EECS 388

Introduction to Computer Security

Lecture 11:

Networking 101

October 1, 2024
Prof. Chen

Web and Network Security

Last two weeks:

- The Web Platform
- Web Attacks and Defenses
- HTTPS and the Web PKI
- HTTPS Attacks and Defenses

This week:

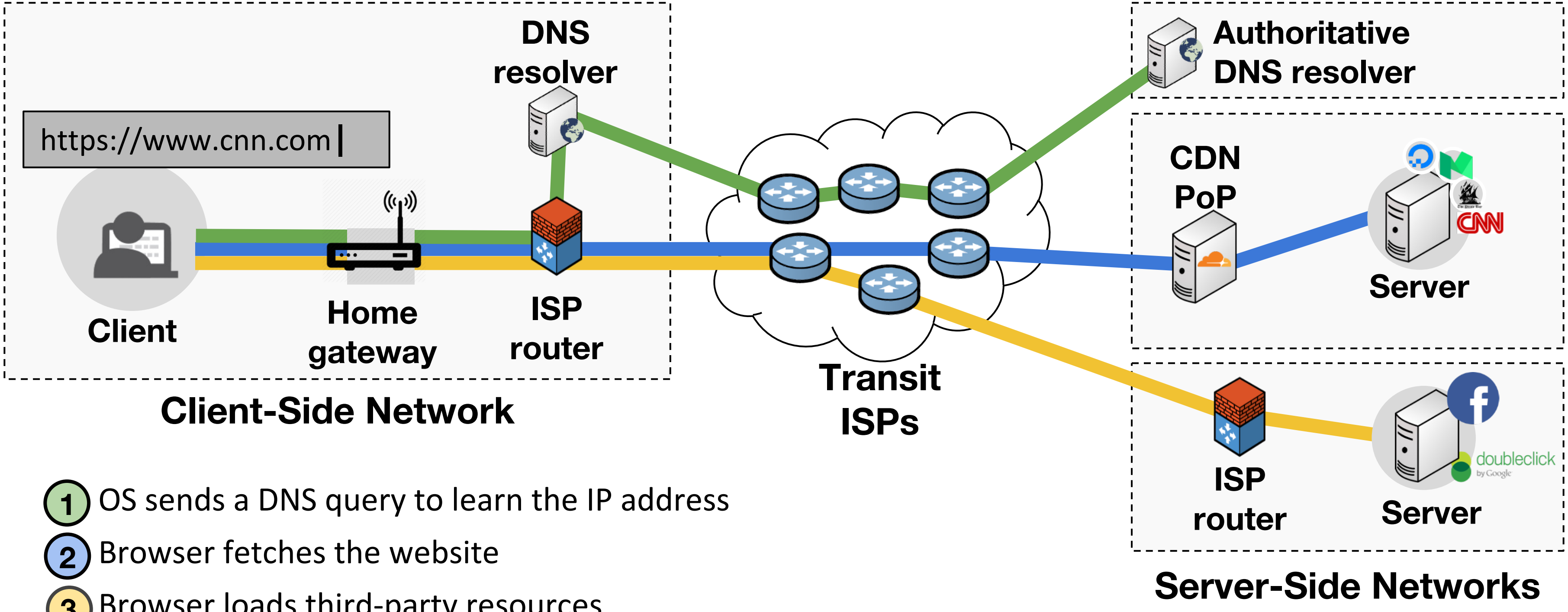
- **Networking 101** (Take EECS 489 for networks in-depth!)
- Networking 102 (We couldn’t squeeze it all into one lecture)

Later:

- Network Defense
- Privacy and Anonymity
- Censorship and Circumvention

The Internet: A Network of Networks

What happens when a user visits www.cnn.com in the browser?



1 OS sends a DNS query to learn the IP address

2 Browser fetches the website

3 Browser loads third-party resources

Internet Concepts

The Internet is a global network that provides **best-effort** delivery of **packets** between connected **hosts**.

A packet is a short, structured sequence of bytes:

header: metadata used by network

payload: the data to be transported

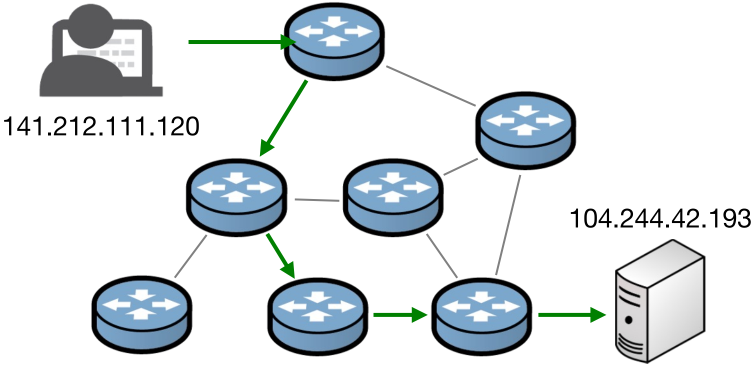
Best-effort: Any packets might get **dropped**.

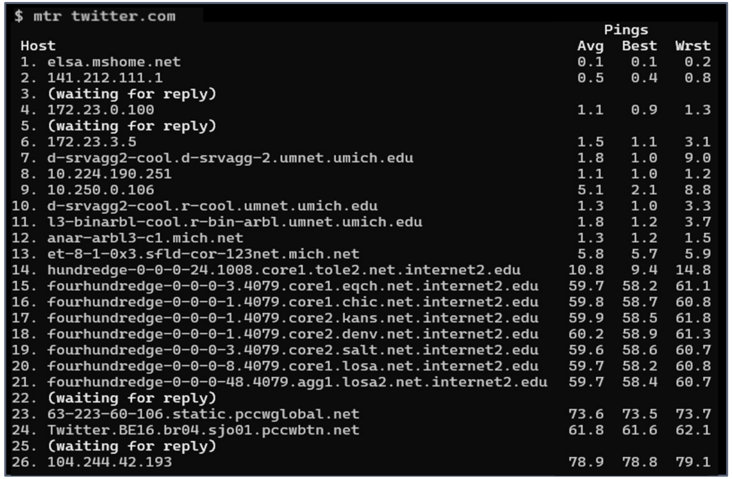
Every host has a unique identifier (IP address).

Internet routing: A series of **routers** receive packets:

1. Look at the destination address in packet header
2. Send packet **one hop** towards the destination

Example: Campus to twitter.com takes 26 hops





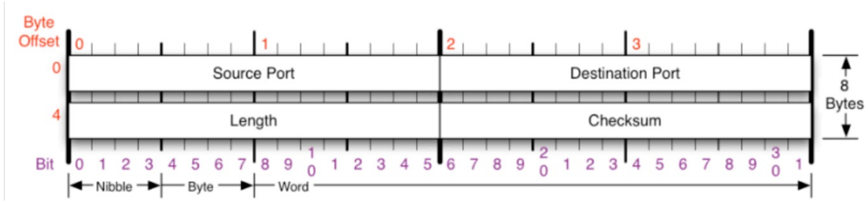
```
$ mtr twitter.com
                                                                  Pings
 Host                                                        Avg  Best  Wrst
 1. elsa.mshome.net                                          0.1   0.1   0.2
 2. 141.212.111.1                                            0.5   0.4   0.8
 3. (waiting for reply)
 4. 172.23.0.100                                             1.1   0.9   1.3
 5. (waiting for reply)
 6. 172.23.3.5                                               1.5   1.1   3.1
 7. d-srvagg2-cool.d-srvagg-2.umnet.umich.edu                1.8   1.0   9.0
 8. 10.224.190.251                                           1.1   1.0   1.2
 9. 10.250.0.106                                             5.1   2.1   8.8
10. d-srvagg2-cool.r-cool.umnet.umich.edu                    1.3   1.0   3.3
11. l3-binarbl-cool.r-bin-arbl.umnet.umich.edu               1.8   1.2   3.7
12. anar-arbl3-c1.mich.net                                   1.3   1.2   1.5
13. et-8-1-0x3.sfld-cor-123net.mich.net                      5.8   5.7   5.9
14. hundredge-0-0-0-24.1008.core1.tole2.net.internet2.edu   10.8   9.4  14.8
15. fourhundredge-0-0-0-3.4079.core1.eqch.net.internet2.edu 59.7  58.2  61.1
16. fourhundredge-0-0-0-1.4079.core1.chic.net.internet2.edu 59.8  58.7  60.8
17. fourhundredge-0-0-0-1.4079.core2.kans.net.internet2.edu 59.9  58.5  61.8
18. fourhundredge-0-0-0-1.4079.core2.denv.net.internet2.edu 60.2  58.9  61.3
19. fourhundredge-0-0-0-3.4079.core2.salt.net.internet2.edu 59.6  58.6  60.7
20. fourhundredge-0-0-0-8.4079.core1.losa.net.internet2.edu 59.7  58.2  60.8
21. fourhundredge-0-0-0-48.4079.agg1.losa2.net.internet2.edu 59.7  58.4  60.7
22. (waiting for reply)
23. 63-223-60-106.static.pccwglobal.net                     73.6  73.5  73.7
24. Twitter.BE16.br04.sjo01.pccwbtn.net                     61.8  61.6  62.1
25. (waiting for reply)
26. 104.244.42.193                                          78.9  78.8  79.1
```

Network Protocols

Network protocols define how hosts communicate. They specify:

Syntax: How communication is *structured*.

Data format and order of messages



Example: Packet header data structure

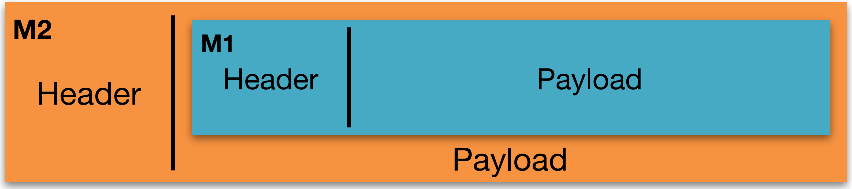
Semantics: What communication *means*.

Actions on transmit/receipt of message or timeout. What assumptions can be made.

Internet protocols are **open standards**, specified in Requests for Comment (RFCs)

Protocols are often used together, via the mechanism of encapsulation.

A protocol P1 can draw on services from a "lower-layer" protocol P2:



M2

Header

M1

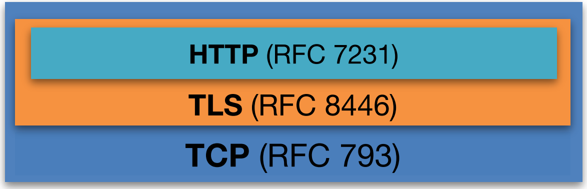
Header

Payload

Payload

Message M1 of P1 is **encapsulated** into a message M2 of P2 by setting **payload** of M2 to bytes of M1.

Example: **HTTPS**



HTTP (RFC 7231)

TLS (RFC 8446)

TCP (RFC 793)

Protocol Layering

Networks use a stack of **protocol layers**.

Layers define **abstraction boundaries** and give each layer its own responsibilities (i.e., “**separation of concerns**”).

At a given layer, all layers above and below are opaque:

- **Lower layers** provide services to layers above (don’t care what they do).
- **Higher layers** use services of layers below (don’t care how they work).

For Internet applications, we commonly use the **five-layer reference model**.*

* This model is simpler than the 7-layer OSI model taught in 489.

Layer 5 **Application**	Defines how individual applications communicate (e.g., HTTP, SSH, DNS)
Layer 4 **Transport**	Adds features (ports, connections, encryption) on top of bare packets (e.g., UDP, TCP, TLS, QUIC)
Layer 3 **Network**	Gets packets to the final destination, over arbitrarily many hops (e.g., IP)
Layer 2 **Link**	Provides a point-to-point link to get packets to next hop (e.g., Ethernet)
Layer 1 **Physical**	How bits get translated into electrical, optical, or radio signals

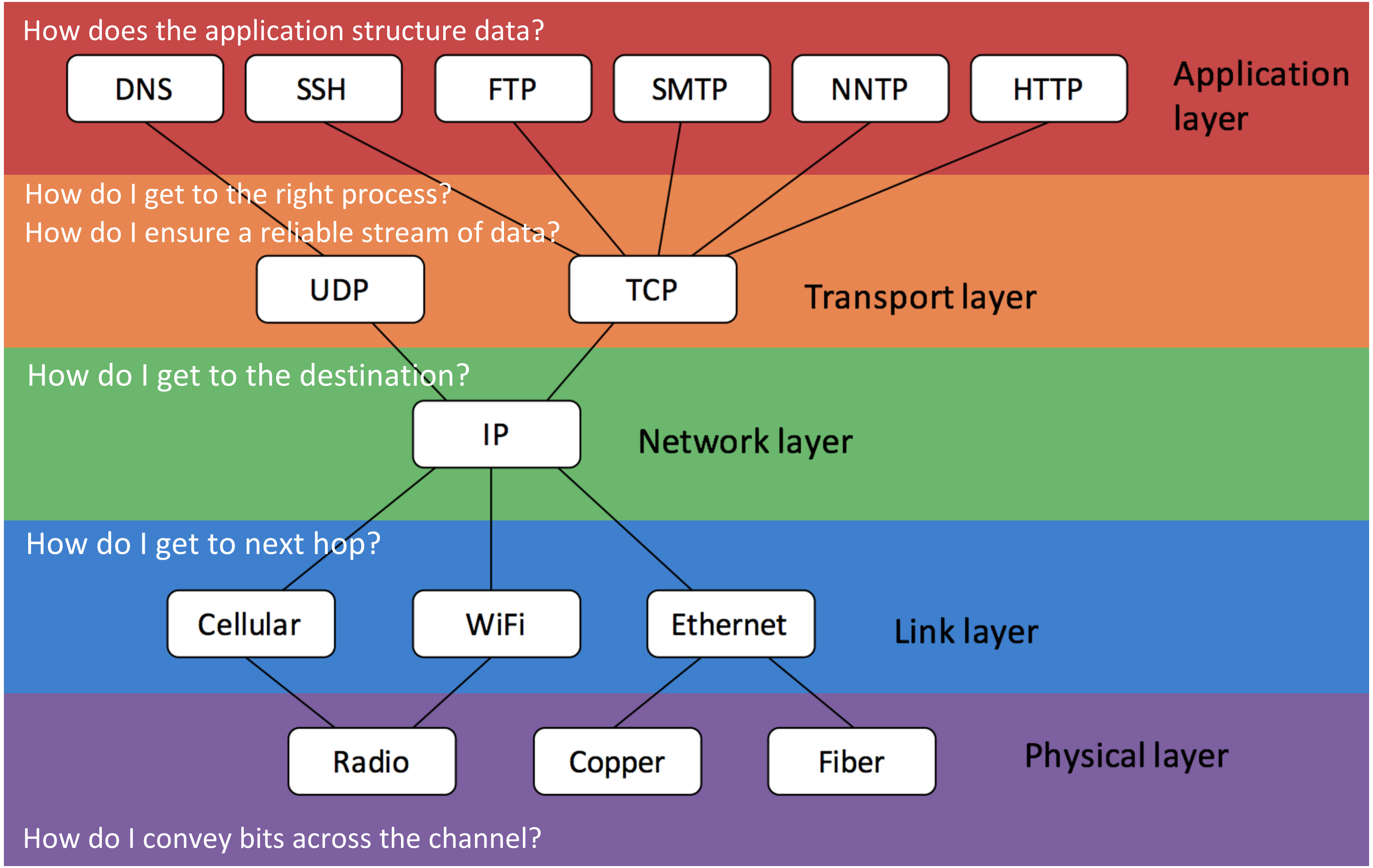
Modularity and Interoperability

Network layering allows **modularity**:

- Many applications
- Various transport services
- Variety of point- to-point links

IP is the "**thin waist**" of this hourglass-shaped architecture.

This helps ensure that Internet hosts can interoperate.

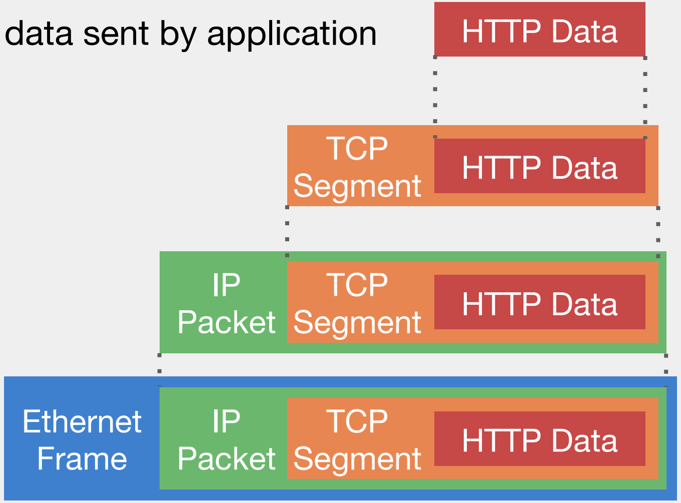

How does the application structure data?
DNS
SSH
FTP
SMTP
NNTP
HTTP
Application layer
How do I get to the right process?
How do I ensure a reliable stream of data?
UDP
TCP
Transport layer
How do I get to the destination?
IP
Network layer
How do I get to next hop?
Cellular
WiFi
Ethernet
Link layer
Radio
Copper
Fiber
Physical layer
How do I convey bits across the channel?

Internet Protocol Encapsulation

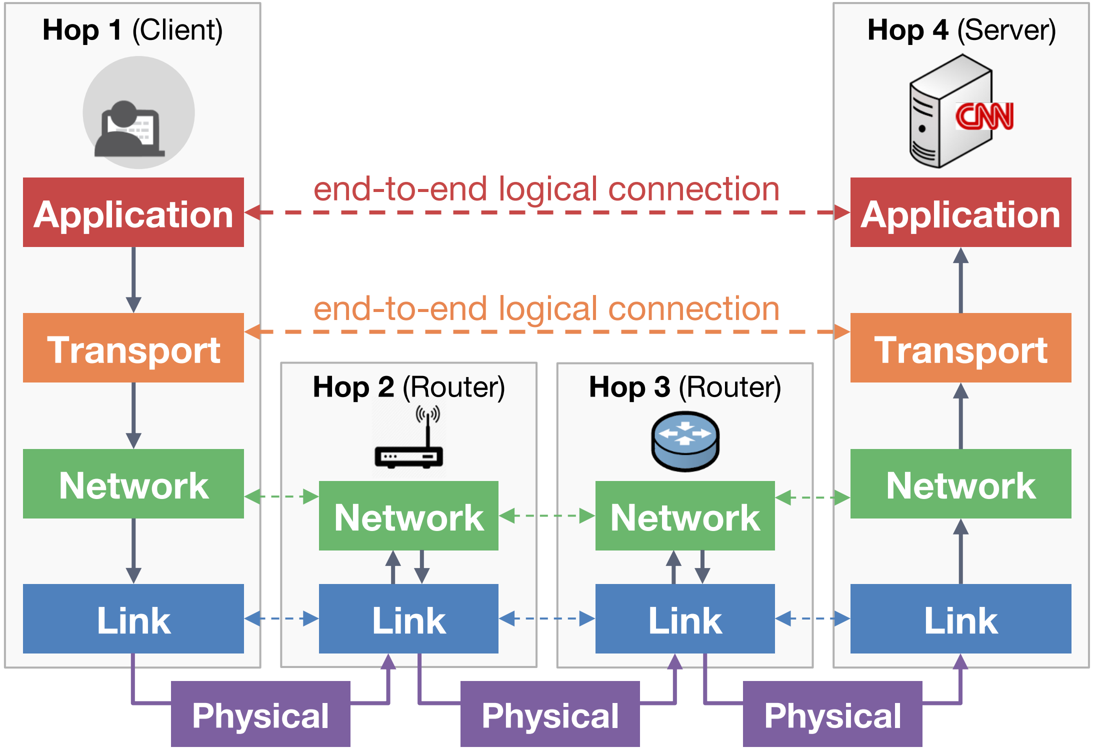
Each lower-layer protocol further encapsulates the application data.

The layers are later decapsulated in **reverse order**.

data sent by application



encapsulated data sent over physical layer



Network Threats

Common protocol vulnerabilities:

Plaintext transmission
Passive attackers can eavesdrop on unencrypted communication.

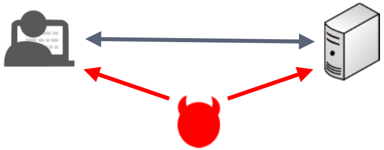
No source authentication
The source address on packets you receive can be spoofed, can't trust it.

No cryptographic integrity
Protocols don't prevent data modification. (Checksums stop data corruption, not attacks.)

No built-in bandwidth control
Attacks can deny service by flooding hosts or the network itself with traffic.

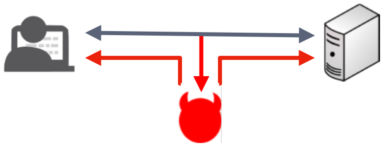
Network threat models:

Off-path attacker. Network participant. Can talk to hosts, but typically can't see victim's packets.



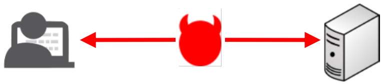
Most attackers, initially

On-path attacker. Sees *copy* of victim's packets. Can add packets, but typically **can't** change/block.



WiFi sniffing or Ethernet passive tap

In-path attacker. MITM. Can see, add, change, or block victim's packets.



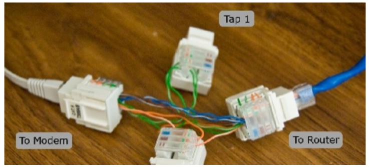
Obtain via ARP spoofing, BGP hijacking, DNS attacks

Physical Layer

Physical How bits get translated into electrical, optical, or radio signals

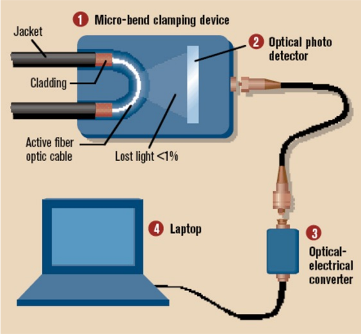
Wired electrical links

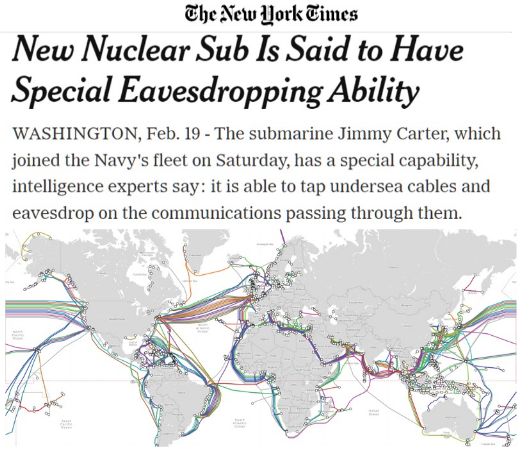
can be physically tapped to eavesdrop or inject data.



Fiber optic links

can be tapped too!





The New York Times

New Nuclear Sub Is Said to Have Special Eavesdropping Ability

WASHINGTON, Feb. 19 - The submarine Jimmy Carter, which joined the Navy's fleet on Saturday, has a special capability, intelligence experts say: it is able to tap undersea cables and eavesdrop on the communications passing through them.

Defenses:

Physical security? Encrypt at higher layers

Radio links (e.g., WiFi, Bluetooth, cellular)

Eavesdropping: aided by omnidirectional transmission. Very long range using capable receivers (e.g., distant drone).

Jamming: simple to deny availability using inexpensive (illegal) devices.

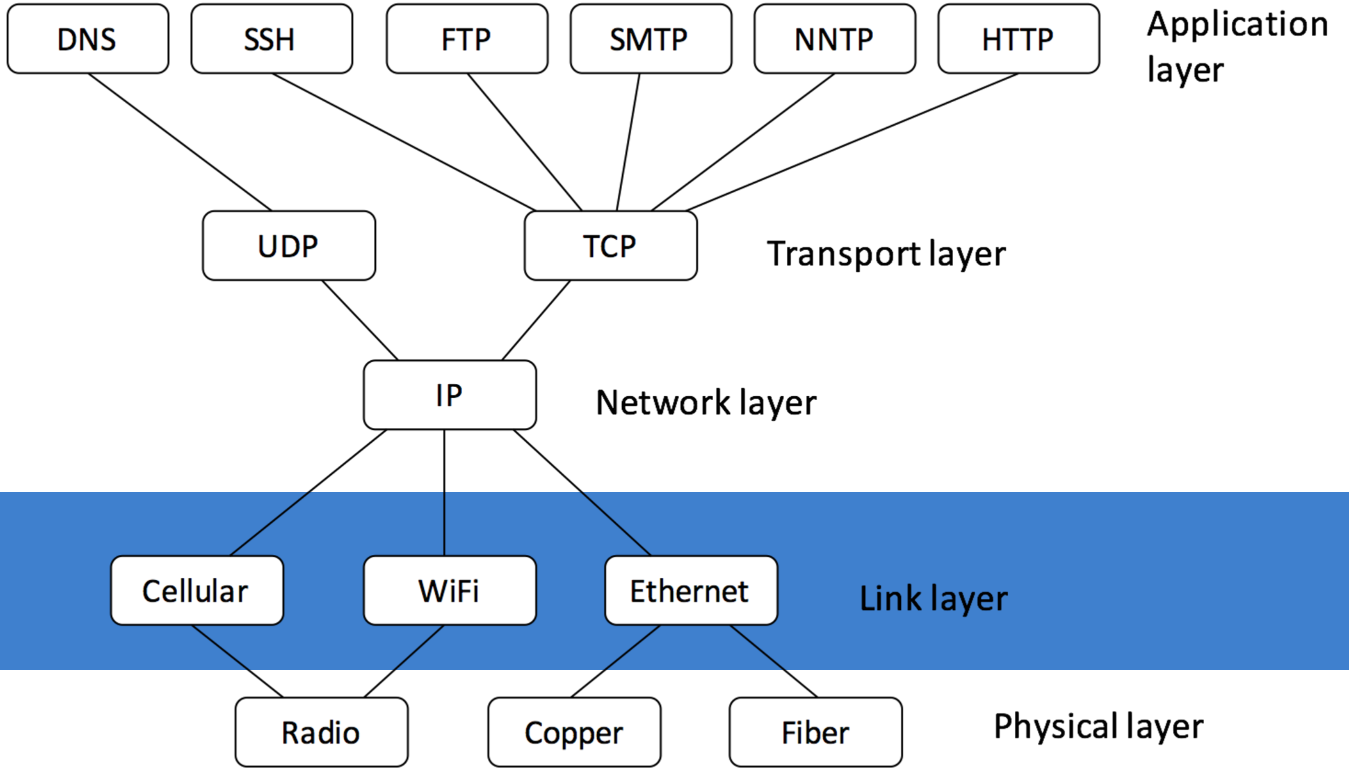
IMSI catchers: devices sold to law enforcement can simulate cell tower for surveillance purposes.

Weak encryption: WiFi and cellular support encryption, but designs are historically poor.

Broken: WEP, WPA, WPA2, 2G, 3G, 4G

Link Layer

Link	Provides a point-to-point link to get packets to next hop



How do I get to the next hop?

Link Layer

Link Provides a point-to-point link to get packets to next hop

Assumes: Nodes have a physical connection.

Task: Transfer bytes between two hosts on the local, physically connected network.

Ethernet: most common link-layer protocol. Send ~1500 byte packets (“**frames**”) to other hosts on a **local network**, addressed by MAC.

Operates over physical wired links or WiFi.

Frames are plaintext. They may be lost, reordered, corrupted, duplicated, or attacked.

Ethernet devices ship with globally unique* 48-bit **MAC address** “media access control”

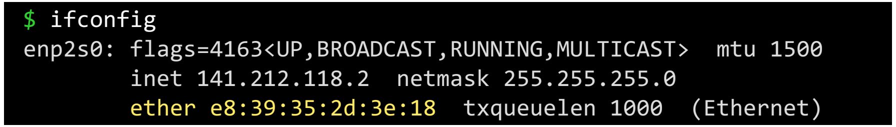
(Unrelated to “message authentication codes”, sorry!)

First three bytes identify device manufacturer.

EtherType field gives payload’s layer-3 protocol:

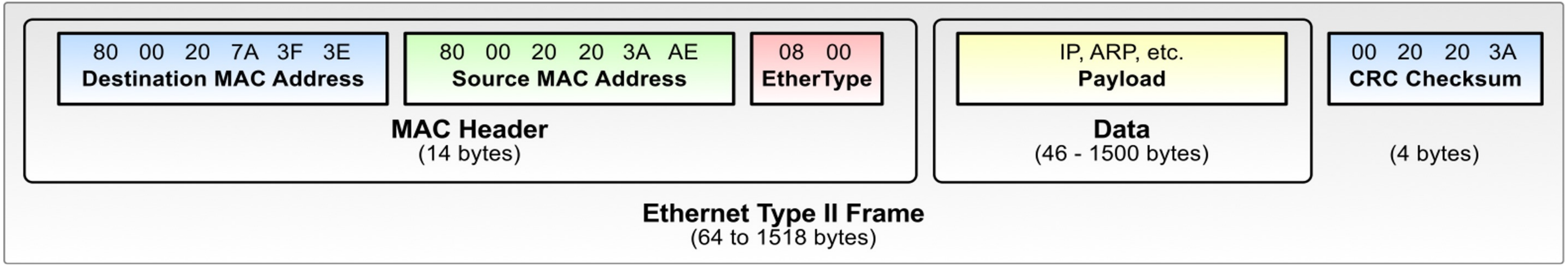
0x0800: IPv4 0x0806: ARP 0x86DD: IPv6

Run `ifconfig` to find your MAC address:



```
$ ifconfig
enp2s0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST>  mtu 1500
        inet 141.212.118.2  netmask 255.255.255.0
        ether e8:39:35:2d:3e:18  txqueuelen 1000  (Ethernet)
```

* Clients can **change their MACs** arbitrarily!



80 00 20 7A 3F 3E
Destination MAC Address

80 00 20 20 3A AE
Source MAC Address

08 00
EtherType

MAC Header
(14 bytes)

IP, ARP, etc.
Payload

Data
(46 - 1500 bytes)

00 20 20 3A
CRC Checksum
(4 bytes)

Ethernet Type II Frame
(64 to 1518 bytes)

Ethernet Switching

Link | Provides a point-to-point link to get packets to next hop

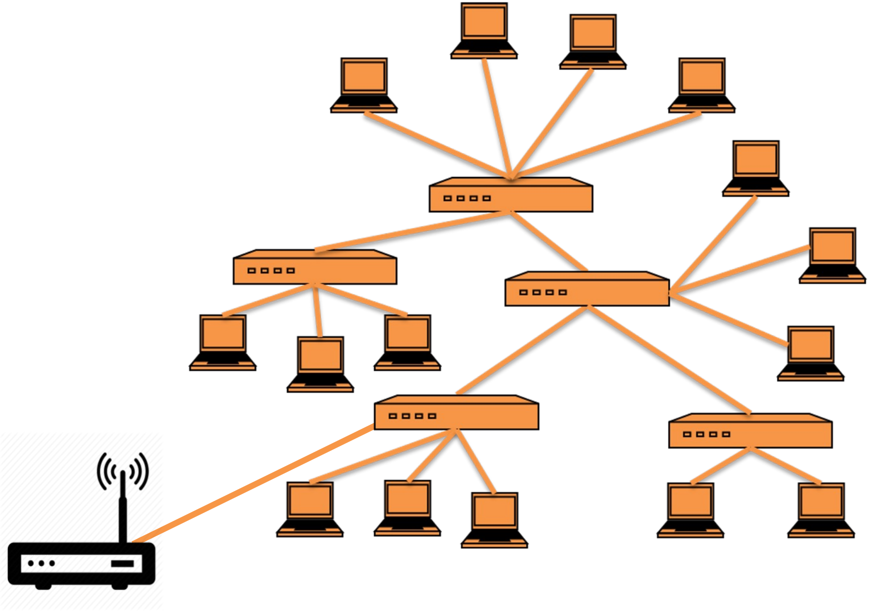
Each port on an **Ethernet switch** connects to a host **or another switch**. Basic algorithm:

- Switch **learns** what MACs are on each port by inspecting source addresses of sent frames.
- If switch knows MAC address M is at port P, sends frames destined for M only to port P.
- Otherwise, **broadcasts** the frame to all ports.

No guarantee frames not sent to other hosts!

[Why do we need IP for larger networks?]

Multiple switches can be arranged into a **tree** to form a larger **local-area network (LAN)**:

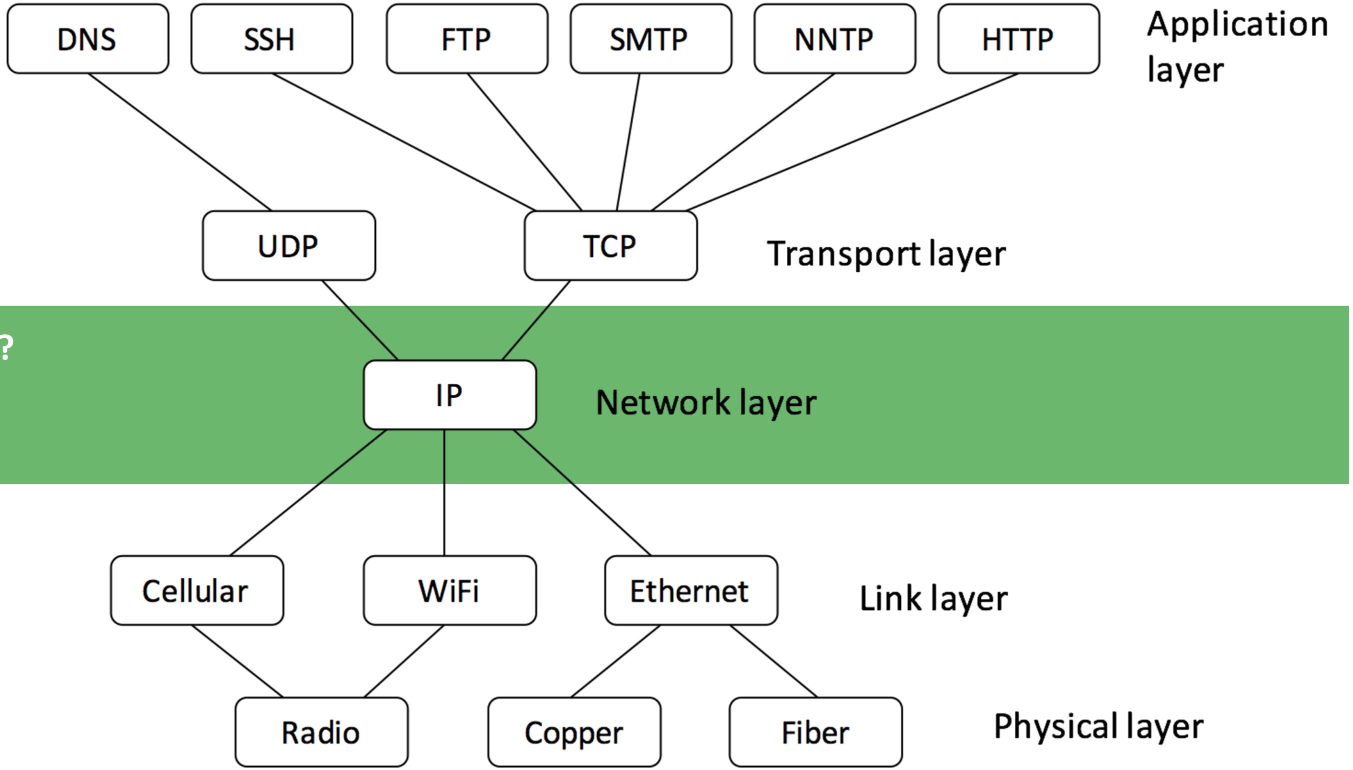


WiFi access points connect to the wired network and convey Ethernet frames via radio.

Network Layer

Network | Gets packets to final destination over arbitrarily many hops

How do I get to the destination?



DNS SSH FTP SMTP NNTP HTTP

Application layer

UDP TCP

Transport layer

IP

Network layer

Cellular WiFi Ethernet

Link layer

Radio Copper Fiber

Physical layer

Internet Protocol

Network	Gets packets to final destination over arbitrarily many hops

Internet Protocol (IP) delivers packets ("datagrams") to Internet destinations.

Encapsulates TCP and UDP (later).
Encapsulated in link-layer (Ethernet) frames.

Defines what packets must look like to be processed by routers.

Two version: **IPv4**, **IPv6**.

Routers forward an IP packet along to try to get it to the destination host.

Routers don't need to understand any other part of the packet.

IP provides:

- **Routing:** Get packet to destination IP address
- **Fragmentation and reassembly:** Split data into "right size" packets and reassemble

IP ***doesn't*** provide: **Everything else!**

- No ordering guarantees.
- No retransmission.
- No (real) error checking.
- No acknowledgement of receipt.
- No "connections."
- **No security.**

Packets are in plaintext. May be reordered, lost, corrupted, duplicated, or attacked.

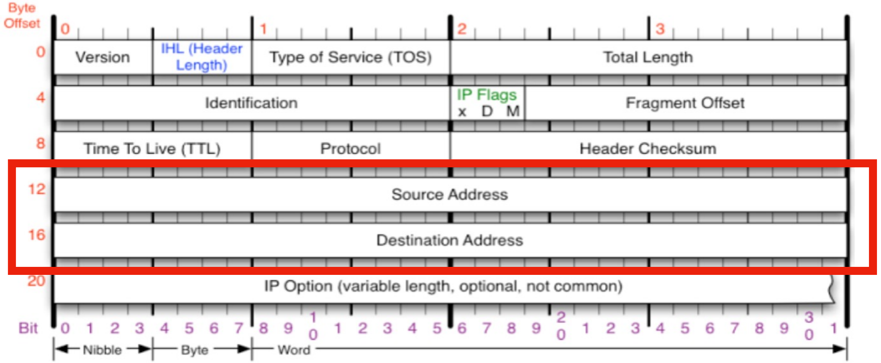
IP Header/Addresses

Network | Gets packets to final destination over arbitrarily many hops

Every packet starts with an **IP header** that tells routers/hosts what to do with the packet.

All values are filled in by sending host:

- Sender sets **source address**
  Source can be faked by attacker.
- Sender sets **destination address**
  Routers forward datagram toward that address.



Packets have a checksum, but it's not cryptographic.
MiTM can change any IP packet.

Every host assigned a (mostly) unique address:

IPv4 address: 32 bits
Written as 4 bytes in form `A.B.C.D` where A,...,D are 8-bit integers in decimal (called "dotted quad")
e.g., 192.168.1.1

IPv6 address: 128 bits
Written as 16 bytes in form `AA:BB::XX:YY:ZZ` where AA,...,ZZ are 16-bit integers in hexadecimal and :: implies zero bytes
e.g., 2620:0:e00:b::53 = 2620:0:e00:b:0:0:0:53

IP packet Spoofing

If a host sets a fake source address, the **"spoofed" packets** will reach destinations, but responses will be routed to the real hosts.

Network Gateways

Network | Gets packets to final destination over arbitrarily many hops

The Internet is a collection of interconnected layer-2 (link layer) networks (LANs).

IP addresses always contain two parts:

Network prefix: Used to identify the destination network (like a ZIP code).

Host suffix: Used to identify the destination host on that network (like a house number).

The network prefix may be any length.

To indicate which bits are part of the network prefix, we write the address in **CIDR notation** (e.g., 141.212.120.0/24) or provide an explicit **network mask** (e.g., 255.255.255.0).

Example: The network 141.212.120.0/24 consists of the 256 IP addresses 141.212.120.*.

An IP host must be provisioned with its own IP address and the IP address of a router on the local network (the "**network gateway**").

(These settings are often provided automatically by a local **DHCP server**.)

When sending a packet, the host compares its IP address to the destination's:

- **If network prefixes are the same:**
  Destination is on the same LAN.
  Send packet directly, in an Ethernet frame.
- **If network prefixes are different:**
  Destination is on a different LAN.
  Send packet to the gateway for onward delivery, in an Ethernet frame.

ARP and ARP Spoofing

Network | Gets packets to final destination over arbitrarily many hops

Issue: How does a host know what MAC address to use to reach a given IP address?

Address Resolution Protocol (ARP) is a layer-3 protocol for mapping IP addresses to MAC addresses:

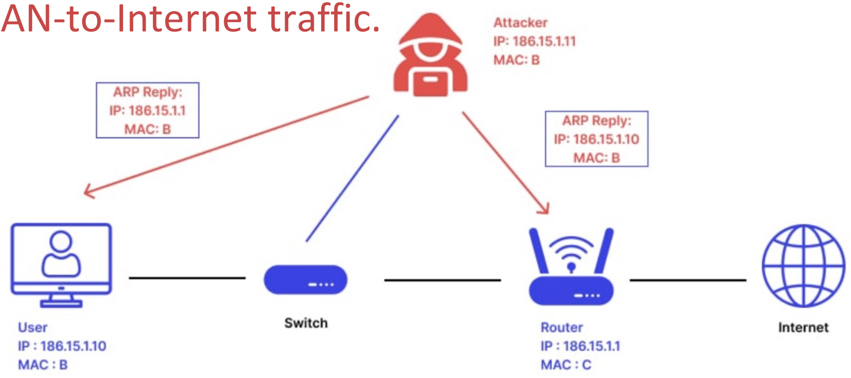
1. Host that needs MAC address M corresponding to IP address A ***broadcasts*** an **ARP packet** to entire LAN asking, *"who has IP address A?"*
2. Host that has IP address A will reply, *"IP address A is at MAC address M."*
3. Host H caches <IP A: MAC M>

Since any host on the LAN can send ARP requests and replies, any host can claim to be another host on the local network!

ARP Spoofing: Host X can force IP traffic between hosts A and B to flow through X:

- Claim IP_A is at attacker's MAC address M_X.
- Claim IP_B is at attacker's MAC address M_X.
- Re-send IP_A's traffic to M_A, and vice versa.

By **spoofing the gateway**, attacker can MITM all LAN-to-Internet traffic.

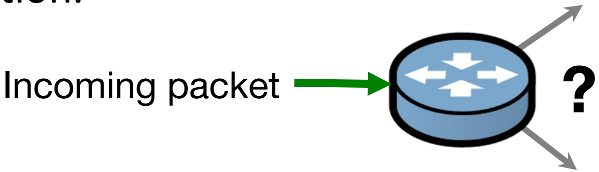


Routing and BGP

Network | Gets packets to final destination over arbitrarily many hops

A router is a device with multiple link layers, each part of a different network.

Examines incoming packet's destination IP and **forwards it** out a link that will get it closer to the destination.

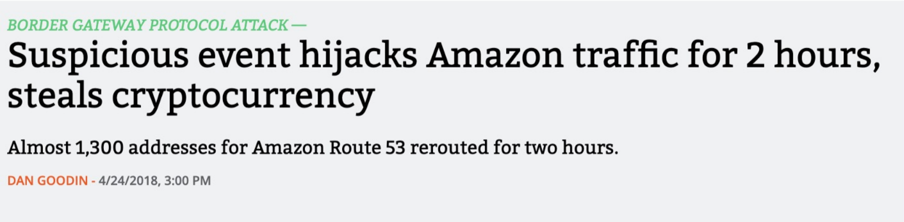


Issue: How does each router know where to send the packet next?

ISPs use Border Gateway Protocol (BGP) to **announce** their network prefixes and connectivity to neighbors. Routers compute **route tables** from these announcements.

BGP has no authentication: Compromised ISP can *announce someone else's network*!

BGP hijacking is common, usually due to operator error, but sometime malicious. Packets to hijacked network are routed to attacker, who can eavesdrop or MiTM.



BORDER GATEWAY PROTOCOL ATTACK —

Suspicious event hijacks Amazon traffic for 2 hours, steals cryptocurrency

Almost 1,300 addresses for Amazon Route 53 rerouted for two hours.

DAN GOODIN - 4/24/2018, 3:00 PM

Defense: RPKI

Cryptographic method of signing records that associate a BGP route announcement with the correct originating ISP. *Slow adoption…*

Private Addresses/NAT

Network Gets packets to final destination over arbitrarily many hops

IPv4 Address Exhaustion

All ~2^{32} IPv4 addresses have been assigned, current cost is ~$40/IP on secondary markets.

(IPv6 has ~2^{128} addresses–should be plenty!)

IPv4 Private Address Ranges

Three IPv4 networks are reserved for “private” use. ISPs won’t route them, but can assign to your own devices for communicating on LAN:

10.0.0.0/8	(2^{24} addresses)
172.16.0.0/12	(2^{20} addresses)
192.168.0.0/16	(2^{16} addresses)

IPv4 Loopback Address

127.0.0.1 (`localhost`) always refers to the local machine. Not reachable from anywhere else.

Network Address Translation (NAT)

Most residential ISPs assign customers only one IPv4 address.

Issue: How to allow many devices at home?

Home routers assign each device an address from Private Address Range (via DHCP).

Router rewrites Internet-bound packets to replace private source address with its public IP address. Recognizes response packets and rewrites in reverse.

(This complicates hosting servers at home.)

Security benefit: Devices at private IPs not directly addressable from the output world.

Coming Up

Reminders:

Web Project due Thursday at 6 PM

Midterm Exam, Friday, October 18, 7–8:30 PM

Networking 102

TCP, UDP, and DNS attacks

Network Defense

Denial of service attacks;
firewalls, IDSes, VPNs, zero trust