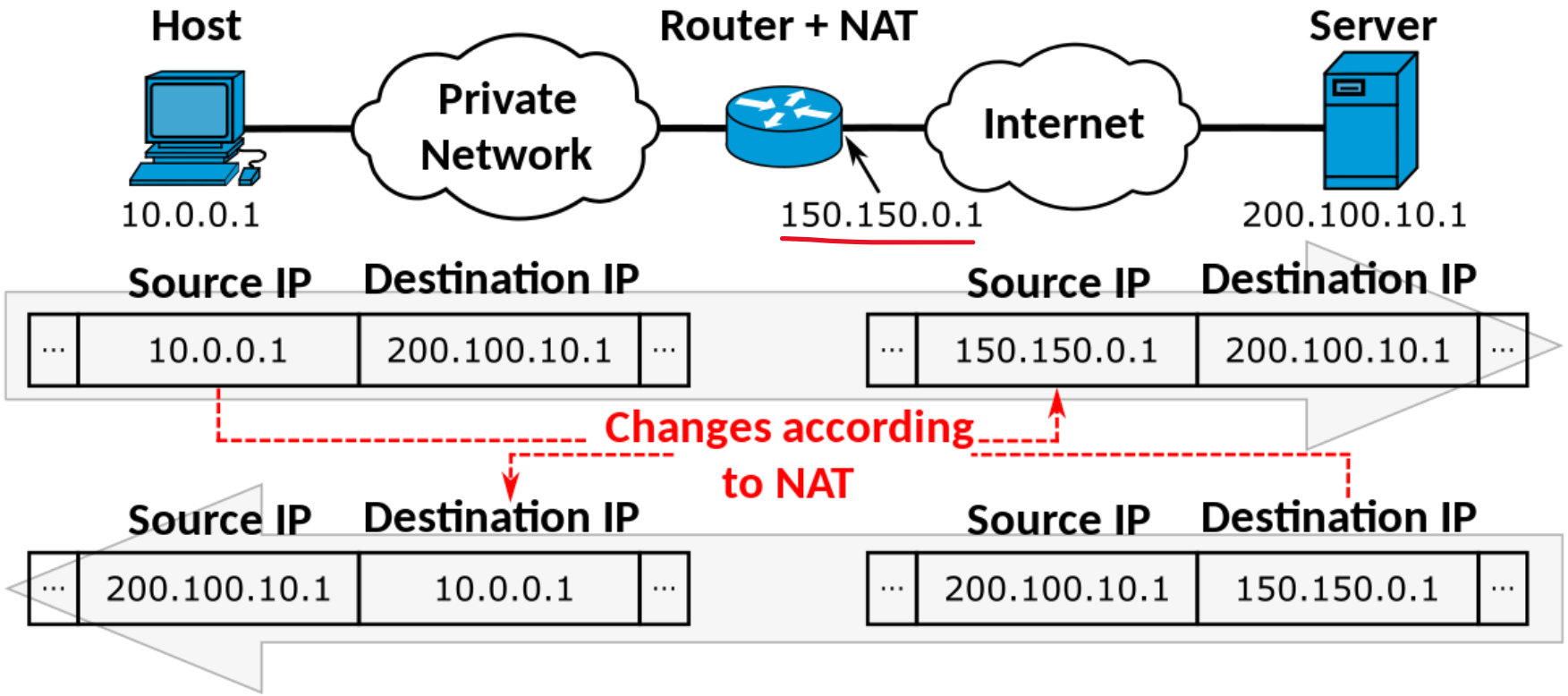
When a host is automatically assigned an IP address, it might keep that one forever, or the IP addresses can be temporary

(more common)



NAT is a translation of multiple private IP addresses to one single public IP address

- Hides details of inner home network from outside world
- All incoming traffic will have same public IP, all outgoing will have same public IP



Link Layer

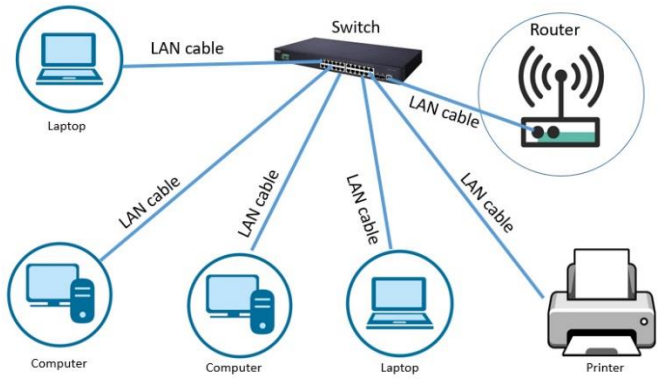
The link layer is responsible for the **actual node-to-node delivery** of data and ensure error-free transmission of information (handles a variety of mediums)

MAC (Media Access Control) Addresses

- function: *used "locally" to get frame from one interface to another physically-connected interface (**same network**, in IP-addressing sense)*
- 48 bit MAC address (for most LANs) burned in NIC ROM, also sometimes software settable

Address Resolution Protocol (ARP)

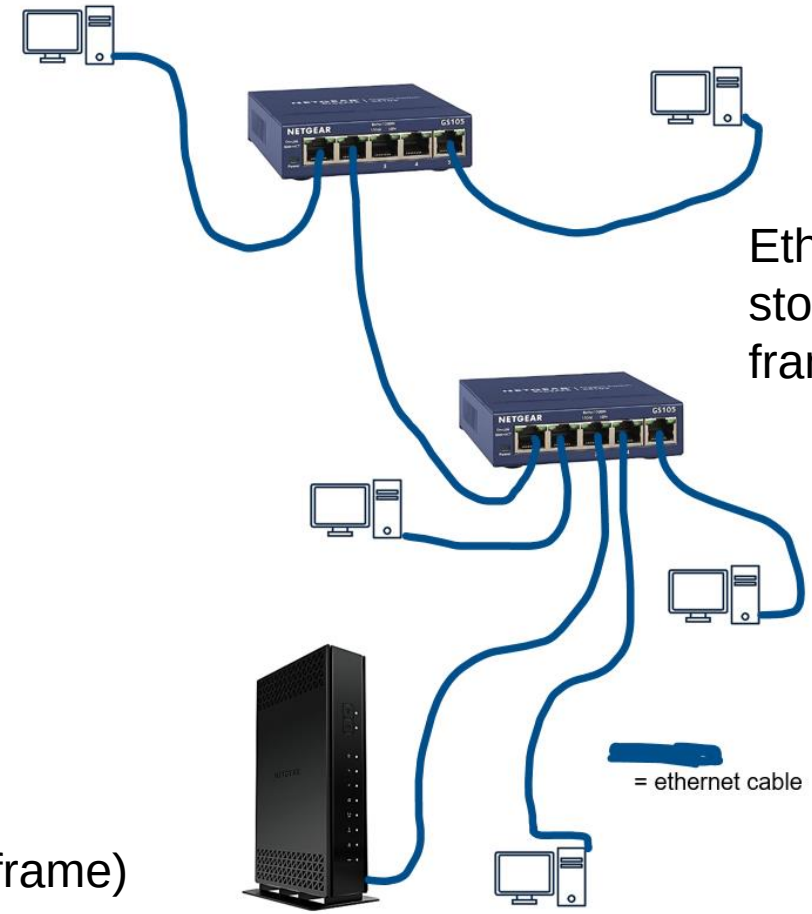
- Protocol used to map IP address to MAC addresses
- Very commonly used, as we need the MAC address of each host in our path
- Broadcast sent on all interfaces



Local Area Network

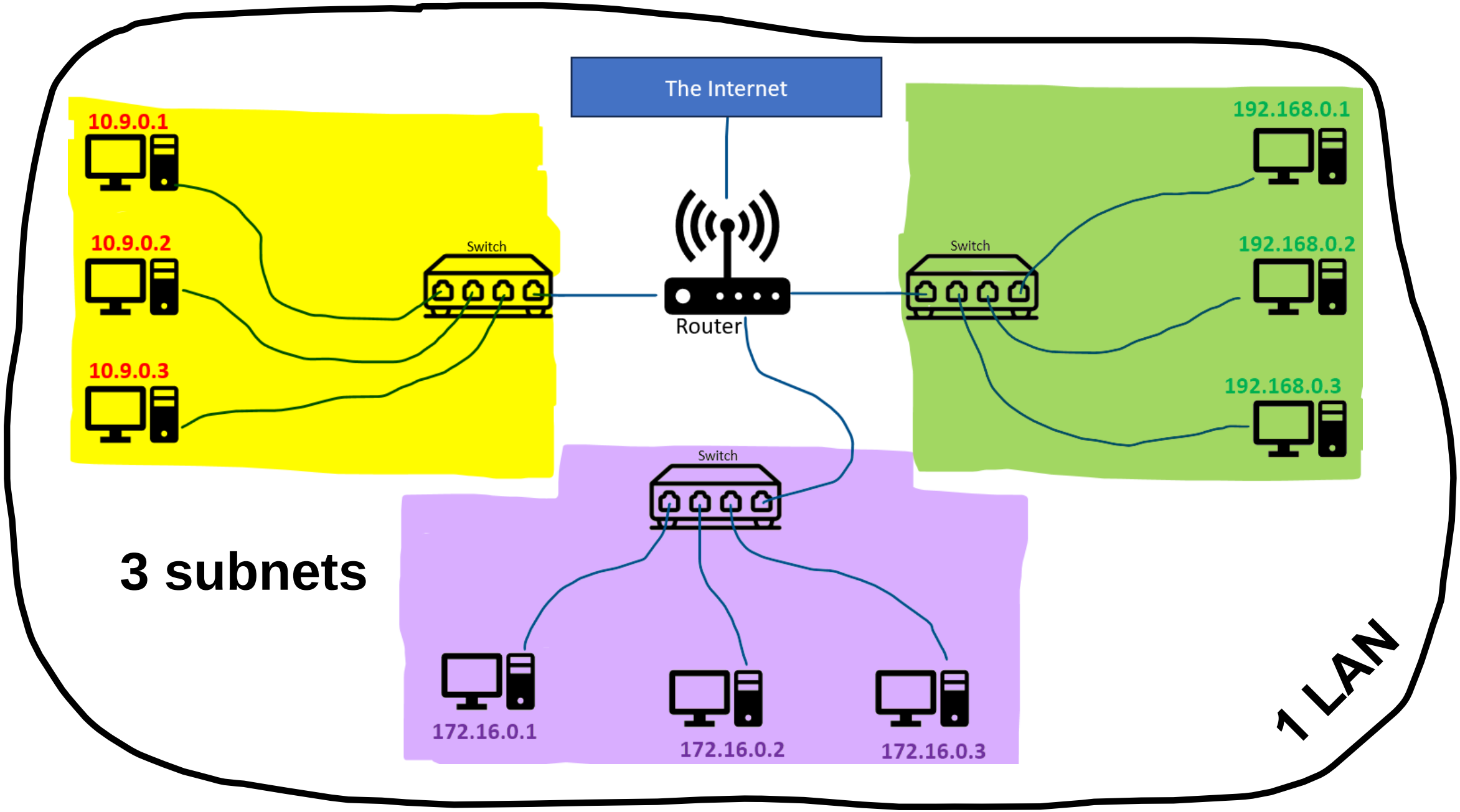
(Wi-fi = wifi frame)

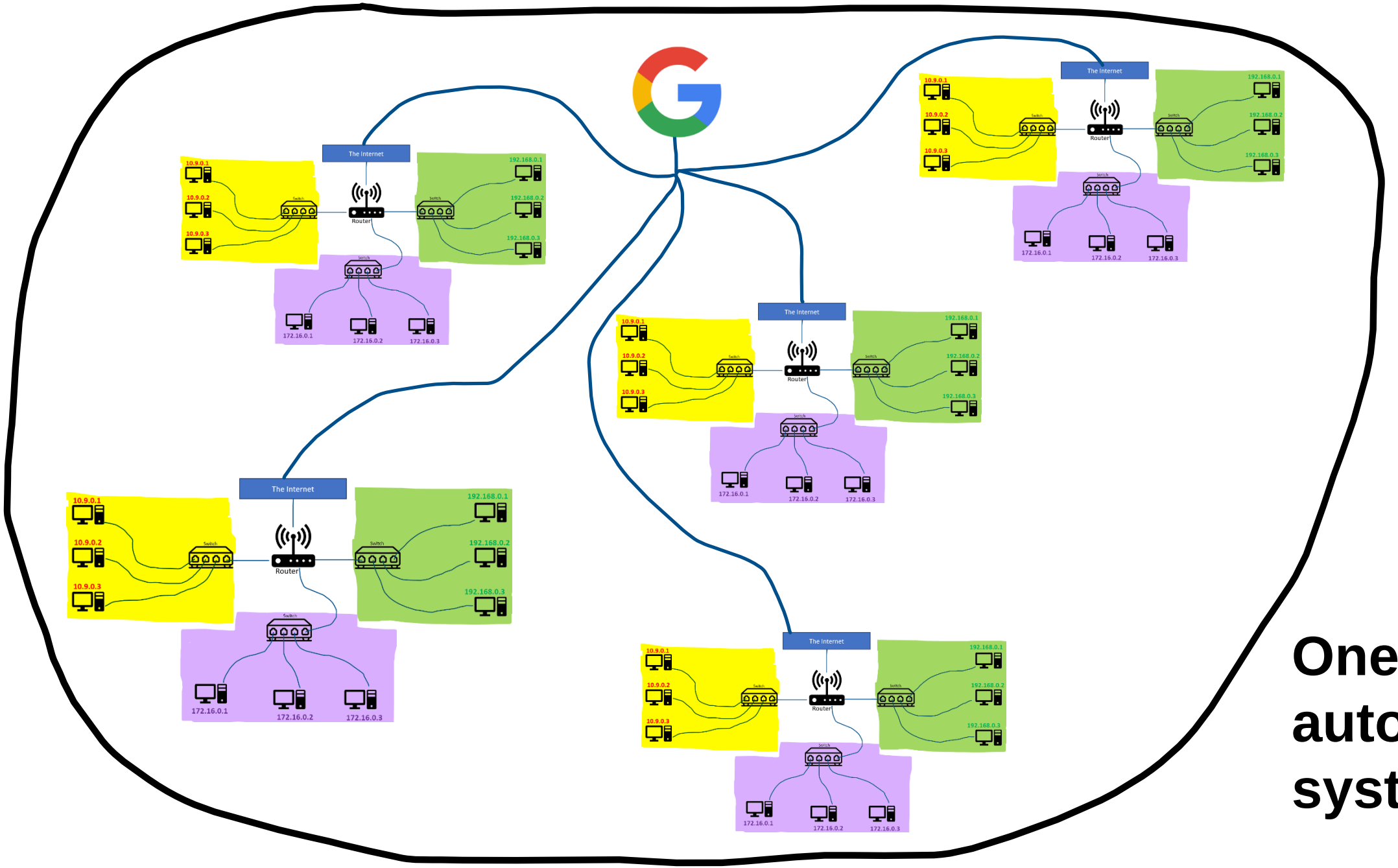
Ethernet: Dominant wired LAN technology



Ethernet **switches** will store and forward ethernet frames

Routers will forward frames leaving the LAN


The Internet
10.9.0.1
10.9.0.2
10.9.0.3
Switch
Router
192.168.0.1
192.168.0.2
192.168.0.3
Switch
Switch
172.16.0.1
172.16.0.2
172.16.0.3
3 subnets
1 LAN

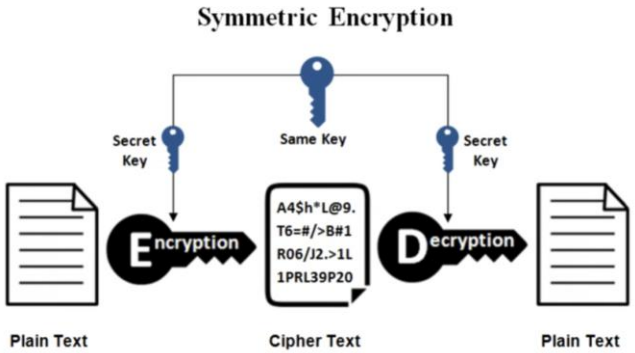


One autonomous system

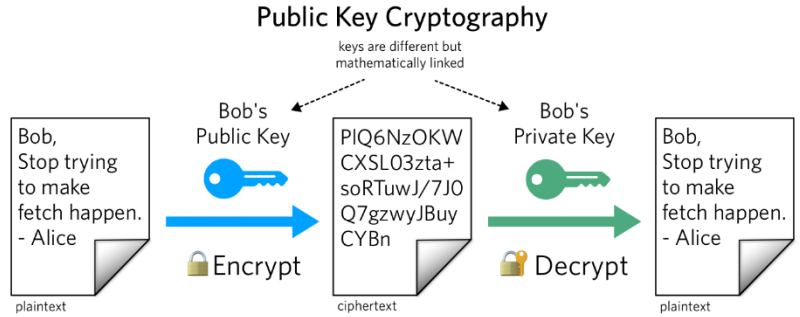
Confidentiality- Making sure that only the sender and receiver can read the message (encryption)

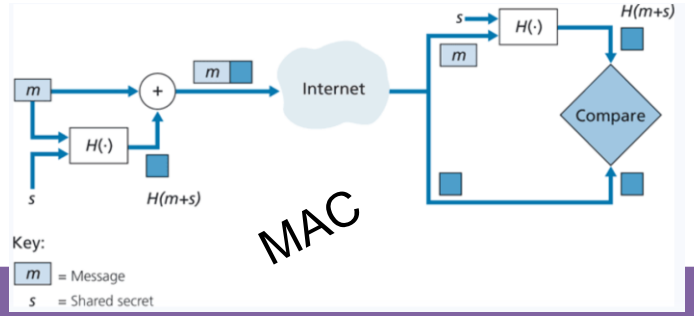
Authentication- Making sure that you are communicating with the person you think you are (encryption + hashing)

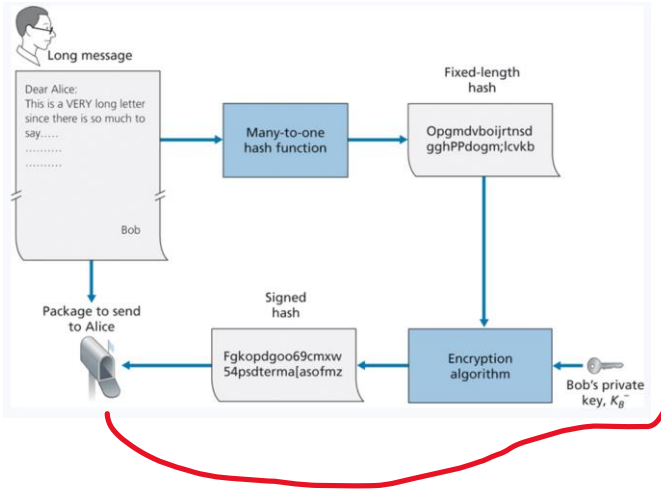
Integrity- Making sure the message does not get tampered with during transmission (hashing)



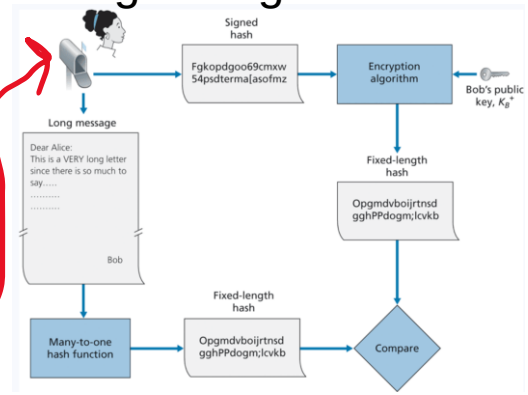
All this is implemented in **TLS/SSL** at the Session + Presentation Layer







Digital Signatures



OSI MODEL

Application Layer

Provides protocols for sending and receiving data between services and web applications (HTTP)

Messages

OSI MODEL

Application Layer	Provides protocols for sending and receiving data between services and web applications (HTTP)	**Messages**
Presentation Layer	Encoding, Compressing, and Encrypting Data	**Messages**

OSI MODEL

Application Layer	Provides protocols for sending and receiving data between services and web applications (HTTP)	**Messages**
Presentation Layer	Encoding, Compressing, and Encrypting Data	**Messages**
Session Layer	Authentication. Manages, monitors, “sessions” between endpoints	**Messages**

OSI MODEL

Application Layer	Provides protocols for sending and receiving data between services and web applications (HTTP)	**Messages**
Presentation Layer	Encoding, Compressing, and Encrypting Data	**Messages**
Session Layer	Authentication. Manages, monitors, “sessions” between endpoints	**Messages**
Transport Layer	Provides host-to-host, **reliable data transfer**, and dictates the flow of data	**Segments**

OSI MODEL

Application Layer	Provides protocols for sending and receiving data between services and web applications (HTTP)	**Messages**
Presentation Layer	Encoding, Compressing, and Encrypting Data	**Messages**
Session Layer	Authentication. Manages, monitors, “sessions” between endpoints	**Messages**
Transport Layer	Provides host-to-host, **reliable data transfer**, and dictates the flow of data	**Segments**
Network Layer	Forwarding and Routing of Data. Logical Addressing	**Datagrams**

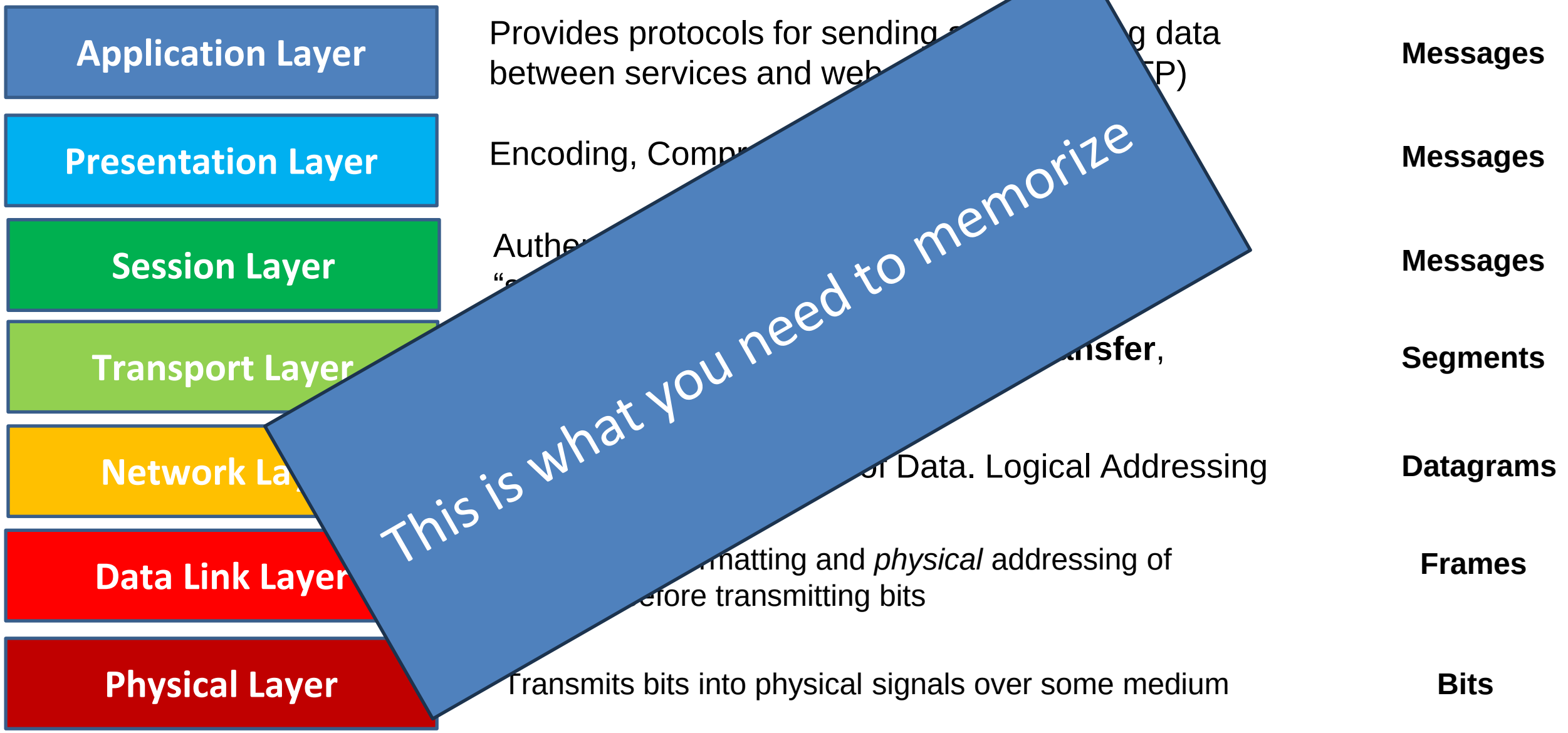
OSI MODEL

Application Layer	Provides protocols for sending and receiving data between services and web applications (HTTP)	**Messages**
Presentation Layer	Encoding, Compressing, and Encrypting Data	**Messages**
Session Layer	Authentication. Manages, monitors, “sessions” between endpoints	**Messages**
Transport Layer	Provides host-to-host, **reliable data transfer**, and dictates the flow of data	**Segments**
Network Layer	Forwarding and Routing of Data. Logical Addressing	**Datagrams**
Data Link Layer	Handles the formatting and *physical* addressing of the data before transmitting bits	**Frames**

OSI MODEL

Application Layer	Provides protocols for sending and receiving data between services and web applications (HTTP)	**Messages**
Presentation Layer	Encoding, Compressing, and Encrypting Data	**Messages**
Session Layer	Authentication. Manages, monitors, “sessions” between endpoints	**Messages**
Transport Layer	Provides host-to-host, **reliable data transfer**, and dictates the flow of data	**Segments**
Network Layer	Forwarding and Routing of Data. Logical Addressing	**Datagrams**
Data Link Layer	Handles the formatting and *physical* addressing of the data before transmitting bits	**Frames**
Physical Layer	Transmits bits into physical signals over some medium	**Bits**

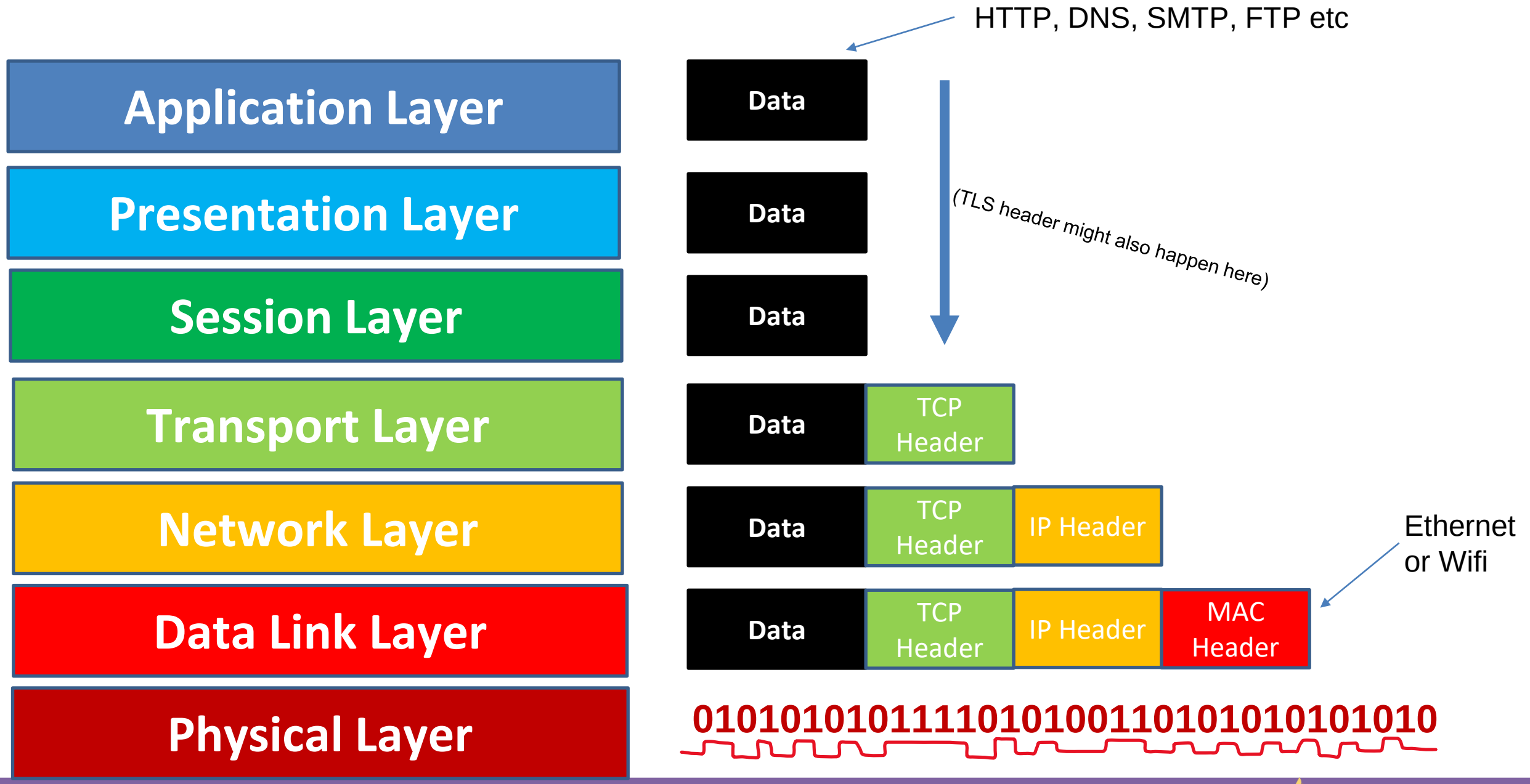
OSI MODEL


Application Layer
Presentation Layer
Session Layer
Transport Layer
Network La
Data Link Layer
Physical Layer
Provides protocols for sending
g data
between services and web
TP)
Encoding, Compr
Authe
“s
nsfer,
of Data. Logical Addressing
matting and *physical* addressing of
efore transmitting bits
Transmits bits into physical signals over some medium
Messages
Messages
Messages
Segments
Datagrams
Frames
Bits
This is what you need to memorize

OSI MODEL

Application Layer	A	Away
Presentation Layer	Penguin	Pizza
Session Layer	Said	Sausage
Transport Layer	That	Throw
Network Layer	Nobody	Not
Data Link Layer	Drinks	Do
Physical Layer	Pepsi	Please

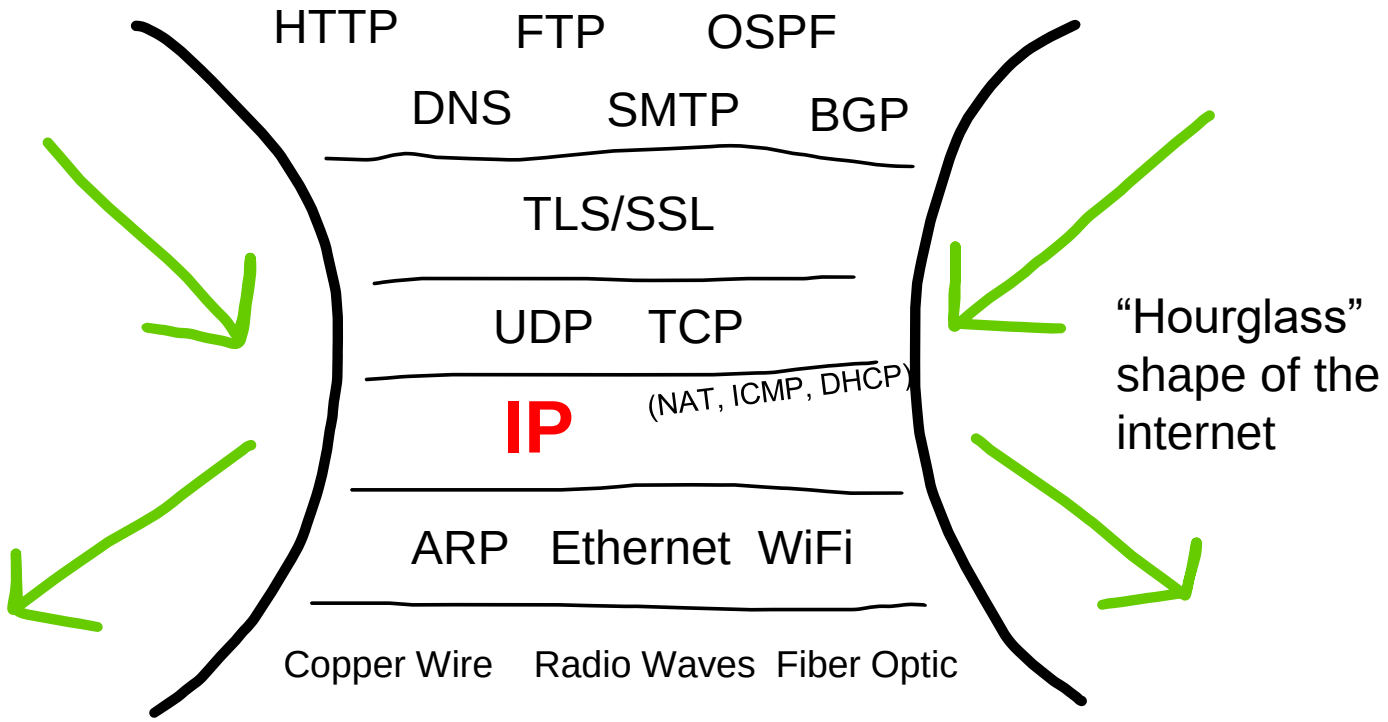
OSI MODEL


Application Layer
Presentation Layer
Session Layer
Transport Layer
Network Layer
Data Link Layer
Physical Layer
HTTP, DNS, SMTP, FTP etc
Data
Data
Data
(TLS header might also happen here)
Data
TCP Header
Data
TCP Header
IP Header
Ethernet or Wifi
Data
TCP Header
IP Header
MAC Header
01010101011110101001101010101010

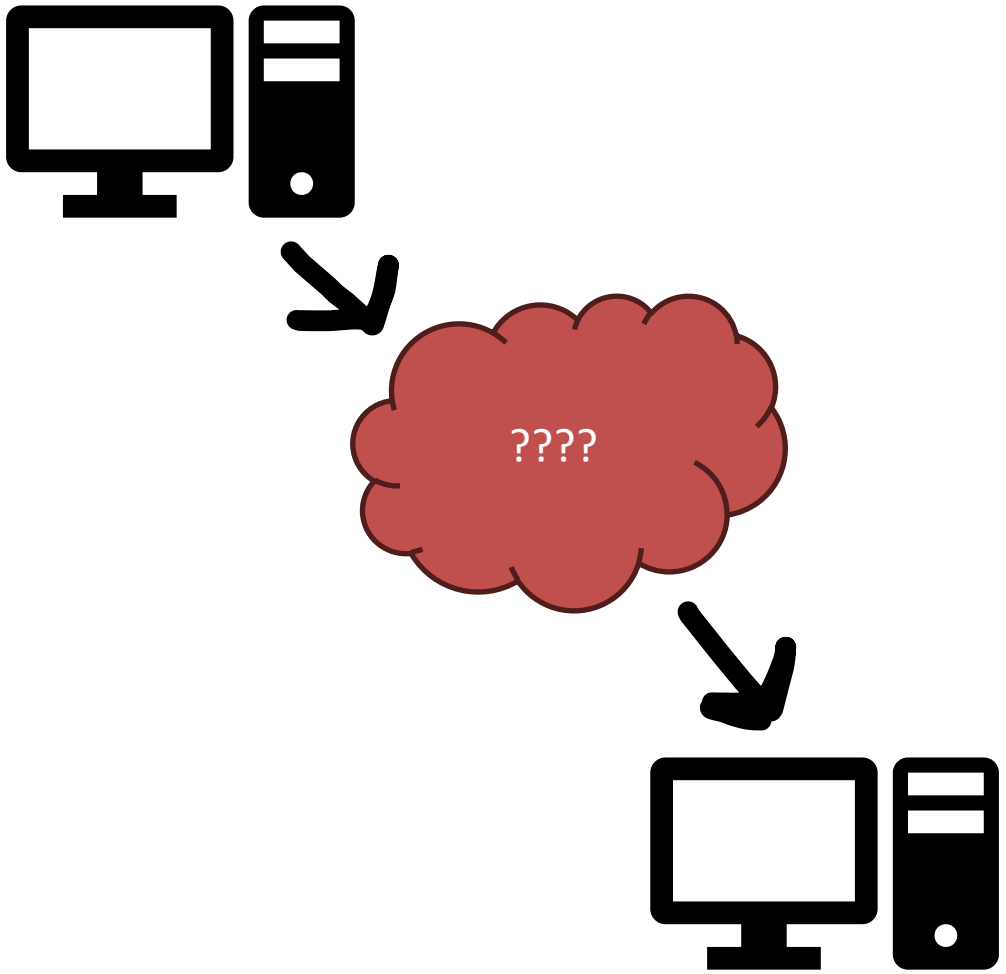
Course Outcomes

By the end of this course, students should be be able to:

- List the network layers and explain their function in end-to-end communications
- Explain different network architectures and tradeoffs in the design decisions
- Explain the functions of various Network protocols (HTTP, DNS, TCP/IP, FTP, etc)
- Design and implement network application
- Analyze network traffic
- Measure network performance
- Explore security issues in networks and understand important defense mechanisms



"Hourglass" shape of the internet



Any Questions?

Thank You!

Thank you for your patience, flexibility, and kindness ☺

I know things were not perfect, but I am happy with how things went

I hope you enjoyed this class, and I hope the stuff you learned will be helpful in your career. With most devices connected to the internet, its important to understand these basic networks concepts!

I will be teaching 232 and 132 next semester 😎

Connect with me on LinkedIn!

If I can be of assistance to you for anything in the future, please let me know!

Congrats to those that are graduating next weekend! I hope you find a job that you love!