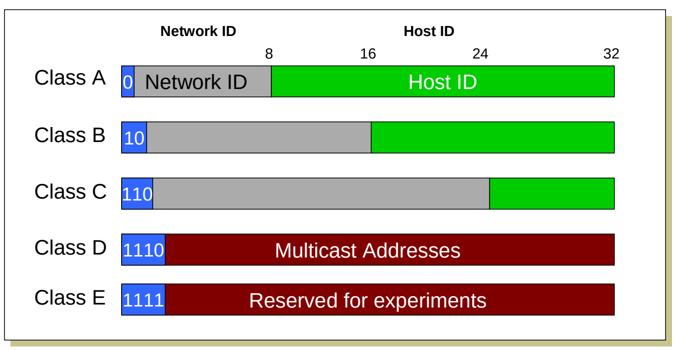
15-441 Computer Networking

Lecture 9 – IP Packets

Overview

- Last lecture
  - How does choice of address impact network architecture and scalability?
  - What do IP addresses look like?
- This lecture
  - Modern IP addresses
  - How to get an IP address?
  - What do IP packets look like?
  - How do routers work?

IP Address Classes
(Some are Obsolete)



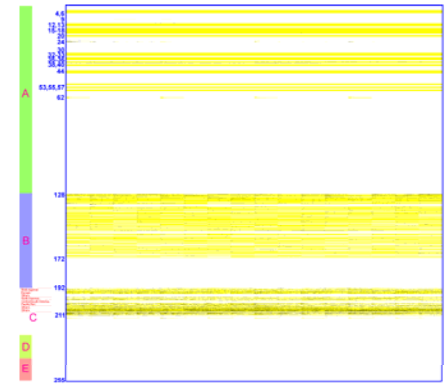
Outline

- CIDR IP addressing
- Forwarding examples
- IP Packet Format

IP Address Problem (1991)

- Address space depletion
  - In danger of running out of classes A and B
  - Why?
    - Class C too small for most domains
    - Very few class A – very careful about giving them out
    - Class B – greatest problem
- Class B sparsely populated
  - But people refuse to give it back
- Large forwarding tables
  - 2 Million possible class C groups

IP Address Utilization ('97)



http://www.caida.org/outreach/resources/learn/ipv4space/

Classless Inter-Domain Routing (CIDR) – RFC1338

- Allows arbitrary split between network & host part of address
  - Do not use classes to determine network ID
  - Use common part of address as network number
  - E.g., addresses 192.4.16 - 192.4.31 have the first 20 bits in common. Thus, we use these 20 bits as the network number → 192.4.16/20
- Enables more efficient usage of address space (and router tables) → How?
  - Use single entry for range in forwarding tables
  - Combined forwarding entries when possible

CIDR Example

- Network is allocated 8 class C chunks, 200.10.0.0 to 200.10.7.255
  - Allocation uses 3 bits of class C space
  - Remaining 20 bits are network number, written as 201.10.0.0/21
- Replaces 8 class C routing entries with 1 combined entry
  - Routing protocols carry prefix with destination network address
  - Longest prefix match for forwarding

IP Addresses: How to Get One?

Network (network portion):

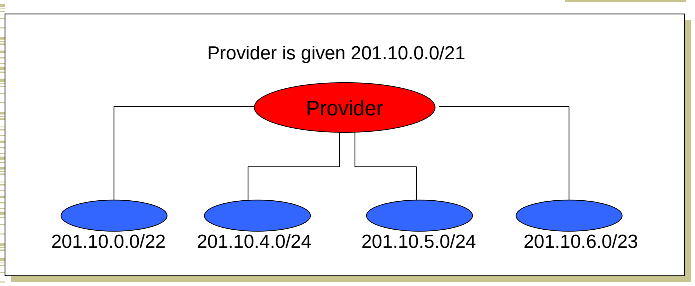
- Get allocated portion of ISP's address space:

ISP's block	11001000 00010111 00010000	00000000	200.23.16.0/20
Organization 0	11001000 00010111 00010000	00000000	200.23.16.0/23
Organization 1	11001000 00010111 00010010	00000000	200.23.18.0/23
Organization 2	11001000 00010111 00010100	00000000	200.23.20.0/23
...	.....	....	....
Organization 7	11001000 00010111 00011110	00000000	200.23.30.0/23

IP Addresses: How to Get One?

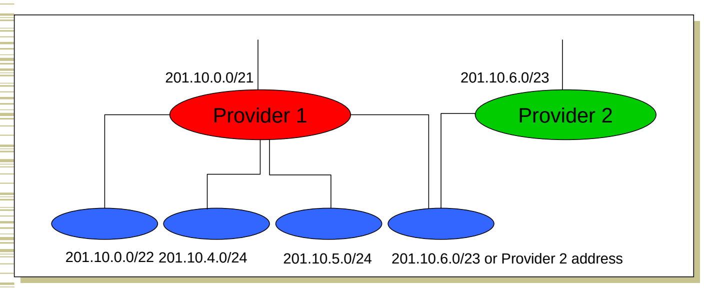
- How does an ISP get block of addresses?
  - From **Regional Internet Registries** (RIRs)
    - ARIN (North America, Southern Africa), APNIC (Asia-Pacific), RIPE (Europe, Northern Africa), LACNIC (South America)
- How about a single host?
  - Hard-coded by system admin in a file
  - DHCP: Dynamic Host Configuration Protocol: dynamically get address: "plug-and-play"
    - Host broadcasts "DHCP discover" msg
    - DHCP server responds with "DHCP offer" msg
    - Host requests IP address: "DHCP request" msg
    - DHCP server sends address: "DHCP ack" msg

CIDR Illustration



CIDR Implications

- Longest prefix match!!



Outline

- CIDR IP addressing
- Forwarding examples
- IP Packet Format

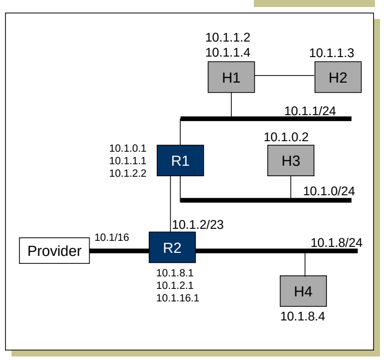
Host Routing Table Example

```
Destination      Gateway        Genmask         Iface
128.2.209.100    0.0.0.0        255.255.255.255 eth0
128.2.0.0        0.0.0.0        255.255.0.0     eth0
127.0.0.0        0.0.0.0        255.0.0.0       lo
0.0.0.0          128.2.254.36   0.0.0.0         eth0
```

- From "netstat –rn"
- Host 128.2.209.100 when plugged into CS ethernet
- Dest 128.2.209.100 → routing to same machine
- Dest 128.2.0.0 → other hosts on same ethernet
- Dest 127.0.0.0 → special loopback address
- Dest 0.0.0.0 → default route to rest of Internet
  - Main CS router: gigrouter.net.cs.cmu.edu (128.2.254.36)

Routing to the Network

- Packet to 10.1.1.3 arrives
- Path is R2 – R1 – H1 – H2

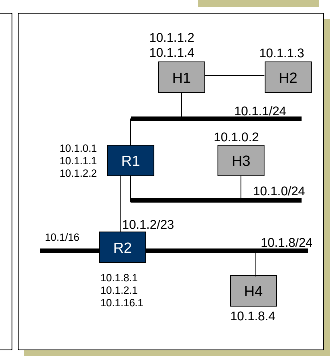


Routing Within the Subnet

- Packet to 10.1.1.3
- Matches 10.1.0.0/23

Routing table at R2

Destination	Next Hop	Interface
127.0.0.1	127.0.0.1	lo0
Default or 0/0	provider	10.1.16.1
10.1.8.0/24	10.1.8.1	10.1.8.1
10.1.2.0/23	10.1.2.1	10.1.2.1
10.1.0.0/23	10.1.2.2	10.1.2.1

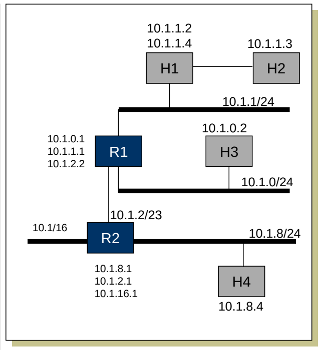


Routing Within the Subnet

- Packet to 10.1.1.3
- Matches 10.1.1.1/31
  - Longest prefix match

Routing table at R1

Destination	Next Hop	Interface
127.0.0.1	127.0.0.1	lo0
Default or 0/0	10.1.2.1	10.1.2.2
10.1.0.0/24	10.1.0.1	10.1.0.1
10.1.1.0/24	10.1.1.1	10.1.1.4
10.1.2.0/23	10.1.2.2	10.1.2.2
10.1.1.2/31	10.1.1.2	10.1.1.2



Aside: Interaction with Link Layer

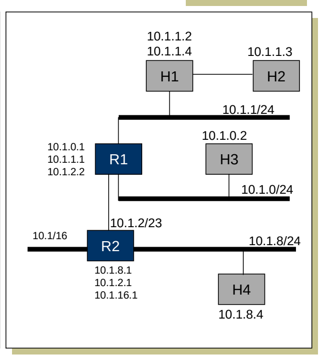
- How does one find the Ethernet address of a IP host?
- ARP
  - Broadcast search for IP address
    - E.g., “who-has 128.2.184.45 tell 128.2.206.138” sent to Ethernet broadcast (all FF address)
  - Destination responds (only to requester using unicast) with appropriate 48-bit Ethernet address
    - E.g, “reply 128.2.184.45 is-at 0:d0:bc:f2:18:58” sent to 0:c0:4f:d:ed:c6

Routing Within the Subnet

- Packet to 10.1.1.3
- Direct route
  - Longest prefix match

Routing table at H1

Destination	Next Hop	Interface
127.0.0.1	127.0.0.1	lo0
Default or 0/0	10.1.1.1	10.1.1.2
10.1.1.0/24	10.1.1.2	10.1.1.1
10.1.1.3/31	10.1.1.2	10.1.1.2



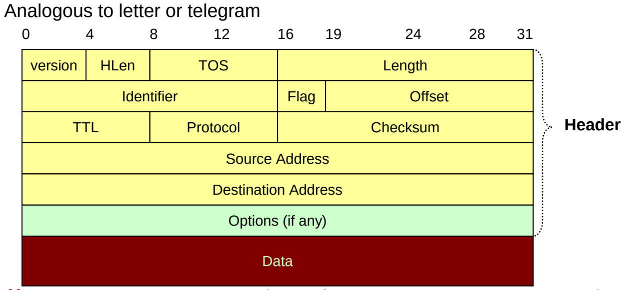
Outline

- CIDR IP addressing
- Forwarding examples
- IP Packet Format

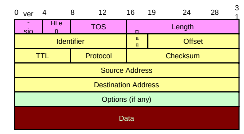
IP Service Model

- Low-level communication model provided by Internet
- Datagram
  - Each packet self-contained
    - All information needed to get to destination
    - No advance setup or connection maintenance
  - Analogous to letter or telegram

IPv4 Packet Format



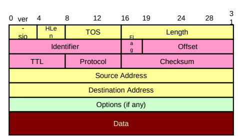
IPv4 Header Fields



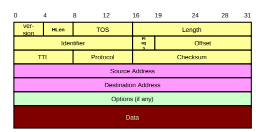
- Version: IP Version
  - 4 for IPv4
- HLen: Header Length
  - 32-bit words (typically 5)
- TOS: Type of Service
  - Priority information
- Length: Packet Length
  - Bytes (including header)
- Header format can change with versions
  - First byte identifies version
- Length field limits packets to 65,535 bytes
  - In practice, break into much smaller packets for network performance considerations

IPv4 Header Fields

- Identifier, flags, fragment offset → used primarily for fragmentation
- Time to live
  - Must be decremented at each router
  - Packets with TTL=0 are thrown away
  - Ensure packets exit the network
- Protocol
  - Demultiplexing to higher layer protocols
  - TCP = 6, ICMP = 1, UDP = 17…
- Header checksum
  - Ensures some degree of header integrity
  - Relatively weak – 16 bit
- Options
  - E.g. Source routing, record route, etc.
  - Performance issues
    - Poorly supported



IPv4 Header Fields

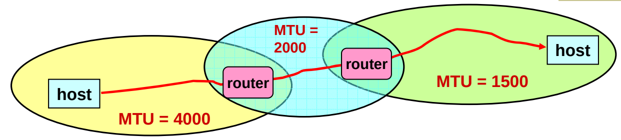


- Source Address
  - 32-bit IP address of sender
- Destination Address
  - 32-bit IP address of destination
- Like the addresses on an envelope
- Globally unique identification of sender & receiver

IP Delivery Model

- *Best effort service*
  - Network will do its best to get packet to destination
- Does NOT guarantee:
  - Any maximum latency or even ultimate success
  - Sender will be informed if packet doesn't make it
  - Packets will arrive in same order sent
  - Just one copy of packet will arrive
- Implications
  - Scales very well
  - Higher level protocols must make up for shortcomings
    - Reliably delivering ordered sequence of bytes → TCP
  - Some services not feasible
    - Latency or bandwidth guarantees

IP Fragmentation



- Every network has own Maximum Transmission Unit (MTU)
  - Largest IP datagram it can carry within its own packet frame
    - E.g., Ethernet is 1500 bytes
  - Don't know MTUs of all intermediate networks in advance
- IP Solution
  - When hit network with small MTU, fragment packets

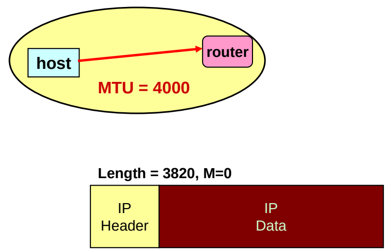
Reassembly

- Where to do reassembly?
  - End nodes or at routers?
- End nodes
  - Avoids unnecessary work where large packets are fragmented multiple times
  - If any fragment missing, delete entire packet
- Dangerous to do at intermediate nodes
  - How much buffer space required at routers?
  - What if routes in network change?
    - Multiple paths through network
    - All fragments only required to go through destination

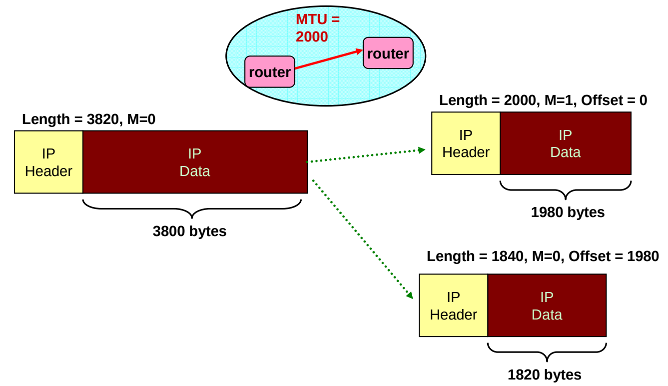
Fragmentation Related Fields

- Length
  - Length of IP fragment
- Identification
  - To match up with other fragments
- Flags
  - Don't fragment flag
  - More fragments flag
- Fragment offset
  - Where this fragment lies in entire IP datagram
  - Measured in 8 octet units (13 bit field)

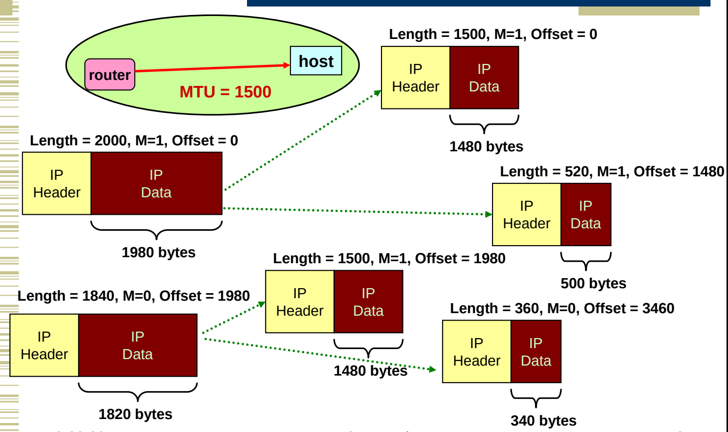
IP Fragmentation Example #1



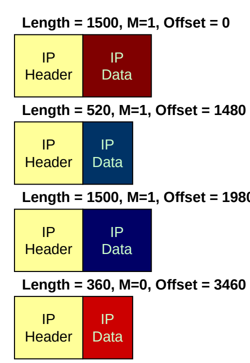
IP Fragmentation Example #2



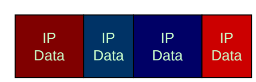
IP Fragmentation Example #3



IP Reassembly



- Fragments might arrive out-of-order
  - Don't know how much memory required until receive final fragment
- Some fragments may be duplicated
  - Keep only one copy
- Some fragments may never arrive
  - After a while, give up entire process



Fragmentation and Reassembly Concepts

- Demonstrates many Internet concepts
- Decentralized
  - Every network can choose MTU
- Connectionless
  - Each (fragment of) packet contains full routing information
  - Fragments can proceed independently and along different routes
- Best effort
  - Fail by dropping packet
  - Destination can give up on reassembly
  - No need to signal sender that failure occurred
- Complex endpoints and simple routers
  - Reassembly at endpoints

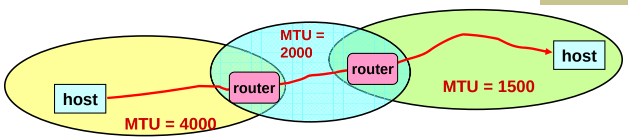
Fragmentation is Harmful

- Uses resources poorly
  - Forwarding costs per packet
  - Best if we can send large chunks of data
  - Worst case: packet just bigger than MTU
- Poor end-to-end performance
  - Loss of a fragment
- Path MTU discovery protocol → determines minimum MTU along route
  - Uses ICMP error messages
- Common theme in system design
  - Assure correctness by implementing complete protocol
  - Optimize common cases to avoid full complexity

Internet Control Message Protocol (ICMP)

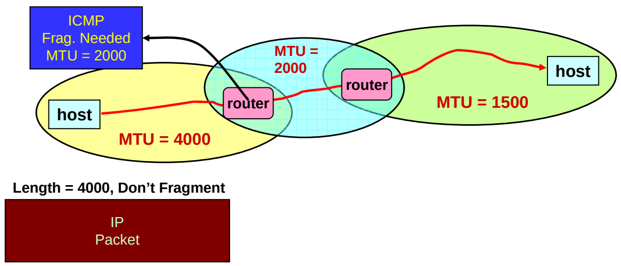
- Short messages used to send error & other control information
- Examples
  - Ping request / response
    - Can use to check whether remote host reachable
  - Destination unreachable
    - Indicates how packet got & why couldn't go further
  - Flow control
    - Slow down packet delivery rate
  - Redirect
    - Suggest alternate routing path for future messages
  - Router solicitation / advertisement
    - Helps newly connected host discover local router
  - Timeout
    - Packet exceeded maximum hop limit

IP MTU Discovery with ICMP

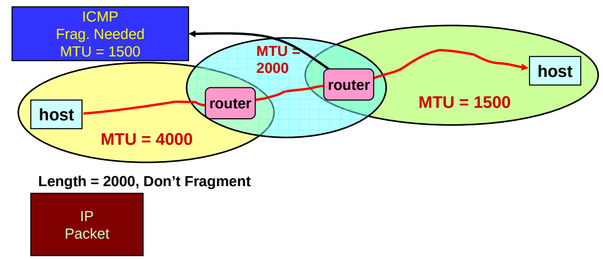


- Typically send series of packets from one host to another
- Typically, all will follow same route
  - Routes remain stable for minutes at a time
- Makes sense to determine path MTU before sending real packets
- Operation
  - Send max-sized packet with "do not fragment" flag set
  - If encounters problem, ICMP message will be returned
    - "Destination unreachable: Fragmentation needed"
    - Usually indicates MTU encountered

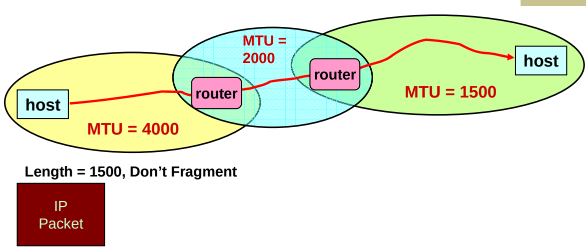
IP MTU Discovery with ICMP



IP MTU Discovery with ICMP



IP MTU Discovery with ICMP



- When successful, no reply at IP level
  - "No news is good news"
- Higher level protocol might have some form of acknowledgement

Important Concepts

- Base-level protocol (IP) provides minimal service level
  - Allows highly decentralized implementation
  - Each step involves determining next hop
  - Most of the work at the endpoints
- ICMP provides low-level error reporting
- IP forwarding → global addressing, alternatives, lookup tables
- IP addressing → hierarchical, CIDR
- IP service → best effort, simplicity of routers
- IP packets → header fields, fragmentation, ICMP

Next Lecture

- How do forwarding tables get built?
- Routing protocols
  - Distance vector routing
  - Link state routing

EXTRA SLIDES

The rest of the slides are FYI

Hierarchical Addressing Details

- Flat → would need router table entry for every single host… way too big
- Hierarchy → much like phone system…

- Hierarchy
  - Address broken into segments of increasing specificity
    - 412 (Pittsburgh area) 268 (Oakland exchange) 8734 (Seshan's office)
    - Pennsylvania / Pittsburgh / Oakland / CMU / Seshan
  - Route to general region and then work toward specific destination
  - As people and organizations shift, only update affected routing tables

Hierarchical Addressing Details

- Uniform Hierarchy
  - Segment sizes same for everyone
    - 412 (Pittsburgh area) 268 (Oakland exchange) 8734 (Seshan's office)
  - System is more homogeneous and easier to control
    - Requires more centralized planning
- Nonuniform Hierarchy
  - Number & sizes of segments vary according to destination
    - Pennsylvania / Pittsburgh / Oakland / CMU / Seshan
    - Delaware / Smallville / Bob Jones
  - System is more heterogenous & decentralized
    - Allows more local autonomy

CIDR

- Supernets
  - Assign adjacent net addresses to same org
  - Classless routing (CIDR)
- How does this help routing table?
  - Combine forwarding table entries whenever all nodes with same prefix share same hop

Aggregation with CIDR

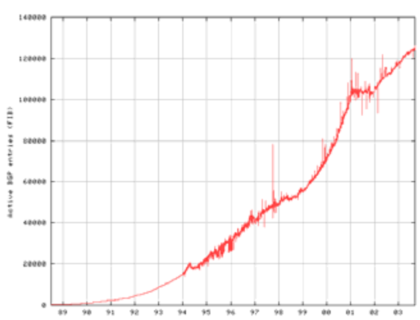
- Original Use: Aggregate Class C Addresses
- One organization assigned contiguous range of class C's
  - e.g., Microsoft given all addresses 207.46.192.X -- 207.46.255.X
  - Specify as CIDR address 207.46.192.0/18

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**

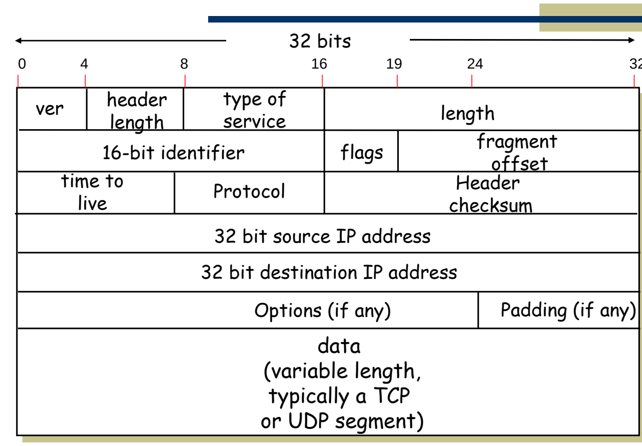
  - Represents 2^6 = 64 class C networks
- Use single entry in routing table
  - Just as if were single network address

Size of Complete Routing Table



- Source: www.cidr-report.org
- Shows that CIDR has kept # table entries in check
  - Currently require 124,894 entries for a complete table
  - Only required by backbone routers

IPv4 Header – RFC791 (1981)



ICMP: Internet Control Message Protocol

- Used by hosts, routers, gateways to communication network-level information
  - Error reporting: unreachable host, network, port, protocol
  - Echo request/reply (used by ping)
- Network-layer “above” IP:
  - ICMP msgs carried in IP datagrams
- ICMP message: type, code plus first 8 bytes of IP datagram causing error

Type	Code	description
0	0	echo reply (ping)
3	0	dest. network unreachable
3	1	dest host unreachable
3	2	dest protocol unreachable
3	3	dest port unreachable
3	6	dest network unknown
3	7	dest host unknown
4	0	source quench (congestion control - not used)
8	0	echo request (ping)
9	0	route advertisement
10	0	router discovery
11	0	TTL expired
12	0	bad IP header

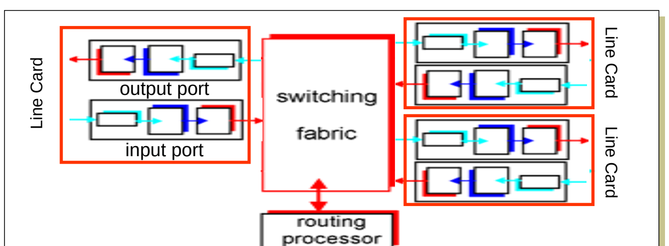
Outline

- Router Internals

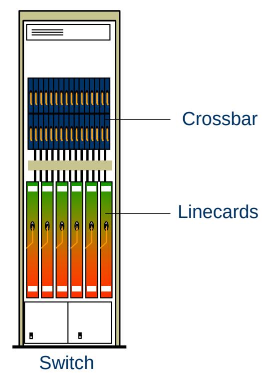
Router Architecture Overview

Two key router functions:

- Run routing algorithms/protocol (RIP, OSPF, BGP)
  - Done by routing processor
- *Switching* datagrams from incoming to outgoing link
  - Common case handled by line cards



Router Physical Layout



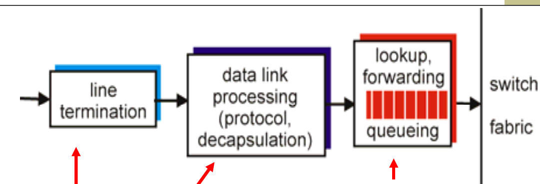
Juniper T series

Cisco 12000

Line Cards

- Often uses special purpose hardware (e.g. ASICs)
- Network interface cards
- Fast path (common-case) processing
  - Decrement TTL
  - Recompute checksum
  - Forward to next hop line card
    - Forwarding engine

Line Card: Input Port

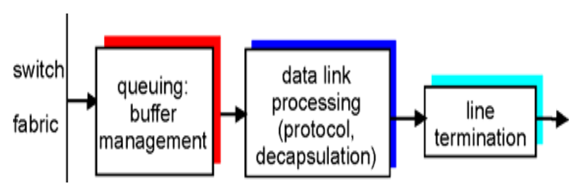


Physical layer:
bit-level reception

Data link layer:
e.g., Ethernet

Decentralized switching:
- Process common case (fast-path) packets
  - Decrement TTL, update checksum, forward packet
- Given datagram dest., lookup output port using routing table in input port memory
- Queue needed if datagrams arrive faster than forwarding rate into switch fabric

Line Card: Output Port



- Queuing required when datagrams arrive from fabric faster than the line transmission rate

Buffering

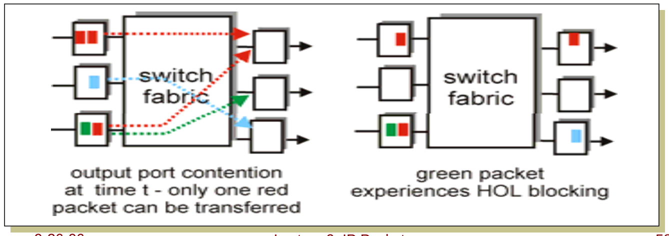
- Suppose we have N inputs and M outputs
  - Multiple packets for same output → output contention
  - Switching fabric may force different inputs to wait → Switch contention
- Solution – buffer packets when/where needed
- What happens when these buffers fill up?
  - Packets are THROWN AWAY!! This is where packet loss comes from

Switch Buffering

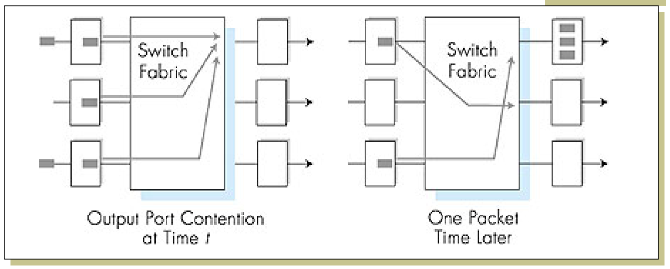
- 3 types of switch buffering
  - Input buffering
    - Fabric slower than input ports combined → queuing may occur at input queues
      - Can avoid any input queuing by making switch speed = N x link speed
  - Output buffering
    - Buffering when arrival rate via switch exceeds output line speed
  - Internal buffering
    - Can have buffering inside switch fabric to deal with limitations of fabric

Input Port Queuing

- Which inputs are processed each slot – schedule?
- Head-of-the-Line (HOL) blocking: datagram at front of queue prevents others in queue from moving forward



Output Port Queuing



- Scheduling discipline chooses among queued datagrams for transmission
  - Can be simple (e.g., first-come first-serve) or more clever (e.g., weighted round robin)

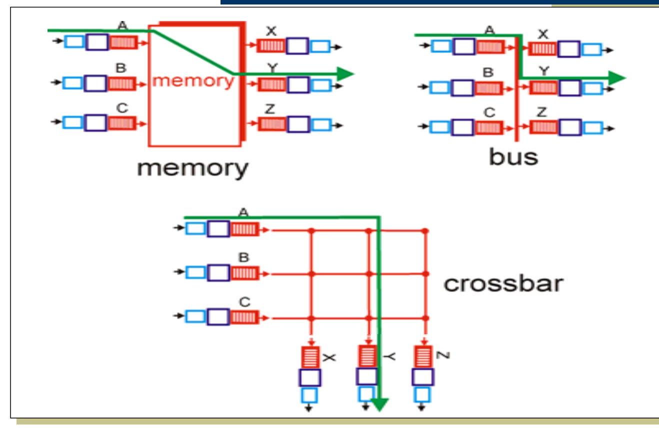
Virtual Output Queuing

- Maintain per output buffer at input
- Solves head of line blocking problem
- Each of MxN input buffer places bid for output
- Challenge: map bids to schedule of interconnect transfers

Network Processor

- Runs routing protocol and downloads forwarding table to forwarding engines
- Performs "slow" path processing
  - ICMP error messages
  - IP option processing
  - Fragmentation
  - Packets destined to router

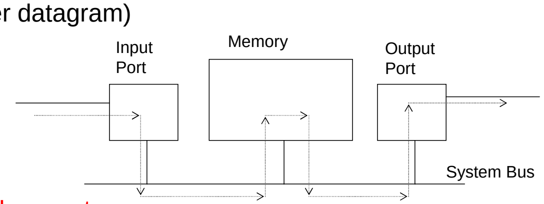
Three Types of Switching Fabrics



Switching Via a Memory

First generation routers → looked like PCs

- Packet copied by system's (single) CPU
- Speed limited by memory bandwidth (2 bus crossings per datagram)

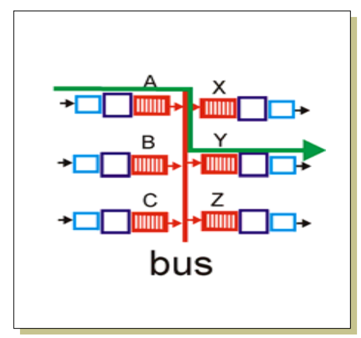


Modern routers

- Input port processor performs lookup, copy into memory
- Cisco Catalyst 8500

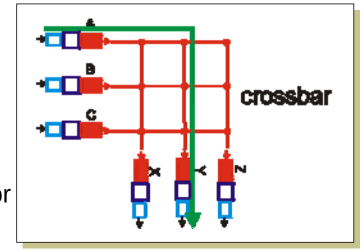
Switching Via a Bus

- Datagram from input port memory to output port memory via a shared bus
- Bus contention: switching speed limited by bus bandwidth
- 1 Gbps bus, Cisco 1900: sufficient speed for access and enterprise routers (not regional or backbone)



Switching Via an Interconnection Network

- Overcome bus bandwidth limitations
- Crossbar provides full NxN interconnect
  - Expensive
- Banyan networks & other interconnection nets initially developed to connect processors in multiprocessor
  - Typically less capable than complete crossbar
- Cisco 12000: switches Gbps through the interconnection network

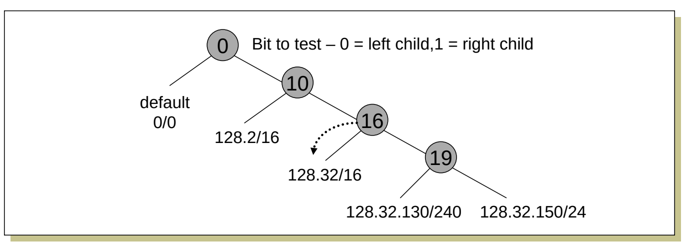


Outline

- Route Lookup

How To Do Longest Prefix Match

- Traditional method – Patricia Tree
  - Arrange route entries into a series of bit tests
- Worst case = 32 bit tests
  - Problem: memory speed is a bottleneck



Speeding up Prefix Match - Alternatives

- Content addressable memory (CAM)
  - Hardware based route lookup
  - Input = tag, output = value associated with tag
  - Requires exact match with tag
    - Multiple cycles (1 per prefix searched) with single CAM
    - Multiple CAMs (1 per prefix) searched in parallel
  - Ternary CAM
    - 0,1,don't care values in tag match
    - Priority (I.e. longest prefix) by order of entries in CAM

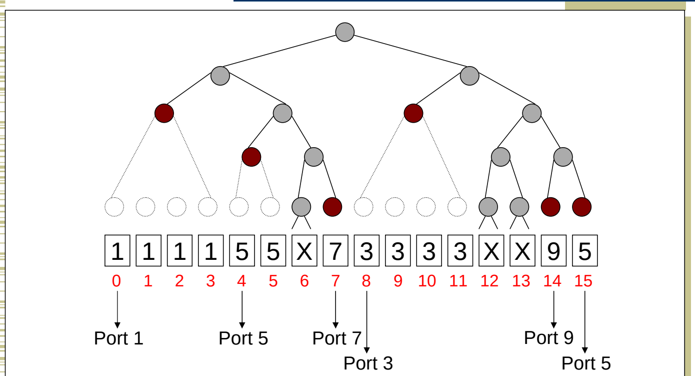
Speeding up Prefix Match - Alternatives

- Route caches
  - Packet trains → group of packets belonging to same flow
  - Temporal locality
  - Many packets to same destination
- Other algorithms
  - Routing with a Clue [Bremler-Barr – Sigcomm 99]
    - Clue = prefix length matched at previous hop
    - Why is this useful?

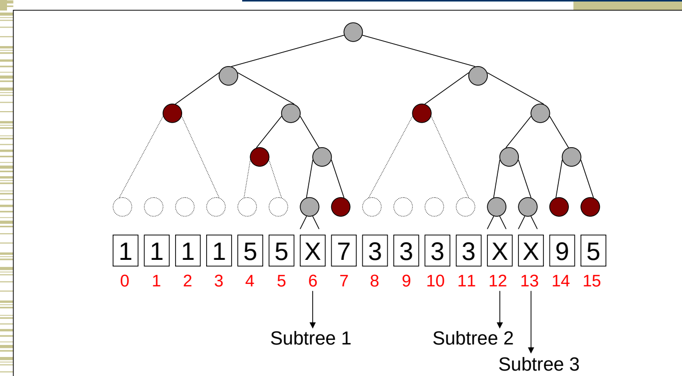
Speeding up Prefix Match - Alternatives

- Cut prefix tree at 16/24/32 bit depth
  - Fill in prefix tree entries by creating extra entries
    - Entries contain output interface for route
  - Add special value to indicate that there are deeper tree entries
    - Only keep 24/32 bit cuts as needed
- Example cut prefix tree at 16 bit depth
  - Only 64K entries

Prefix Tree



Prefix Tree



Cut Prefix Tree

- Scaling issues
  - How would it handle IPv6
- Other possibilities
  - Why were the cuts done at 16/24/32 bits?

Where did they learn all that network stuff….

- It takes years of training at top institutes to become CMU faculty ☺