Network Layer

Mike Freedman
COS 461: Computer Networks

http://www.cs.princeton.edu/courses/archive/spr14/cos461/

IP Protocol Stack: Key Abstractions

Application	Applications	
Transport	Reliable streams	Messages
Network	Best-effort *global* packet delivery	
Link	Best-effort *local* packet delivery	

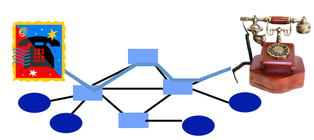
2

Best-Effort Global Packet Delivery

3

Circuit Switching (e.g., Phone Network)

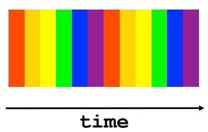
- Source establishes connection
  - Reserve resources along hops in the path
- Source sends data
  - Transmit data over the established connection
- Source tears down connection
  - Free the resources for future connections

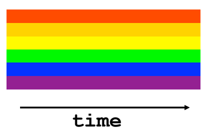


4

Circuit Switching: Static Allocation

Q: Frequency-Division vs. Time-Division

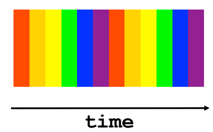


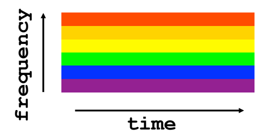


5

Circuit Switching: Static Allocation

- Time-division
  - Each circuit allocated certain time slots
- Frequency-division
  - Each circuit allocated certain frequencies

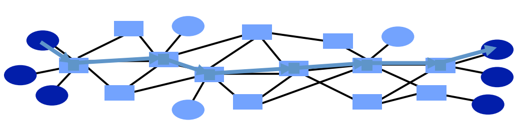




6

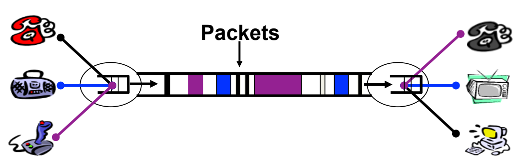
Packet Switching

- Message divided into packets
  - Header identifies the destination address
- Packets travel separately through the network
  - Forwarding based on the destination address
  - Packets may be buffered temporarily
- Destination reconstructs the message



7

Packet Switching: Statistical (Time Division) Multiplexing



- Intuition: Traffic by computer end-points is bursty!
  - Versus: Telephone traffic not bursty (e.g., constant 56 kbps)
  - One can use network while others idle
- Packet queuing in network: tradeoff space for time
  - Handle short periods when outgoing link demand > link speed

8

Best Effort: Celebrating Simplicity

- Packets may be lost, corrupted, reordered
- Never having to say you're sorry…
  - Don't reserve bandwidth and memory
  - Don't do error detection and correction
  - Don't remember from one packet to next
- Easier to survive failures
  - Transient disruptions are okay during failover
- Easier to support on many kinds of links
  - Important for *inter*connecting different networks

9

Is Best Effort Good Enough?

- Packet loss and delay
  - Sender can resend
- Packet corruption
  - Receiver can detect, and sender can resend
- Out-of-order delivery
  - Receiver can put the data back in order
- Packets follow different paths
  - Doesn't matter
- Network failure
  - Drop the packet
- Network congestion
  - Drop the packet

10

Packet vs. Circuit Switching?

• Predictable performance	Circuit
• Network never blocks senders	Packet
• Reliable, in-order delivery	Circuit
• Low delay to send data	Packet
• Simple forwarding	Circuit
• No overhead for packet headers	Circuit
• High utilization under most workloads	Packet
• No per-connection network state	Packet

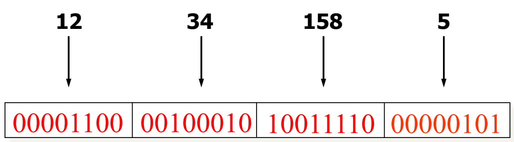
11

Network Addresses

12

IP Address (IPv4)

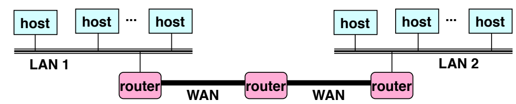
- A unique 32-bit number
- Identifies an interface (on a host, on a router, …)
- Represented in dotted-quad notation



13

Grouping Related Hosts

- The Internet is an "inter-network"
  - Used to connect networks together, not hosts
  - Need to address a network (i.e., group of hosts)



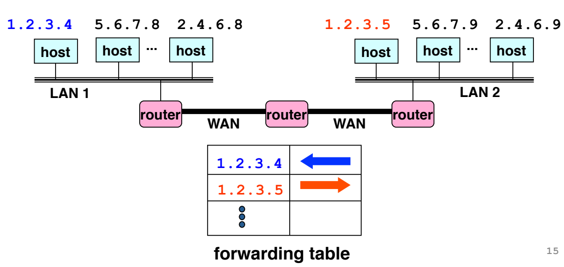
LAN = Local Area Network

WAN = Wide Area Network

14

Scalability Challenge

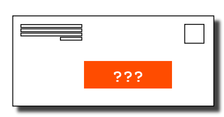
- Suppose hosts had arbitrary addresses
  - Then every router would need a lot of information
  - …to know how to direct packets toward every host



15

Hierarchical Addressing in U.S. Mail

- Addressing in the U.S. mail
  - Zip code: 08540
  - Building: 35 Olden Street
  - Room in building: 308
  - Name of occupant: Mike Freedman

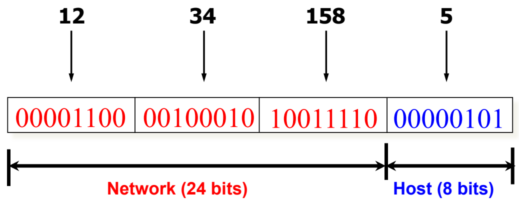


- Forwarding the U.S. mail
  - Deliver to the post office in the zip code
  - Assign to mailman covering the building
  - Drop letter into mailbox for building/room
  - Give letter to the appropriate person

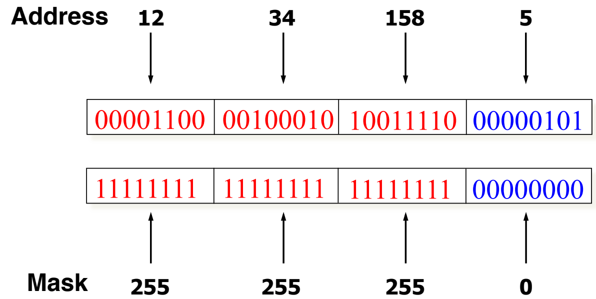
16

Hierarchical Addressing: IP Prefixes

- Network and host portions (left and right)
- 12.34.158.0/24 is a 24-bit **prefix** with 2^8 addresses

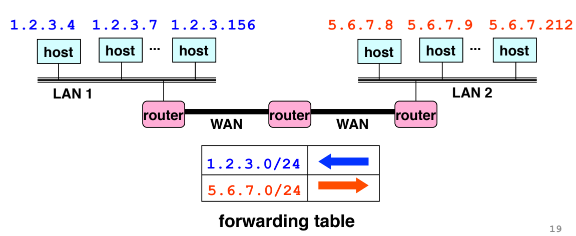


IP Address and 24-bit Subnet Mask



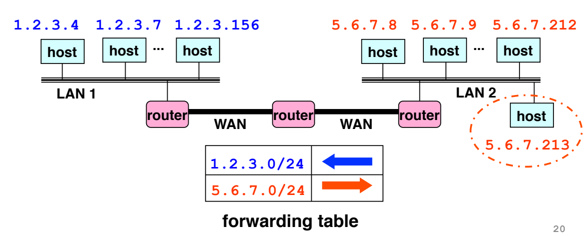
Scalability Improved

- Number related hosts from a common subnet
  - 1.2.3.0/24 on the left LAN
  - 5.6.7.0/24 on the right LAN



Easy to Add New Hosts

- No need to update the routers
  - E.g., adding a new host 5.6.7.213 on the right
  - Doesn't require adding a new forwarding-table entry



History of IP Address Allocation

Classful Addressing

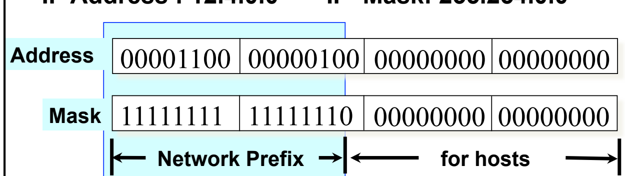
- In the olden days, only fixed allocation sizes
  - Class A: 0*
    - Very large /8 blocks (e.g., MIT has 18.0.0.0/8)
  - Class B: 10*
    - Large /16 blocks (e.g,. Princeton has 128.112.0.0/16)
  - Class C: 110*
    - Small /24 blocks (e.g., AT&T Labs has 192.20.225.0/24)
  - Class D: 1110* for multicast groups
  - Class E: 11110* reserved for future use
- This is why folks use dotted-quad notation!

Classless Inter-Domain Routing (CIDR)

- Use two 32-bit numbers to represent network:

  Network number = IP address + Mask

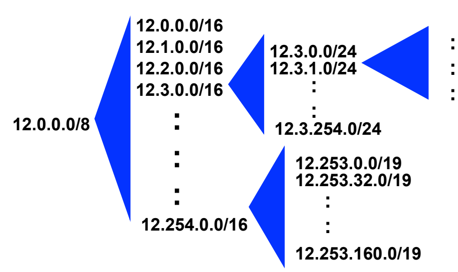
IP Address : 12.4.0.0 IP Mask: 255.254.0.0



Written as 12.4.0.0/15

Hierarchical Address Allocation

- Hierarchy is key to scalability
  - Address allocated in contiguous chunks (prefixes)
  - Today, the Internet has about 400,000 prefixes

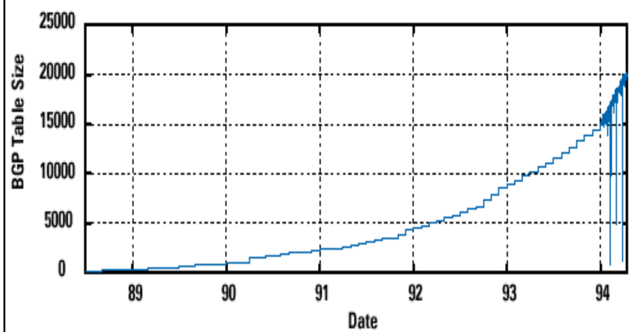


Obtaining a Block of Addresses

- Internet Corporation for Assigned Names and Numbers (ICANN)
  – Allocates large blocks to Regional Internet Registries
- Regional Internet Registries (RIRs)
  – E.g., ARIN (American Registry for Internet Numbers)
  – Allocates to ISPs and large institutions
- Internet Service Providers (ISPs)
  – Allocate address blocks to their customers
  – Who may, in turn, allocate to their customers…

25

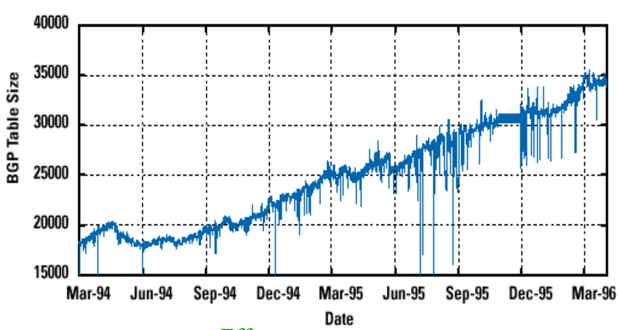
Pre-CIDR (1988-1994): Steep Growth



Growth faster than improvements in equipment capability

26

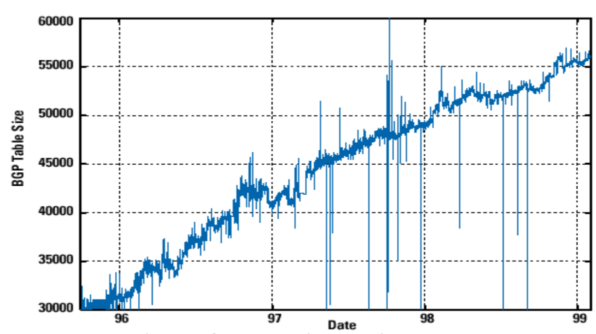
CIDR (1994-1996): Much Flatter



Efforts to aggregate

27

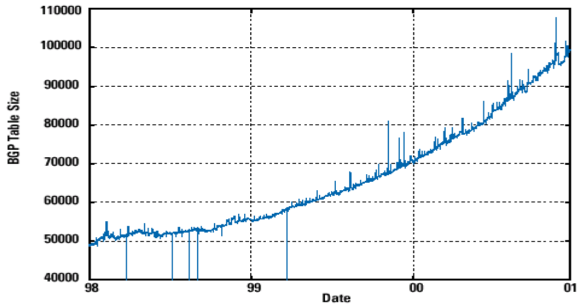
CIDR Growth (1996-1998): Roughly Linear



Good use of aggregation, and peer pressure!

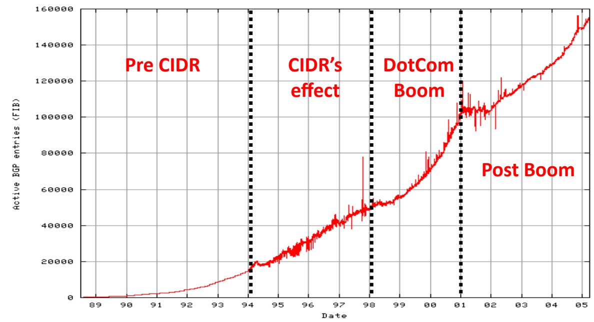
28

DotCom Boom (1998-2001): Steep Growth



Internet boom and increased multi-homing

Long Term Growth (1989-2005)



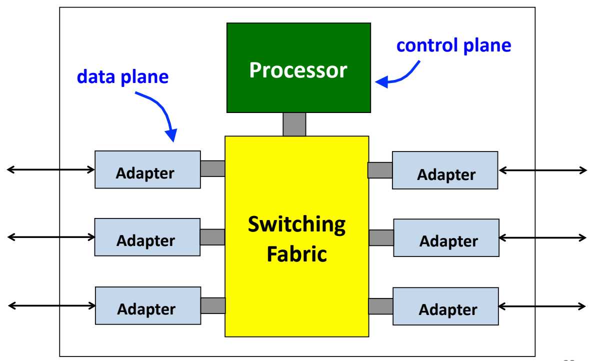
Today we are up to ~400,000 prefixes

Packet Forwarding

Hop-by-Hop Packet Forwarding

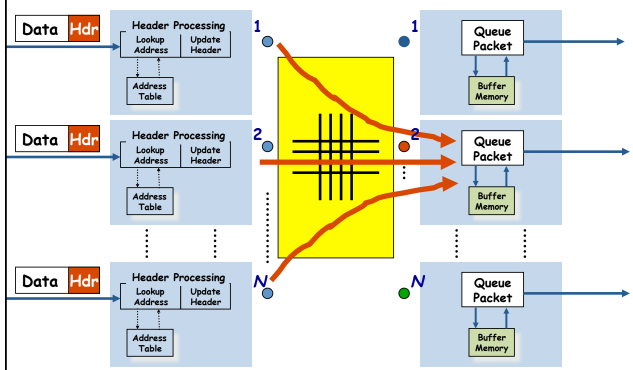
- Each router has a forwarding table
  - Maps destination address to outgoing interface
- Upon receiving a packet
  - Inspect the destination address in the header
  - Index into the table
  - Determine the outgoing interface
  - Forward the packet out that interface
- Then, the next router in the path repeats

IP Router



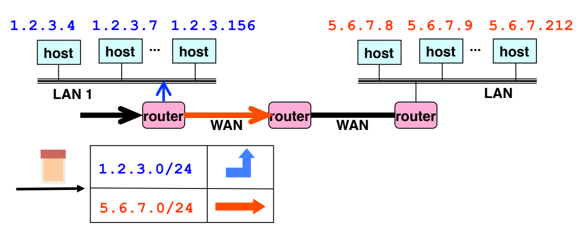
33

Switch Fabric: From Input to Output



Separate Forwarding Entry Per Prefix

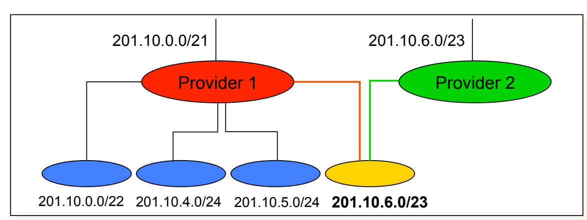
- Prefix-based forwarding
  - Map the destination address to matching prefix
  - Forward to the outgoing interface



35

CIDR Makes Packet Forwarding Harder

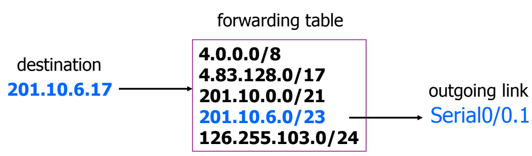
- Forwarding table may have many matches
  - E.g., entries for 201.10.0.0/21 and 201.10.6.0/23
  - The IP address 201.10.6.17 would match both!



36

Longest Prefix Match Forwarding

- Destination-based forwarding
  - Packet has a destination address
  - Router identifies longest-matching prefix
  - Cute algorithmic problem: very fast lookups



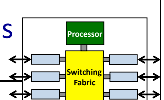
37

Creating a Forwarding Table

- Entries can be statically configured
  - E.g., "map 12.34.158.0/24 to Serial0/0.1"
- But, this doesn't adapt
  - To failures
  - To new equipment
  - To the need to balance load
- That is where the *control plane* comes in
  - Routing protocols

38

Data, Control, & Management Planes



	Data	Control	Management
Time-scale	Packet (ns)	Event (10 ms to sec)	Human (min to hours)
Tasks	Forwarding, buffering, filtering, scheduling	Routing, signaling	Analysis, configuration
Location	Line-card hardware	Router software	Humans or scripts

39

Q's: MAC vs. IP Addressing

- Hierarchically allocated
  A) MAC B) IP C) Both D) Neither
- Organized topologically
  A) MAC B) IP C) Both D) Neither
- Forwarding via exact match on address
  A) MAC B) IP C) Both D) Neither
- Automatically calculate forwarding by observing data
  A) Ethernet switches B) IP routers C) Both D) Neither
- Per connection state in the network
  A) MAC B) IP C) Both D) Neither
- Per host state in the network
  A) MAC B) IP C) Both D) Neither

40

Q's: MAC vs. IP Addressing

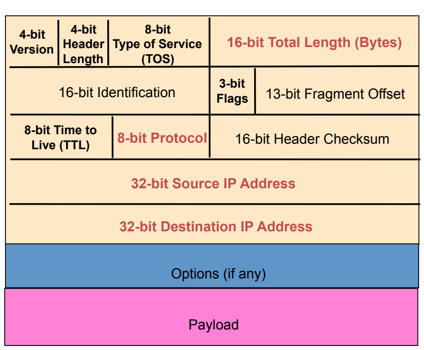
- Hierarchically allocated
  A) MAC B) IP **C) Both** D) Neither
- Organized topologically
  A) MAC **B) IP** C) Both D) Neither
- Forwarding via exact match on address
  A) MAC B) IP C) Both D) Neither
- Automatically calculate forwarding by observing data
  A) Ethernet switches B) IP routers C) Both D) Neither
- Per connection state in the network
  A) MAC B) IP C) Both **D) Neither**
- Per host state in the network
  A) MAC B) IP C) Both D) Neither

41

IP Packet Format

42

IP Packet Structure



43

Conclusion

- Best-effort global packet delivery
  - Simple end-to-end abstraction
  - Enables higher-level abstractions on top
  - Doesn't rely on much from the links below
- IP addressing and forwarding
  - Hierarchy for scalability and decentralized control
  - Allocation of IP prefixes
  - Longest prefix match forwarding
- Next time: transport layer

44

Backup Slides

45

IP Header: Version, Length, ToS

- Version number (4 bits)
  - Necessary to know what other fields to expect
  - Typically "4" (for IPv4), and sometimes "6" (for IPv6)
- Header length (4 bits)
  - Number of 32-bit words in the header
  - Typically "5" (for a 20-byte IPv4 header)
  - Can be more when "IP options" are used
- Type-of-Service (8 bits)
  - Allow different packets to be treated differently
  - Low delay for audio, high bandwidth for bulk transfer

46

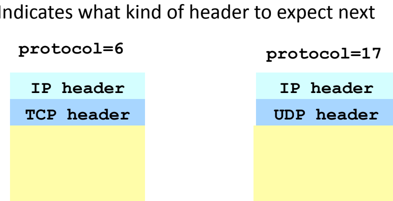
IP Header: Length, Fragments, TTL

- Total length (16 bits)
  - Number of bytes in the packet
  - Max size is 63,535 bytes (2^{16} -1)
  - … though most links impose smaller limits
- Fragmentation information (32 bits)
  - Supports dividing a large IP packet into fragments
  - … in case a link cannot handle a large IP packet
- Time-To-Live (8 bits)
  - Used to identify packets stuck in forwarding loops
  - … and eventually discard them from the network

47

IP Header: Transport Protocol

- Protocol (8 bits)
  - Identifies the higher-level protocol
    - E.g., "6" for the Transmission Control Protocol (TCP)
    - E.g., "17" for the User Datagram Protocol (UDP)
  - Important for demultiplexing at receiving host
    - Indicates what kind of header to expect next



48

IP Header: Header Checksum

- Checksum (16 bits)
  - Sum of all 16-bit words in the header
  - If header bits are corrupted, checksum won't match
  - Receiving discards corrupted packets

```
  134                 Mismatch!   134
+ 212                           + 216
-----        ==>                -----
= 346                           = 350
```


IP Header: To and From Addresses

- Destination IP address (32 bits)
  - Unique identifier for the receiving host
  - Allows each node to make forwarding decisions
- Source IP address (32 bits)
  - Unique identifier for the sending host
  - Recipient can decide whether to accept packet
  - Enables recipient to send a reply back to source