Last Lecture: Data Link Layer

1. *Design goals and issues*
2. *(More on) Error Control and Detection*
3. *Multiple Access Control (MAC)* ✔
4. *Ethernet, LAN Addresses and ARP*
5. *Hubs, Bridges, Switches*
6. *Wireless LANs*
7. WLAN Security
8. Mobile Networking

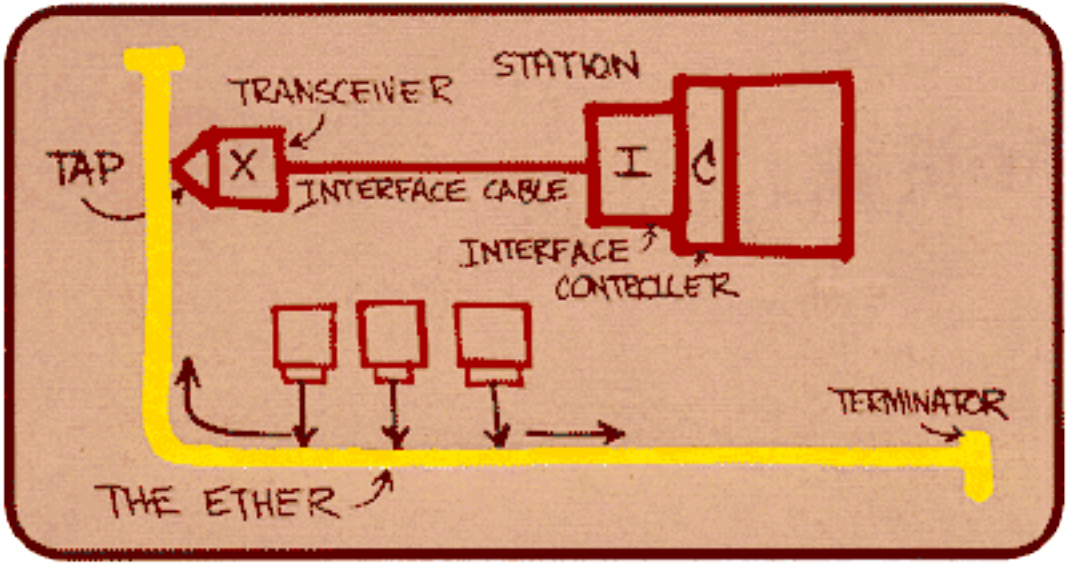
This Lecture: Data Link Layer

1. *Design goals and issues*
2. *(More on) Error Control and Detection*
3. *Multiple Access Control (MAC)*
4. *Ethernet, LAN Addresses and ARP* ✔
5. *Hubs, Bridges, Switches*
6. *Wireless LANs*
7. WLAN Security
8. Mobile Networking

Ethernet

"Dominant" LAN technology:

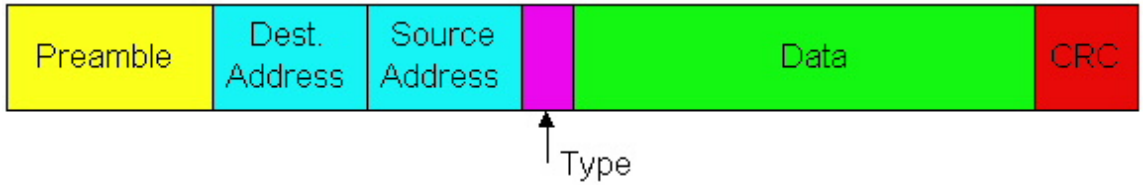
- Cheap $20 for 100Mbs!
- First widely used LAN technology
- Simpler, cheaper than token LANs and ATM
- Kept up with speed race: 10Mbps -- 10 Gbps



Metcalfe's Ethernet sketch

Ethernet Frame Structure

Sending adapter encapsulates IP datagram (or other network layer protocol packet) in *Ethernet frame*

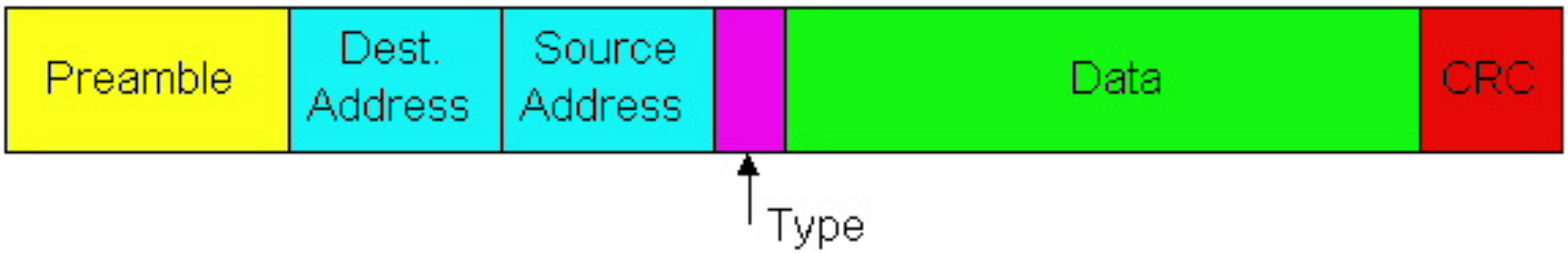


Preamble:

- 7 bytes with pattern 10101010 followed by one byte with pattern 10101011
- used to synchronize receiver, sender clock rates

Ethernet Frame Structure (more)

- *Addresses*: 6 bytes
  - if adapter receives frame with matching destination address, or with broadcast address (eg ARP packet), it passes data in frame to net-layer protocol
  - otherwise, adapter discards frame
- *Type*: indicates the higher layer protocol, mostly IP but others may be supported such as Novell IPX and AppleTalk)
- CRC: if error detected, frame is simply dropped



Ethernet: Unreliable & Connectionless

- *Connectionless:*
  - No handshaking between sending and receiving NICs
- *Unreliable:* receiving NIC doesn't send ACKs or NACKs to sending NIC
  - Stream of datagrams passed to network layer can have gaps (missing datagrams)
  - Gaps will be filled if app is using TCP
  - Otherwise, app will see gaps
- Ethernet's MAC protocol
  - Half-duplex: unslotted CSMA/CD
  - Full-duplex: no protocol, small gap between frames

Ethernet CSMA/CD algorithm

1. NIC receives datagram from network layer, creates frame
2. If NIC *senses channel idle* (for 96 bit times), starts frame transmission. If NIC senses channel busy, waits until channel idle (for 96 bit times), then transmits
3. If NIC transmits entire frame without detecting another transmission, NIC is done with frame!
4. If NIC *detects* another transmission while transmitting, *aborts* and sends (48-bit) *jam signal*
5. After aborting, NIC enters ***exponential backoff***: after *m*th collision, NIC chooses *K* at random from $\{0,1,2,\ldots,2^m-1\}$. NIC waits K·512 bit times, returns to Step 2

Ethernet's CSMA/CD (more)

- Jam Signal: make sure all other transmitters are aware of collision; 48 bits
- Bit time: .1 microsec for 10 Mbps Ethernet ;
  for K=1023, wait time is about 50 msec

See/interact with Java applet on AWL Web site: highly recommended !

- Exponential Backoff:
- *Goal*: adapt retransmission attempts to estimated current load
  - heavy load: random wait will be longer
- first collision: choose K from {0,1}; delay is K· 512 bit transmission times
- after second collision: choose K from {0,1,2,3}…
- after ten collisions, choose K from {0,1,2,3,4,…,1023}

CSMA/CD Efficiency

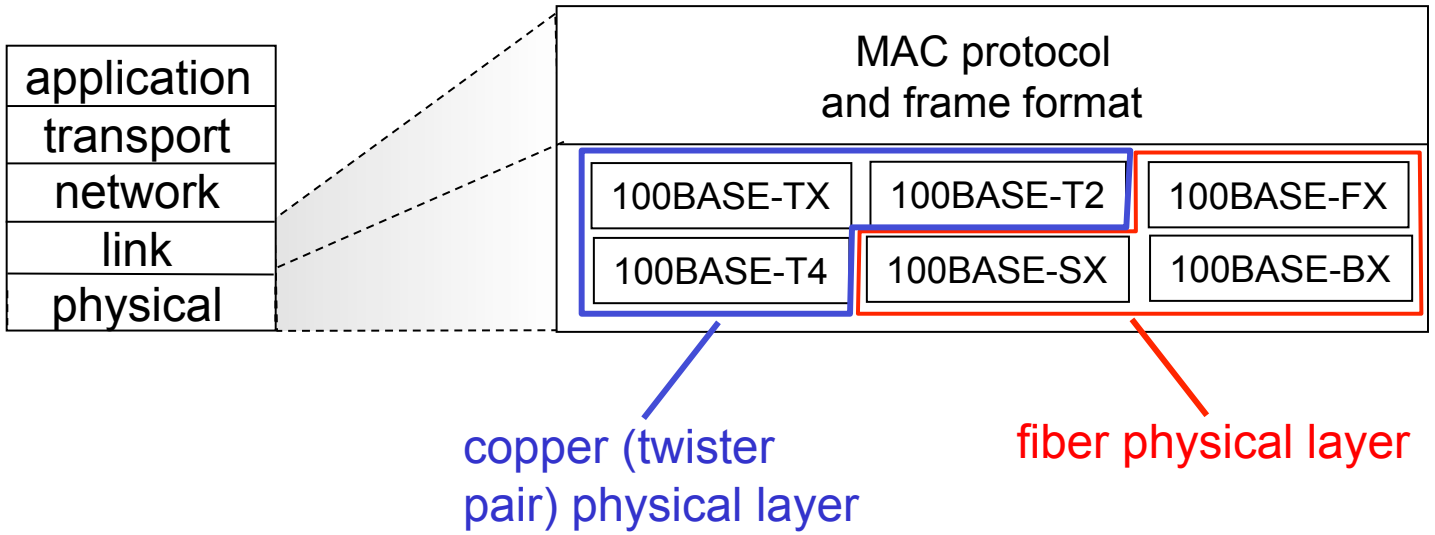
- T_{prop} = max prop between 2 nodes in LAN
- t_{trans} = time to transmit max-size frame

$$\text{efficiency} = \frac{1}{1 + 5t_{prop} / t_{trans}}$$

- Efficiency goes to 1 as t_{prop} goes to 0
- Goes to 1 as t_{trans} goes to infinity
- Much better than ALOHA, yet still decentralized, simple, and cheap

802.3 Ethernet Standards: Link & Physical

- *Many* different Ethernet standards
  - Common: MAC protocol and frame format
  - Different speeds: 10 Mbps, 100 Mbps, 1Gbps, 10G bps, 100Gbps
  - Different physical layer media: fiber, cable



LAN Addresses and ARP

32-bit IP addresses are

- *network-layer* addresses
- used to get a datagram to the destination IP subnet

LAN (or MAC or physical or Ethernet) addresses are

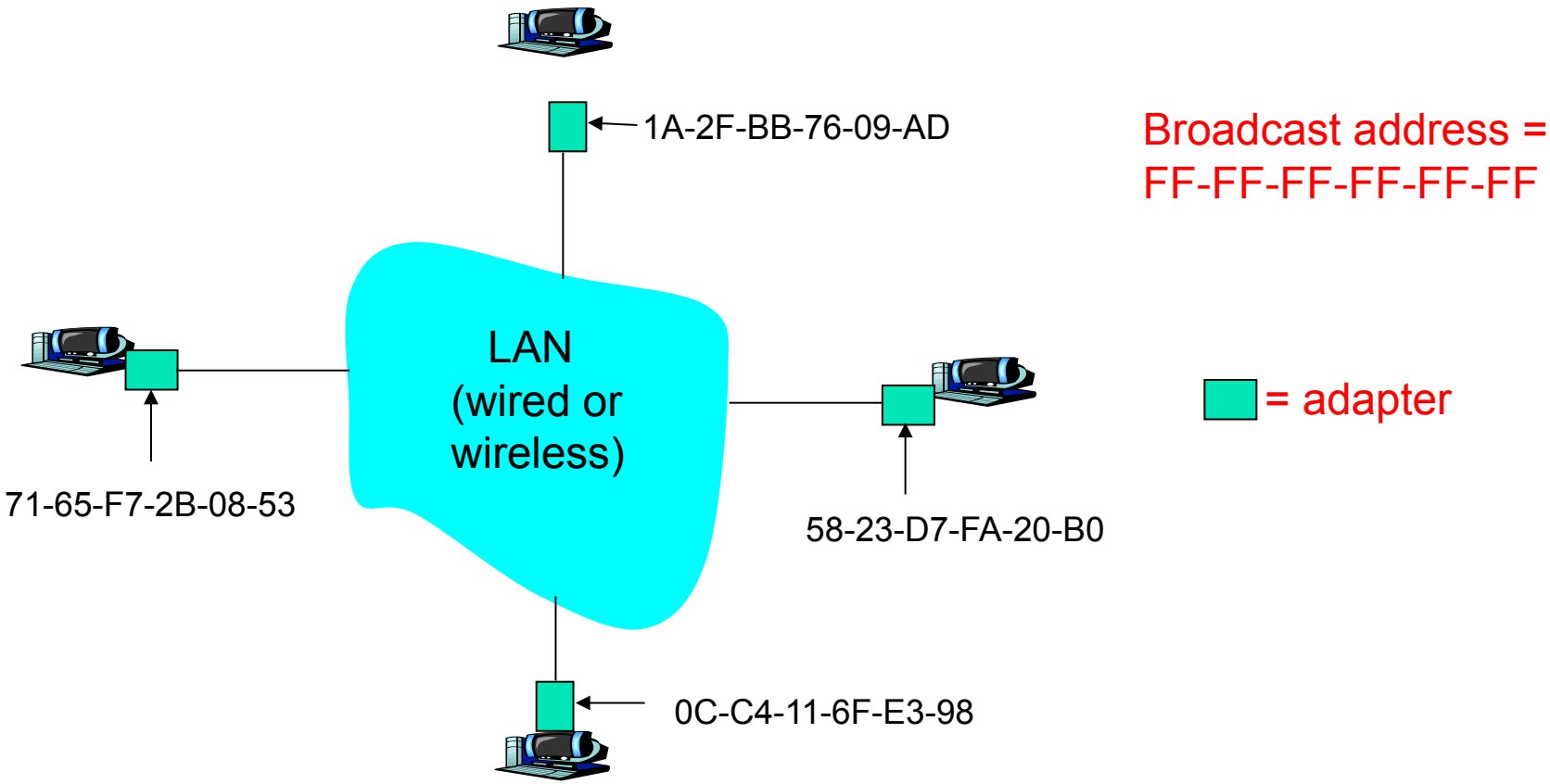
- used to get a frame from one interface to another physically-connected interface (same network)
- 48 bit MAC address (for most LANs)
  - burned in the adapter ROM
  - sometimes software settable

Tips and Tricks

- What are three quick ways to check the MAC address (or physical address) of your network interface card?

LAN Addresses and ARP

Each adapter on LAN has unique LAN address



LAN Address (more)

- MAC address allocation administered by IEEE
- Manufacturer buys portions of MAC address space
- *Analogy*:

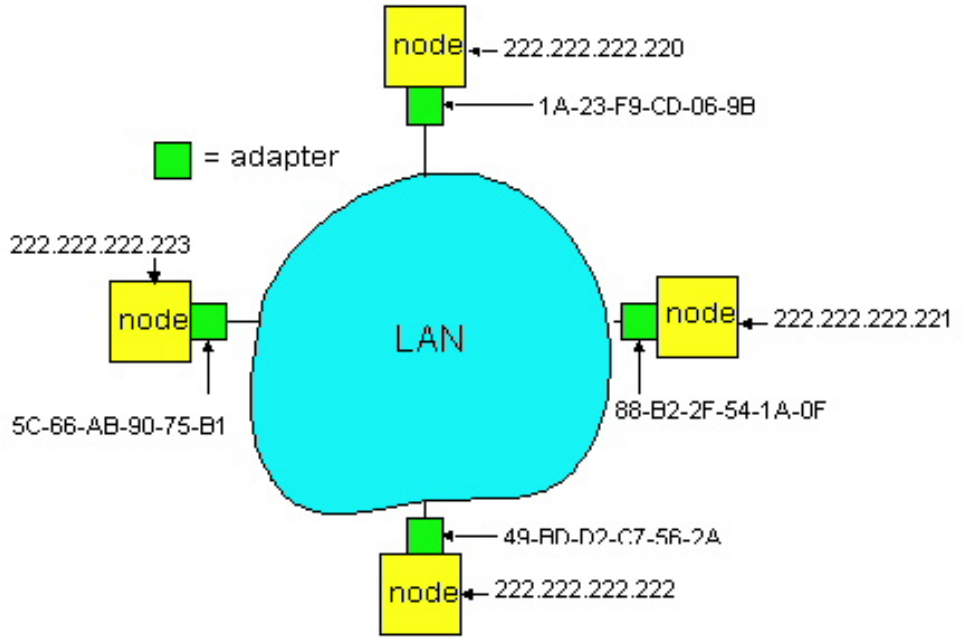
  (a) MAC address: like Social Security Number

  (b) IP address: like postal address

- MAC flat address => portability
  - can move LAN card from one LAN to another
- IP hierarchical address NOT portable
  - depends on IP network to which node is attached

ARP: Address Resolution Protocol

Question: how to determine MAC address of B knowing B's IP address?



- Each IP node (Host, Router) on LAN has an *ARP* table
- ARP Table: IP-MAC address mappings for some LAN nodes

  <IP address; MAC address; TTL>
  - TTL (Time To Live): time after which address mapping will be forgotten (typically 20 min)

ARP Protocol Basics

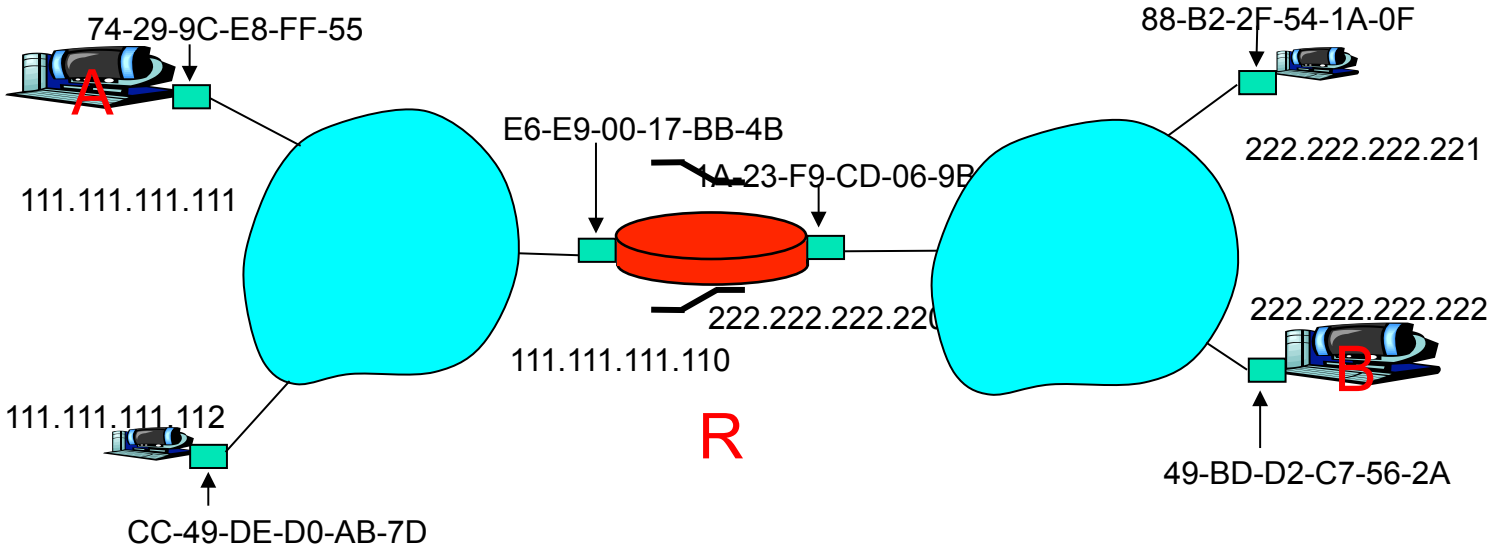
- A knows B's IP and wants to send datagram to B
  - Suppose B's MAC address is not in A's ARP table.
- A broadcasts ARP query packet containing B's IP
- B receives ARP query with its IP on
  - B sends (unicast) reply containing its MAC to A's MAC
- A caches (saves) IP-to-MAC address pair in its ARP
  - Until TTL expires (2 to 20 minutes)

- *ARP is "plug-and-play"*:
  - Nodes create their ARP tables without intervention from network administrator

Routing to Another LAN

Walkthrough: send datagram from A to B via R

assume A knows B's IP address

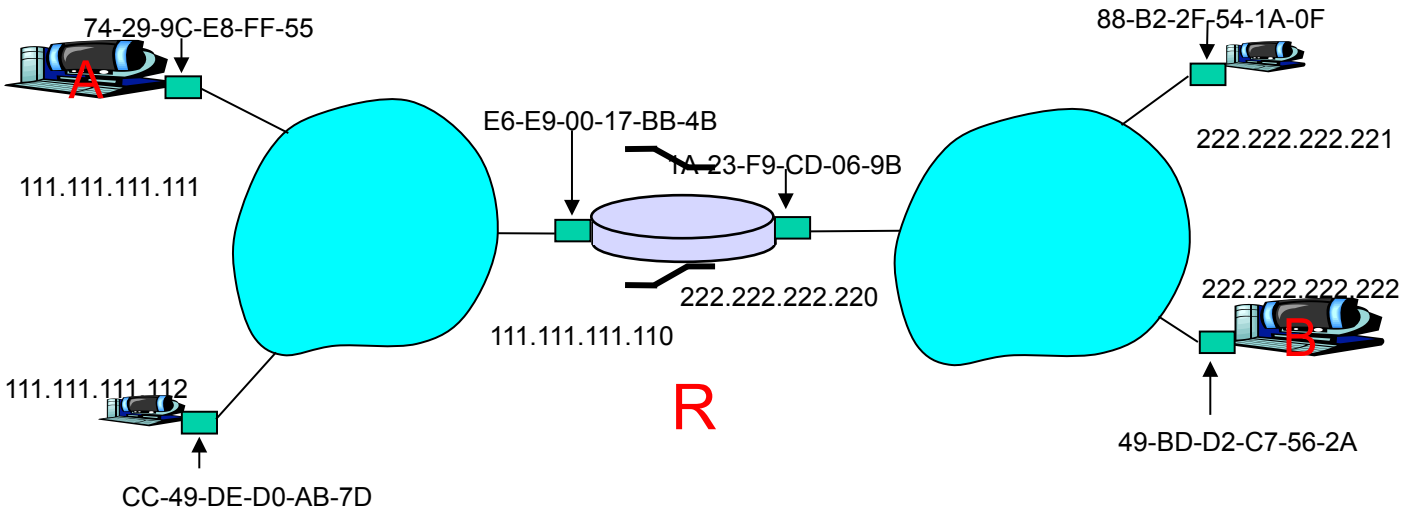


- two ARP tables in router R, one for each IP network (LAN)

- A creates IP datagram with source A, destination B

- A uses ARP to get R's MAC address for 111.111.111.110
- A creates link-layer frame with R's MAC address as dest, frame contains A-to-B IP datagram
- A's NIC sends frame
- R's NIC receives frame
- R removes IP datagram from Ethernet frame, sees its destined to B
- R uses ARP to get B's MAC address
- R creates frame containing A-to-B IP datagram sends to B

This is a really important example – make sure you understand!



Proxy ARP

- Also called *promiscuous ARP* or *ARP hack*

- A router connecting two networks acts as an ARP agent answering ARP requests from both networks
  - Hide two physical networks from each other
  - Both networks can use the same net. ID
  - Hide a group of hosts with older versions of TCP/IP on a separate physical cable, mostly to avoid protocol incompatibilities (subnetting not supported, broadcast address is all 0's instead of all 1's, etc.)

Pros and Cons of Proxy ARP

- Pros
  - proxy ARP can be added to a single router on a network and does not disturb the routing tables of the other routers on the network.
  - Proxy ARP must be used on the network where IP hosts are not configured with a default gateway or do not have any routing intelligence.
- Cons
  - increases the amount of ARP traffic on your segment
  - hosts need larger ARP tables
  - security can be undermined
  - does not work for networks that do not use ARP
  - does not generalize to all network topologies (2 routers connecting 2 networks)

Gratuitous ARP

- Host looks for MAC of its own IP
- Normally done at bootstrap time

- What for?
  - Report error (in syslog) if two hosts were configured with the same IP
  - Host just changed interface card: ARP modules automatically update their caches with the new source (IP,MAC) pair in the query

Reverse ARP (or RARP) – RFC903

- Normally used for diskless workstations
  - RARP requests are broadcast: "does any one know what the IP address of this MAC is?"
  - Replies are unicast: minimal information gain for a broadcast
- Need an RARP server
  - Complex design, system dependent
  - Need one server for each physical cable
  - Not all TCP/IP implementation support it (BOOTP – RFC 1542 - provides IP instead)

Tips and Tricks

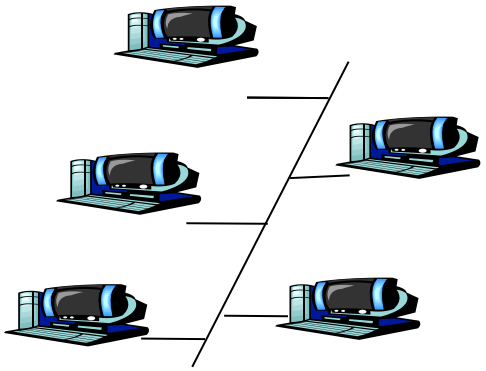
- *ARP poisoning: pretty dangerous*
  - Spoof an ARP reply → redirect traffic
  - Allows a "man-in-the-middle" sniffing on a switched LAN: forces traffic between two stations through the middle man
- Could be used for lots of different attacks
  - Session hijacking
  - UDP denial of service
  - TCP connection killing (instead of RST)
  - Switch cache overflow: block all traffic or turn it into a hub (more on this later)

Tips and Tricks

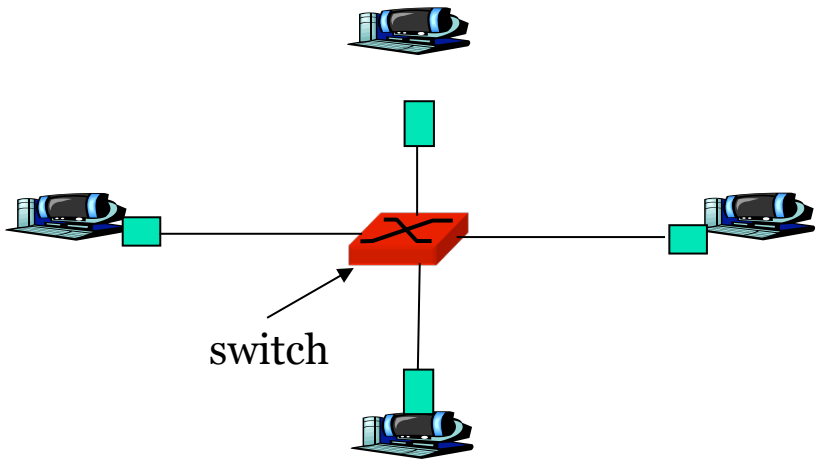
- Counter measures on ARP poisoning
  - Unix: ipfw, ipf (IP filter)
  - Windows: Network Ice/Black Ice ©
  - ArpWatch
  - Session encryption, authentication, etc.
  - Port security (available on some switches): shutdown violating port(s)
  - Hardcoding addresses
    - Management nightmare
    - Not scalable
    - Not supported by some OS vendors (all Windows versions before XP, I don't know about XP)
  - Use switched Ethernet
    - There's another kind of poisoning

From Bus to Star

- Bus topology popular through mid 90s
  - All nodes in the same *collision domain*
- Today: star topology prevails
  - Active *switch* in center
  - Each "spoke" runs a (separate) Ethernet protocol (nodes do not collide with each other)



bus: coaxial cable



star