# ECE 463
# Introduction to Computer Networks

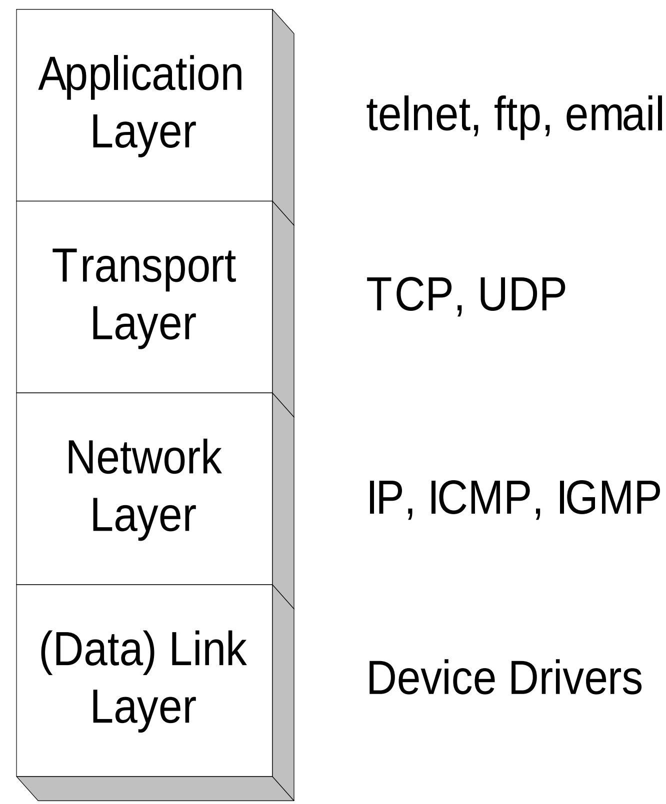
Internetworking:

Addressing and ARP

Sanjay Rao

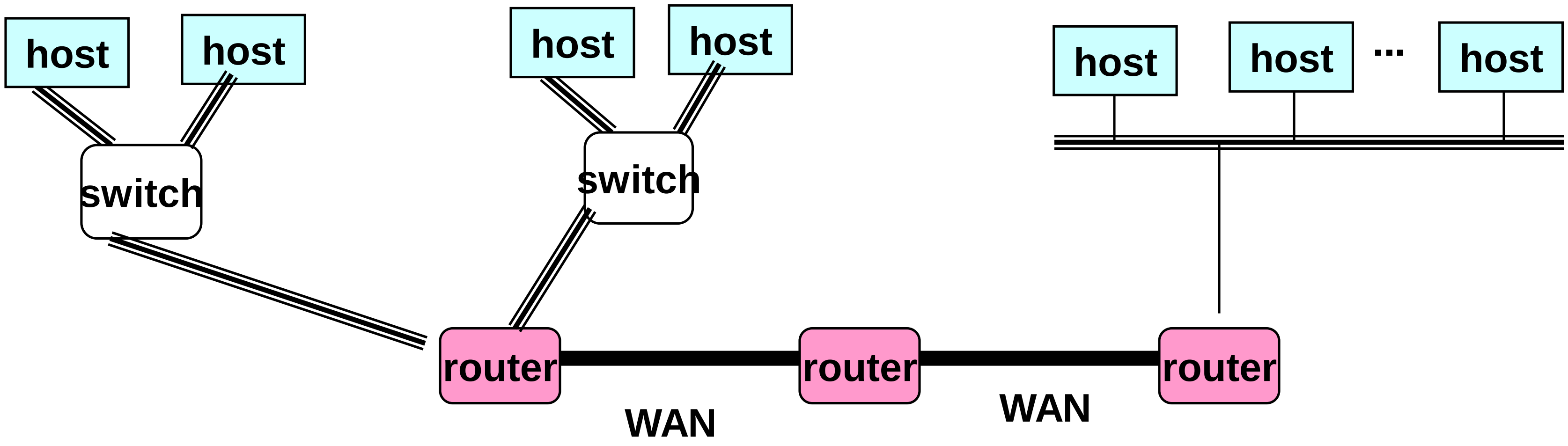
# Layered Protocol Architecture

- The TCP/IP protocol suite is the basis for the networks that we call the Internet.
- The TCP/IP suite has four layers:
- Computers (hosts) implement all four layers. Routers (gateways) only have the bottom two layers.



# What is an Internetwork?

- Multiple incompatible LANs can be physically connected by specialized computers called *routers.*
- The connected networks are called an *internetwork*.
  - The "*Internet*" is one (very big & successful) example of an internetwork

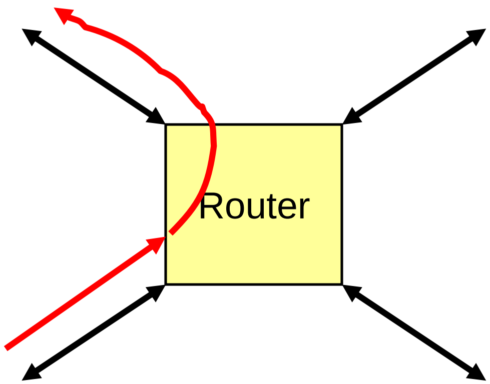



**Connected LANs might be completely different (e.g., Ethernet and ATM)**

# Issues in Designing an Internetwork

- How do I designate a distant host?
  - Addressing / naming
- How do I send information to a distant host?
  - Routing
- Challenge
  - Scalability
    - Ensure ability to grow to worldwide scale

# Router Operation




- Destination-Based Routing
  - Move packet through network via series of hops
- Forwarding:
  - Hardware table-lookup to determine next hop
  - Fast, must be done at line rate (i.e., per packet basis)
- Route table computation
  - How routers determine the routes in the first place
  - Software: more involved protocols

# Possible Addressing Schemes

- Flat
  - e.g., every host identified by its 48-bit MAC address
  - Router would need entry for every host in the world
    - Too big
    - Too hard to maintain as hosts come & go
- Hierarchy
  - Address broken into segments of increasing specificity
    - 765(Lafayette) 494(Purdue) 3399 (my office)
    - Indiana/ W. Lafayette / Purdue/ Sanjay
  - Route to general region and then work toward specific destination
  - As people and organizations shift, only update affected routing tables

# IP Addressing

- IPv4: 32-bit addresses
  - Typically, write in dotted decimal format
    - E.g., 128.2.198.135
    - Each number is decimal representation of byte

0 8 16 24 31

| | | | | |
|---|---|---|---|---|
| 128 | 2 | 198 | 135 | **Decimal** |
| 80 | 02 | c6 | 87 | **Hexadecimal** |
| 0100 0000 | 0000 0010 | 1100 0110 | 1000 0111 | **Binary** |

# IP Addressing and Forwarding

- Routing Table Requirement
  - Flat: For every destination IP address, give next hop
  - Nearly $2^{32}$ ($4.3 \times 10^9$) possibilities!
- Hierarchical Addressing Scheme

| | x | y |
|---|---|---|
| **pfx** | **network** | **host** |

  - Address split into network ID and host ID
    - Purdue has one network ID shared by all hosts within Purdue
  - All packets to given network follow same route
    - Until they reach destination network
  - Fields
    - `pfx` Prefix to specify split between network & host IDs
    - `network` $2^x$ possibilities
    - `host` $2^y$ possibilities

# Uniform vs. Non-uniform hierarchy

- Uniform Hierarchy
  - All hosts have same split of network/host
- Nonuniform Hierarchy
  - Network/host splits may vary

- Discussion: Why non-uniform?

# IP Address Classes: OLD SCHEME

- Class A

| | 7 | 24 |
|---|---|---|
| 0 | network | host |

18.7.22.69

**First digit: 1–126**

- Class B

| | 14 | 16 |
|---|---|---|
| 10 | network | host |

128.46.200.24

**First digit: 128–191**

- Class C

| | 21 | 8 |
|---|---|---|
| 110 | network | host |

205.201.9.200

**First digit: 192–223**

- Classes D, E, F
  - Not commonly used

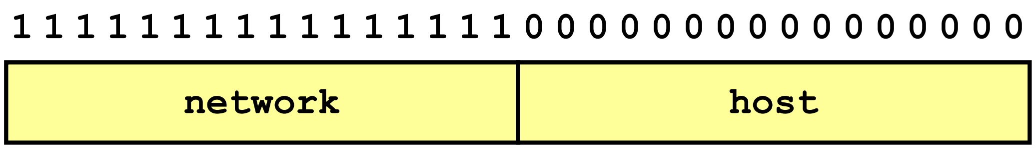
# IP Address Classes

| Class | Count | Hosts |
| --- | --- | --- |
| A | $2^7$-2 = 126 | $2^{24}$-2 = 16,777,214 |
| B | $2^{14}$ = 16,398 | $2^{16}$-2 = 65,534 |
| C | $2^{21}$ = 2,097,512 | $2^8$-2 = 254 |
| Total | 2,114,036 | |

- Partitioning too Coarse
  - Class A wasteful: Few organizations have 16.7 million hosts
  - Class C insufficient: Many organizations require multiple class C's
  - Too few Class B's.
- Too many different Network IDs
  - Routing tables must still have 2.1 million entries

# Classless Interdomain Routing: NEW SCHEME

- Arbitrary Split Between Network & Host IDs
  - Specify either by mask or prefix length




  - E.g.
    - 128.46.0.0 with netmask 255.255.0.0
    - 128.46.0.0/16

# Aggregation with CIDR

- Original Use: Aggregate Class C Addresses

– One organization assigned contiguous range of class C's

- e.g., Organization given all addresses 207.46.192.X -- 207.46.255.X
- Specify as CIDR address 207.46.192.0/18

| 0 | 8 | 16 | 24 | 31 |
|---|---|---|---|---|
| 207 | 46 | 192 | 0 | **Decimal** |
| cf | 2e | c0 | 00 | **Hexadecimal** |
| 1100 1111 | 0010 1110 | 11xx xxxx | xxxx xxxx | **Binary** |

**Upper 18 bits frozen** **Lower 14 bits arbitrary**

- Represents $2^6 = 64$ class C networks

– Use single entry in routing table

- Just as if were single network address

# Routing Table Entry Examples

| Address | Prefix Length | Third Byte | Byte Range |
|---|---|---|---|
| 207.46.0.0 | 19 | $000xxxxx_2$ | 0 – 31 |
| 207.46.32.0 | 19 | $001xxxxx_2$ | 32 – 63 |
| 207.46.64.0 | 19 | $010xxxxx_2$ | 64 – 95 |
| 207.46.128.0 | 18 | $10xxxxxx_2$ | 128 – 191 |
| 207.46.192.0 | 18 | $11xxxxxx_2$ | 192 – 255 |

# Important Concepts

- Hierarchical addressing critical for scalable system
  - Don’t require everyone to know everyone else
  - Reduces amount of updating when something changes
- Non-uniform hierarchy useful for heterogeneous networks
  - Class-based addressing too coarse
  - CIDR helps
- Implementation Challenge
  - Longest prefix matching much more difficult than when no ambiguity

# Longest Prefix Matching Example

Routing Table

| Network | Next Hop | 3rd Octet |
|---|---|---|
| 128.96.170.0/23 | Interface 0 | 1010 1010 |
| 128.96.168.0/23 | Interface 1 | 1010 1000 |
| 128.96.166.0/23 | R2 | 1010 0110 |
| 128.96.164.0/22 | R3 | 1010 0100 |
| (default) | R4 | |

Packet to destination 128.96.167.151: 1010 0111

- Matches two entries.
- Forwarded to R2 (Longer Prefix Match)

# ---ARP--

# LAN Addresses and ARP

32-bit IP address:

- *network-layer* address
- used to get datagram to destination network

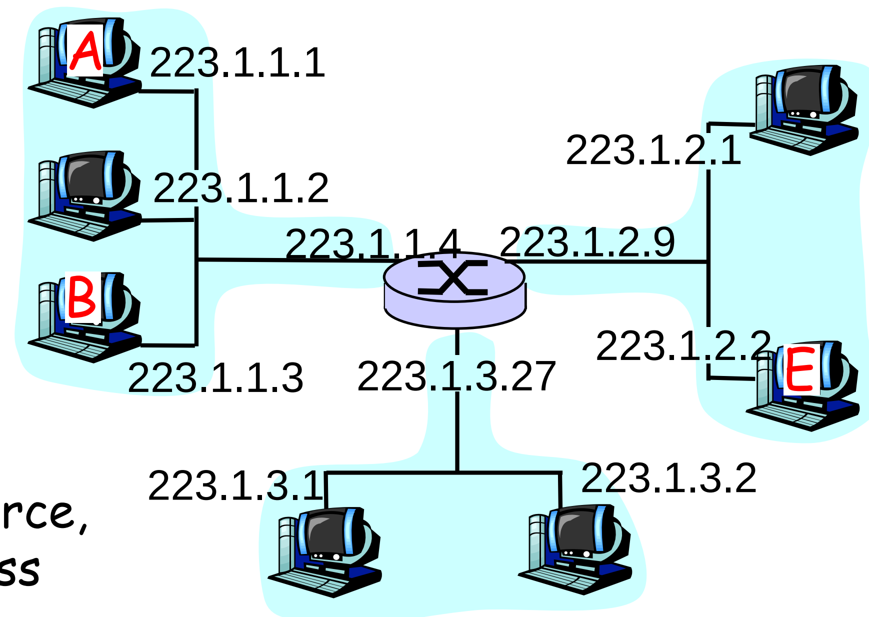
LAN (or MAC or physical) address:

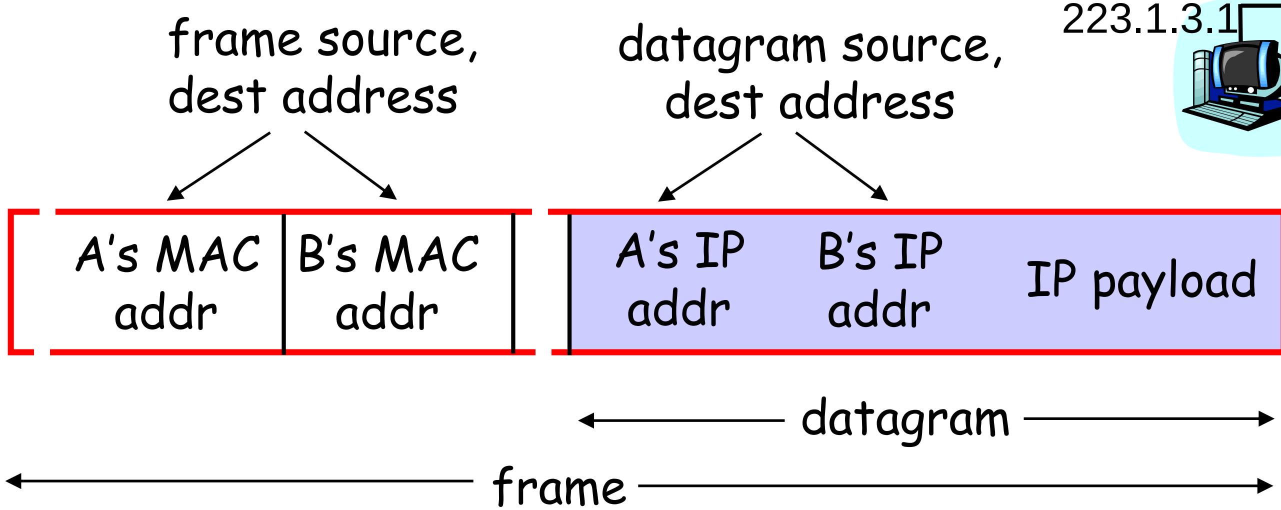
- used to get datagram from one interface to another physically-connected interface (same network)
- 48 bit MAC address

# LAN addresses (more)

Starting at A, given IP datagram addressed to B:

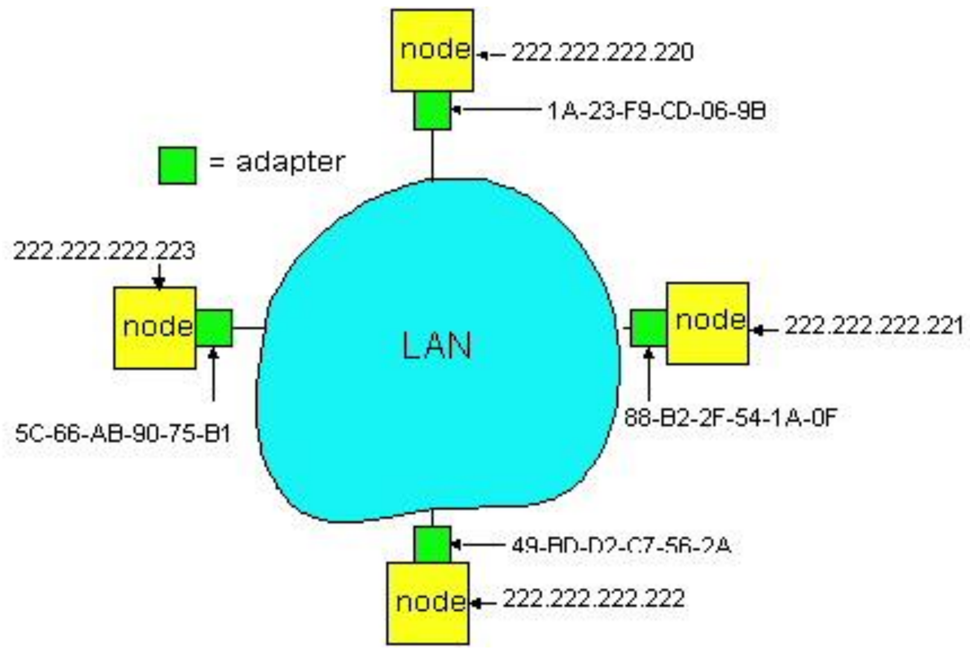
- look up IP address of B, find B on same network as A
- link layer sends datagram to B inside link-layer frame



# ARP: Address Resolution Protocol

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




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

  < IP address; MAC address; TTL>

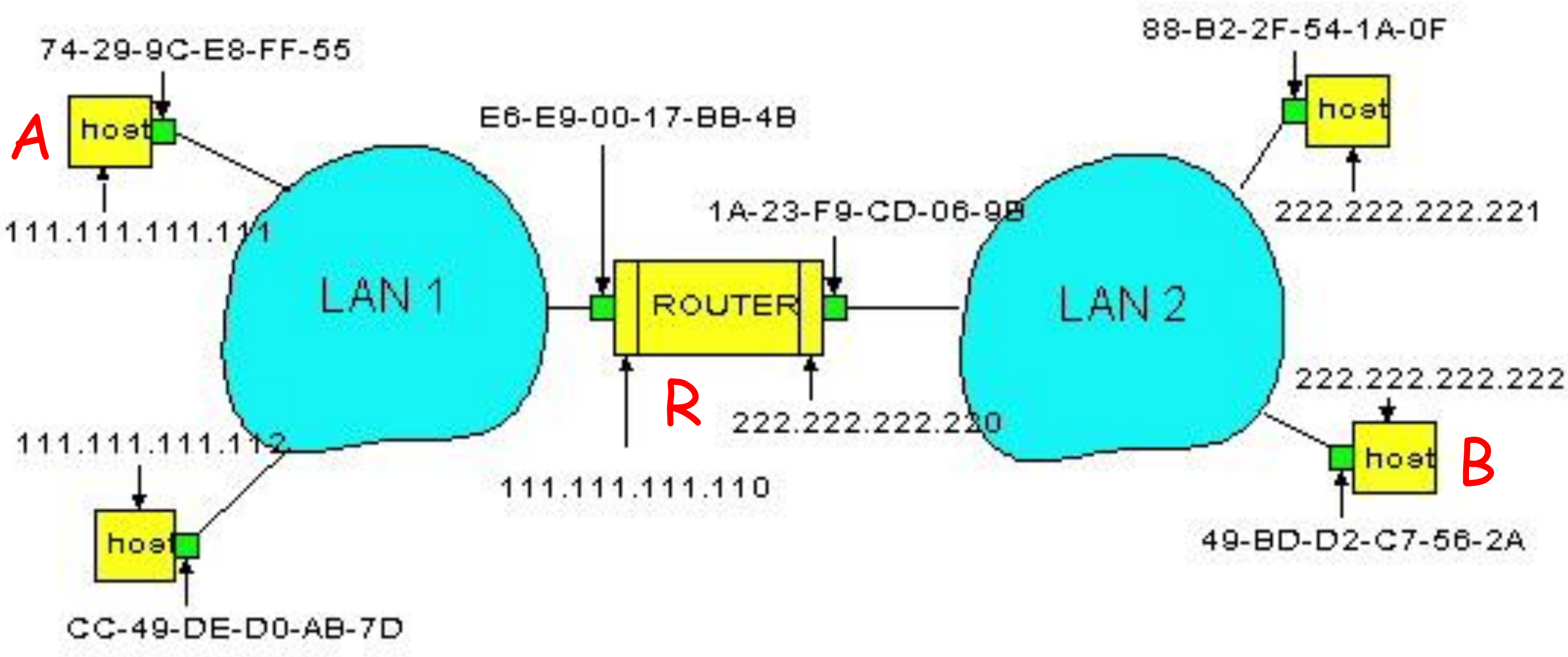
  < ………………………… >

  – TTL (Time To Live): time after which address mapping will be forgotten (typically 20 min)

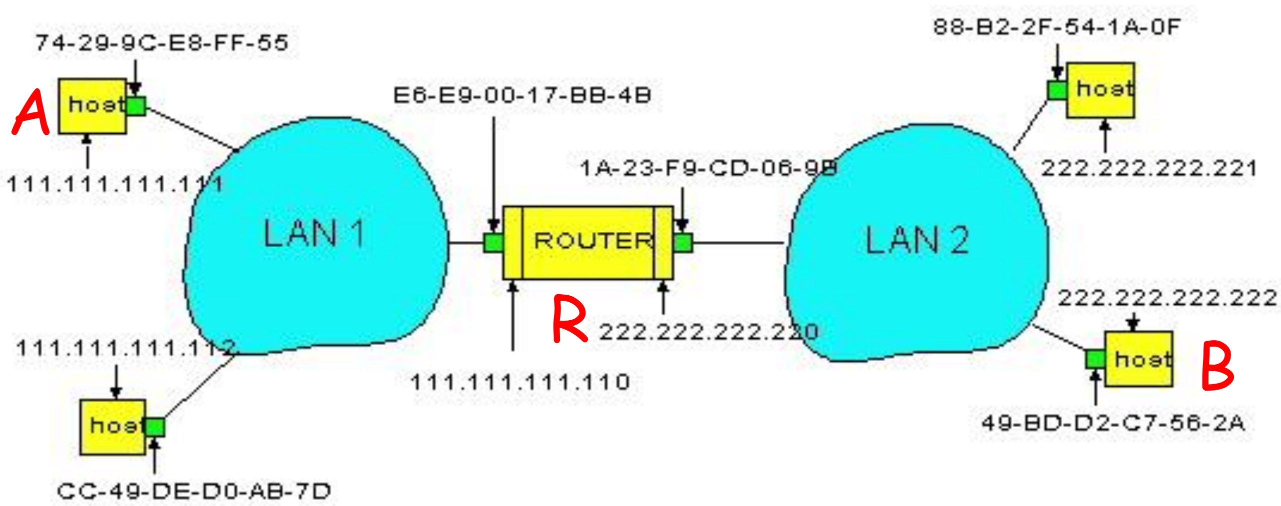
# ARP protocol

- A knows B's IP address, wants to learn physical address of B
- A broadcasts ARP query pkt, containing B's IP address
  - all machines on LAN receive ARP query
- B receives ARP packet, replies (unicast) to A with its (B's) physical layer address
- A caches (saves) IP-to-physical address pairs until information becomes old (times out)
  - soft state: information that times out (goes away) unless refreshed

# Transfer across Networks



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



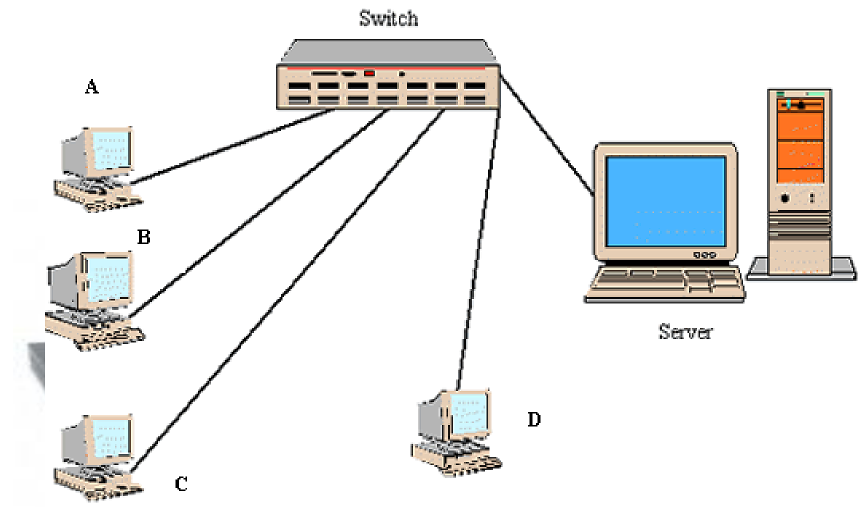
# ARP Demo

- /sbin/arp -n (shows arp table)

- netstat -rn  (shows IP routing table)

- sudo /usr/sbin/tcpdump -n arp host [hostIP]
- (tcpdump of ARP traffic)

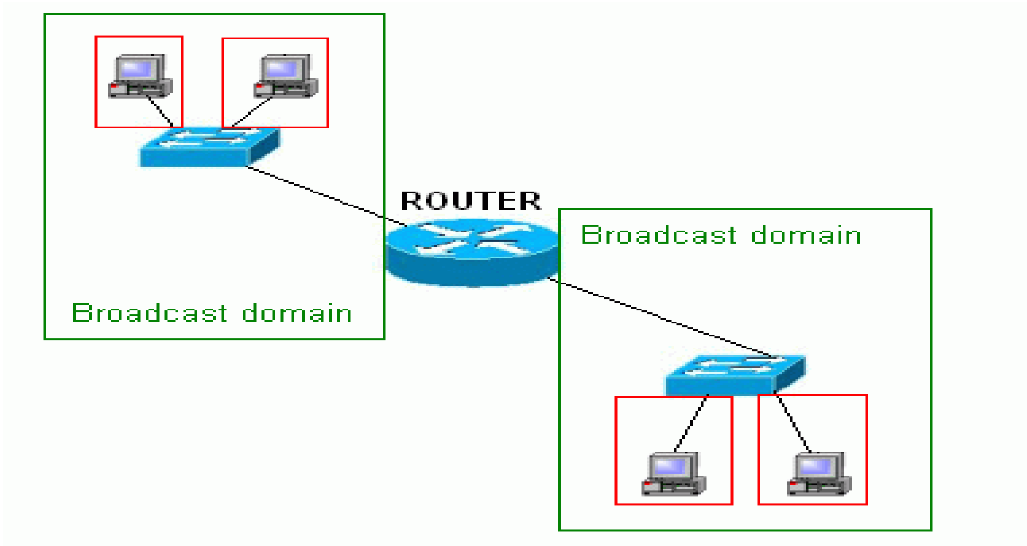
# What is a Switch?




- Switch: Networking device that operate on Ethernet frames – Layer 2 devices
- Forward pkts to the destination's MAC address.
- Plug-and-Play devices - Self-configuring without hardware or software changes.
- Will propagate  broadcast packets (ARP)
- Will not show up in output of "traceroute"

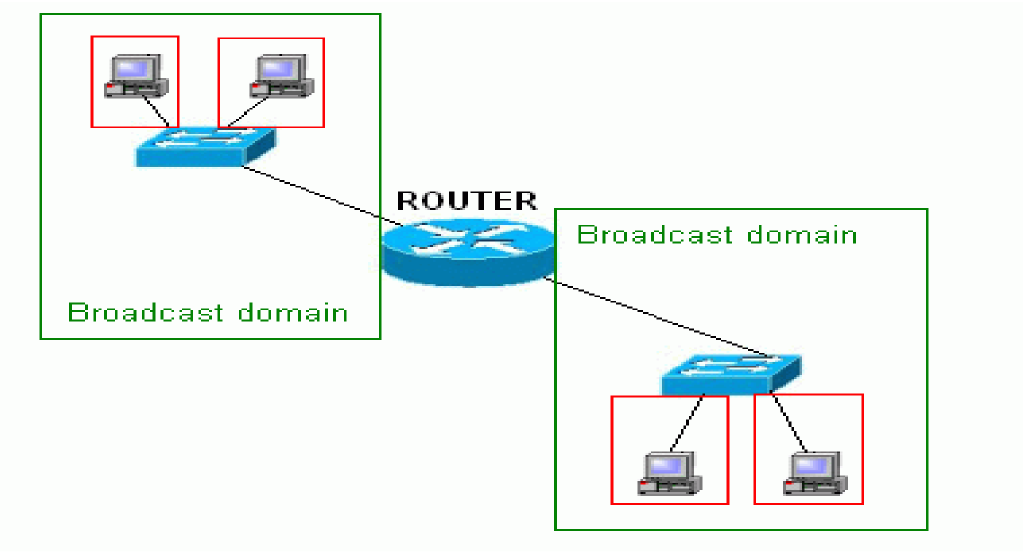
- Example:
  - Host A can have arbitrary MAC address. No "rule" dictating what it should be.
  - Switch will learn address of A over time.
  - A need not be configured with information about the switch.

# What is a Router?




- Router: networking device - forwards data packets across an inter-network.
- Routers operate on Layer 3 of OSI model and use IP address information of the destination to forward the packet.
- Will not propagate broadcast packets (ARP)
- Are not Plug-and-play devices, hosts connected to the routers need their IP addresses to be configured.
  - Router must be configured to indicate packets of certain subnet must be directed on particular interface.
  - IP address of host must be carefully configured to match subnet it is on.
  - Host also configured with router information (typically)

# Broadcast Domain




- Broadcast Domain: 2 hosts in same broadcast domain if a broadcast packet (e.g. ARP packet) sent by one of the hosts will also reach the other host.
- Different than “Collision Domain”.
- Collision Domain: logical network segment where data packets can "collide" with one another for being sent on a shared medium at the same time.

# Physical View of a LAN