Week 9 Lecture 2

Link Layer, LANs

Recall from last lecture we talked about multiple access protocols. We looked at three different protocols: channel partitioning, random access, and taking turns. However, ethernet network setup is not setup using CSMA/CD, we use switches instead. But we do use CSMA/CA (collision avoidance) in 802.11 Wi-Fi.

# Switched LANs

MAC addresses and ARP

* 32-bit IP address: network layer addresses for interface. It is used for layer 3 (network layer) forwarding.
* 48-bit MAC address (for most LANs) burned in NICROM, also sometimes software settable. For example: 71-65-F7-2B-08-53. Each char represents a hexadecimal number, which is 4 bytes. 4\*12 = 48

*Note: MAC address doesn’t have strict meaning and it is completely portable.*

The hardware company gets MAC address from IEEE. This lecture we are looking at how ARP protocol works. **Question: how to determine interface’s MAC address knowing its IP address?**

Table

Description automatically generated

ARP table: Each IP node (host, router) on LAN has table of

* IP/MAC address mappings for some LAN nodes: < IP address; MAC address; TTL>
* TTL: time after which address mapping will be forgotten (typically 20 min)

**Addressing: routing in same LAN**

A want to send datagram to B, if B is on A’s ARP table, then A can know its MAC address from IP address. If B is not on A’s ARP table, then A will broadcast ARP query packet, containing B’s IP address, v B receives ARP packet, replies to A with its (B's) MAC address. Then A will have B’s MAC address, and it stores B in its ARP table.

**Addressing: routing to another LAN**

send datagram from A to B via R: focus on addressing – at IP (datagram) and MAC layer (frame)

1. assume A knows B’s IP address (how?)

* A knows B is not local through subnet mask, discover B via DHCP. A takes B’s IP address and apply subnet mask, the resultant is the network part. If the network part is different from its own network part, then it knows it is not in the same network.

1. assume A knows IP address of first hop router, R (how?)

* default router (discovered via DHCP)

1. assume A knows R’s MAC address (how?)

* A sends out the ARP query for the router, only the router will respond back, and A knows the MAC address of the particular router.

1. A now knows the Destination IP address, it creates link layer frame with R’s MAC address as destination, frame contains A-to-B IP datagram.
2. Frame is sent from A to R, frame received by R, datagram removed, passed up to IP layer
3. R forwards datagram with IP source A, destination B (forwarding table)
4. R creates link-layer frame with B's MAC address as destination, frame contains A-to-B IP datagram

The Source & destination MAC addresses will typically change from hop to hop, but the source & destination IP addresses will typically remain the same

## Ethernet

**Bus vs Start**

* bus: popular through mid-90s, all nodes in same collision domain (can collide with each other), CSMA/CD for media access control
* star: prevails today, active switch in centre, each “spoke” runs a (separate) Ethernet protocol (nodes do not collide with each other), No sharing, no CSMA/CD

Ethernet: unreliable, connectionless

* connectionless: no handshaking between sending and receiving NICs
* unreliable: receiving NIC does not send acks or nacks to send NIC
  + data in dropped frames recovered only if initial sender uses higher layer rdt (e.g., TCP), otherwise dropped data lost

## Switches

* **Definition:** A switch is a link layer device that stores and forwards ethernet frames. It examines incoming a frame’s MAC address, and selectively forwards it to one or more outgoing links.
* Switches are *transparent* in the sense that hosts are unaware of their presence. Switches also don’t need to be configured.
* Hosts have a dedicated, direct connection to a switch. Switches buffer packets and runs at full duplex.
* Switches are *self-learning* in the sense that they learn which hosts can be reached through which interfaces.

Diagram

Description automatically generatedSwitch Forwarding Table: how does switch know A’ reachable via interface 4, B’ reachable via interface 5?

We start from an empty table, if A wants to know A’s MAC address, if will first check the switch, then flood: forward on all interfaces except arriving interface. When a frame is received, the switch learns the location of the sender and records the sender/location pair in its switch table.

Diagram

Description automatically generated

When we have multiple switches, it works exactly the same as in single-switch case! S2 and S3 don’t know anything so they just flood all interfaces. Then S3 will get response from G, So S4 knows S3 has G, then S1 knows from S4 about where G is, then A is able to send packets to G.

## Switches vs Routers

* both are store-and-forward
* both have forwarding tables

**Differences**:

* Different layers
* routers: compute tables using routing algorithms, IP addresses
* switches: learn forwarding table using flooding, learning, MAC addresses

# Wireless Network

**Wireless Links**

**Wi-Fi**

There are multiple types of Wi-Fi protocols including 802.11b, 802.11a, 802.11g and 802.11n. They all have minor differences including their spectrum/range and bit rate.

important differences from wired link

* Decreased signal strength
* Interference from other sources
* Diagram

  Description automatically generatedMultipath propagation (i.e. signals bouncing off objects)

The *free space path loss* is calculated as

* *d* is the distance
* *λ* is the wavelength
* *f* is the frequency
* *c* is the speed of light

Chart, line chart

Description automatically generated**SNR: signal-to-noise ratio**

larger SNR – easier to extract signal from noise (a “good thing”). SNR versus BER tradeoffs

* given physical layer: increase power -> increase SNR->decrease BER
* given SNR: choose physical layer that meets BER requirement, giving highest throughput

SNR may change with mobility: dynamically adapt physical layer (modulation technique, rate)