15-441/641: Computer Networks
The Internet Protocol

Fall 2019
Profs **Peter Steenkiste** & Justine Sherry

https://computer-networks.github.io/fa19/

Carnegie Mellon University

Outline

- The IP protocol
  - IPv4
  - IPv6
- IP in practice
  - NATs
  - Tunnels

Outline

- IP design goals
- Traditional IP addressing
  - Addressing approaches
  - Class-based addressing
  - Subnetting
  - CIDR
- Packet forwarding

Host Routing Table Example

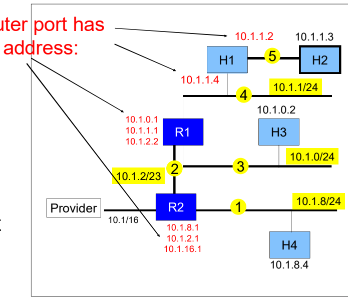
Destination	Gateway	Genmask	Iface
128.2.209.100	0.0.0.0	255.255.255.255	eth0
128.2.0.0	0.0.0.0	255.255.0.0	eth0
127.0.0.0	0.0.0.0	255.0.0.0	lo
0.0.0.0	128.2.254.36	0.0.0.0	eth0

- 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

- Five subnets (yellow)
  1. 10.1.8/24
  2. 10.1.2/23
  3. 10.1.0/24
  4. 10.1.0/24
  5. 10.1.1.3/31
- Packet to 10.1.1.3 arrives from Internet
- Path is R2 – R1 – H1 – H2
- H1 serves as a router for the 10.1.1.2/31 network (2 IP addresses)

Each router port has an IP address:

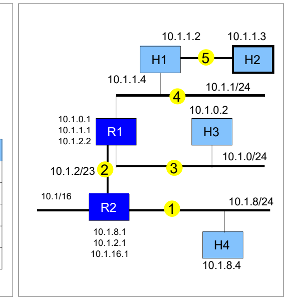


Routing Within the Subnet

- Packet to 10.1.1.3
- Matches 10.1.0.0/23

Routing table at R2

	Destination	Next Hop	Egress Port
	127.0.0.1	-	lo0
	Default or 0/0	provider	10.1.16.1
1	10.1.8.0/24	-	10.1.8.1
2	10.1.2.0/23	-	10.1.2.1
3 4 5	10.1.0.0/23	10.1.2.2	10.1.2.1

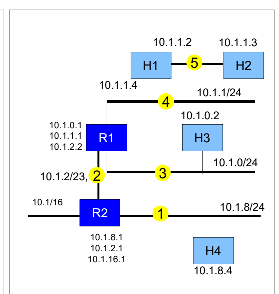


Routing Within the Subnet

- Packet to 10.1.1.3
- Matches 10.1.1.2/31
  - Longest prefix match

Routing table at R1

	Destination	Next Hop	Egress Port
	127.0.0.1	-	lo0
1	Default or 0/0	10.1.2.1	10.1.2.2
2	10.1.2.0/23	-	10.1.2.2
3	10.1.0.0/24	-	10.1.0.1
4	10.1.1.0/24	-	10.1.1.1
5	10.1.1.2/31	10.1.1.4	10.1.1.1

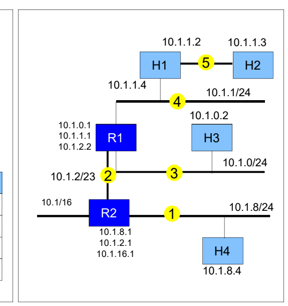


Routing Within the Subnet

- Packet to 10.1.1.3
- Direct route
  - Longest prefix match

Routing table at H1

	Destination	Next Hop	Egress Port
	127.0.0.1	-	lo0
1 2 3	Default or 0/0	10.1.1.1	10.1.1.4
4	10.1.1.0/24	-	10.1.1.2
5	10.1.1.2/31	-	10.1.1.2



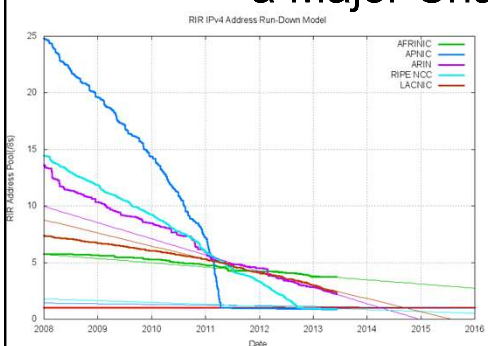
Important Concepts

- Hierarchical addressing critical for scalable system
  - Don't require everyone to know everyone else
  - Reduces number of updates when something changes
- Classless inter-domain routing supports more efficient use of address space
  - Adds complexity to routing, forwarding, …
    - But it is Scalable!

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?
  - Assigned by sys admin (static or dynamic)
  - 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

IP Address Availability Remains a Major Challenge



- Some are in big trouble!
- APNIC: Asia
- AFRINIC: Africa
- ARIN: North America
- LACNIC: Latin America
- RIPE NCC: Europe, Middle East, parts of central Asia

IP Service Model

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

IPv4 Packet Format

0	4	8	12	16	19	24	28 31
version	HLen	TOS		Length			
Identifier				Flag	Offset		
TTL		Protocol		Checksum			
Source Address							
Destination Address							
Options (if any)							
Data							

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
  - Informing the sender if packet does not make it
  - Delivery of packets in same order as they were sent
  - Just one copy of packet will arrive
- Implications
  - Scales very well (really, it does)
  - Higher level protocols must make up for shortcomings, e.g., TCP
  - Some services not feasible (or hard), e.g., latency or bandwidth guarantees

Designing the IP header

- Think of the IP header as an interface
  - Between the source and destination IP modules on end-systems
  - Between the source and network (routers)
  - Contains the information routers need to forward a packet
- Designing an interface
  - What task(s) are we trying to accomplish?
  - What information is needed to do it?
- Header reflects information needed for basic tasks

What are these tasks? (in network)

- Parse packet
- Carry packet to the destination
- Deal with problems along the way
  - Routing loops
  - Corruption
  - Packet too large
- Accommodate evolution
- Specify any special handling

What information do we need?

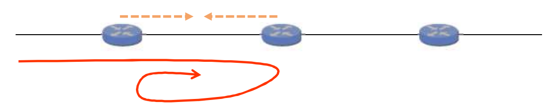
- Parse packet
  - *IP version number (4 bits), packet length (16 bits)*
- Carry packet to the destination
  - *Destination's IP address (32 bits)*
- Deal with problems along the way
  - Loops:
  - Corruption:
  - Packet too large:

What information do we need?

- Parse packet
  - *IP version number (4 bits), packet length (16 bits)*
- Carry packet to the destination
  - *Destination's IP address (32 bits)*
- Deal with problems along the way
  - Loops: *TTL (8 bits)*
  - Corruption: *checksum (16 bits)*
  - Packet too large: *fragmentation fields (32 bits)*

Preventing Loops (TTL)

- Forwarding loops cause packets to cycle for a very looong time
  - Would accumulate to consume all capacity if left unchecked



- Time-to-Live (TTL) Field (8 bits)
  - Decremented at each hop, packet discarded if reaches 0
  - … and "time exceeded" message is sent to the source

Header Corruption (Checksum)

- Checksum (16 bits)
  - Particular form of checksum over packet header
- If not correct, router discards packets
  - So it doesn't act on bogus information
- Checksum recalculated at every router
  - Why?

Fragmentation

- Every link has a "Maximum Transmission Unit" (MTU)
  - Largest number of bits it can carry as one unit
- A router can split a packet into multiple "fragments" if the packet size exceeds the link's MTU
- Must reassemble to recover original packet
- Will return to fragmentation shortly…

What information do we need?

- Parse packet
  - *IP version number (4 bits), packet length (16 bits)*
- Carry packet to the destination
  - *Destination's IP address (32 bits)*
- Deal with problems along the way
  - *TTL (8 bits)*, *checksum (16 bits), fragmentation (32 bits)*
- Accommodate evolution
  - *Version number (4 bits)* *(+ fields for special handling)*
- Specify any special handling

Special handling

- "Type of Service" (8 bits)
  - allow packets to be treated differently based on needs
    - e.g., indicate priority, congestion notification
  - has been redefined several times
- Now called "Differentiated Services Code Point (DSCP)"

Options

- Optional directives to the network
  - Not used very often
  - 16 bits of metadata + option-specific data
- Examples of options
  - Record Route
  - Strict Source Route
  - Loose Source Route
  - Timestamp
  - Various experimental options
  - …

IP Router Implementation: Fast Path versus Slow Path

- Common case: Switched in silicon ("fast path")
  - Almost everything
- Weird cases: Handed to a CPU ("slow path", or "process switched")
  - Fragmentation
  - TTL expiration (traceroute)
  - IP option handling

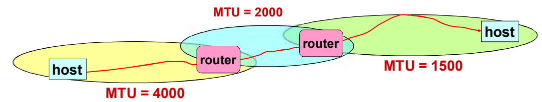
Bottom Line: Not Used!

- Slow path is evil in today's environment
  - "Christmas Tree" attack sets weird IP options, bits, and overloads router
  - Developers cannot (really) use things on the slow path
    - Slows down their traffic – not good for business
    - If it became popular, they are in trouble!

What information do we need?

- Parse packet
  - *IP version number (4 bits), packet length (16 bits)*
- Carry packet to the destination
  - *Destination's IP address (32 bits)*
- Deal with problems along the way
  - *TTL (8 bits), checksum (16 bits), fragmentation (32 bits)*
- Accommodate evolution
  - *version number (4 bits) (+ fields for special handling)*
- Specify any special handling
  - *ToS (8 bits), Options (variable length)*

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, router fragments packet
  - Destination host reassembles the paper – why?

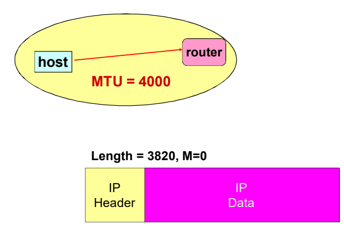
Fragmentation Related Fields

- 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)

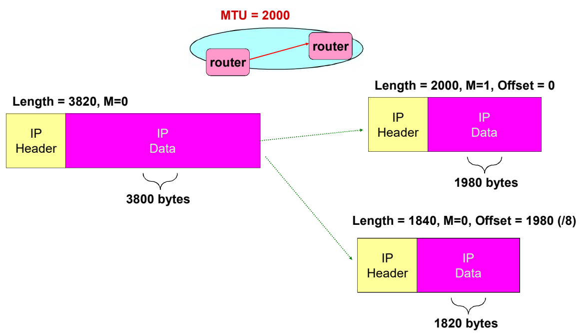
IPv4 Packet Format

0	4	8	12	16	19	24	28 31
version	HLen	TOS		Length			
Identifier				Flag	Offset		
TTL		Protocol		Checksum			
Source Address							
Destination Address							
Options (if any)							
Data							

IP Fragmentation Example #1



IP Fragmentation Example #2



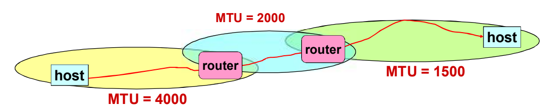
Fragmentation is Harmful

- Uses resources poorly
  - Forwarding costs per packet increases dramatically
  - Better 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)

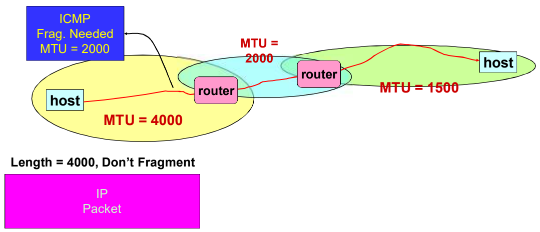
- Short messages used to send error & other control information
- Some functions supported by ICMP:
  - Ping request /response: check whether remote host reachable
  - Destination unreachable: Indicates how packet got & why couldn't go further
  - Flow control: Slow down packet transmit rate
  - Redirect: Suggest alternate routing path for future messages
  - Router solicitation / advertisement: Helps newly connected host discover local router
  - Timeout: Packet exceeded maximum hop limit
- How useful are they functions today?

IP MTU Discovery with ICMP

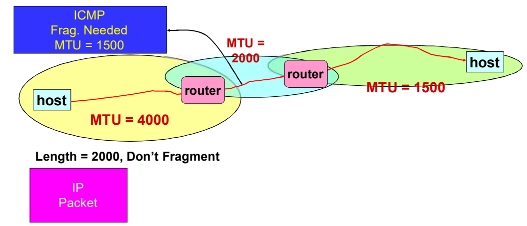


- Typically send series of packets from one host to another
- Typically, all will follow same route – routes are stable for minutes at a time
- Makes sense to determine path MTU before sending real packets
- Operation: Send max-sized packet with "do not fragment" flag set
  - If a router encounters a problem, it will return ICMP message to the sender
    - "Destination unreachable: Fragmentation needed"
    - Usually indicates MTU problem encountered
- ICMP abuse? Other solutions?

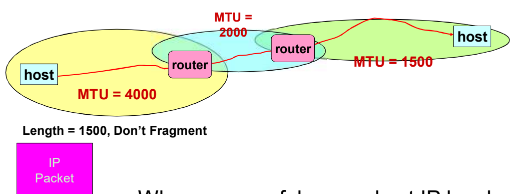
IP MTU Discovery with ICMP



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

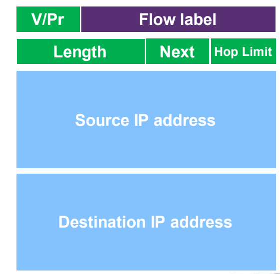
- IP forwarding → global addressing, alternatives, lookup tables
- IP addressing → hierarchical, CIDR
- IP service → best effort, simplicity of routers
- IP packets → header fields, fragmentation, ICMP
  - Interface to higher layers

Outline

- The IP protocol
  - IPv4
  - IPv6
- IP in practice
  - NATs
  - Tunnels

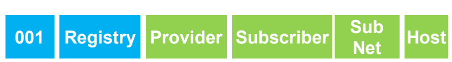
IPv6

- "Next generation" IP
- Most urgent issue: increasing address space.
  - 128 bit addresses
- Simplified header for faster processing:
  - No checksum (why not?)
  - No fragmentation (really?)
- Support for guaranteed services:
  - Priority and flow identifier
- Options handled as "next header"
  - reduces overhead of handling options



IPv6 Address Size Discussion

- Do we need more addresses? Probably, long term
  - Big panic in 90s: "We're running out of addresses!"
  - Big worry: Devices. Small devices. Cell phones, toasters, everything.
- 128 bit addresses provide space for structure (good!)
  - Hierarchical addressing is much easier
  - Assign an entire 48-bit sized chunk per LAN – use Ethernet addresses
  - Different chunks for geographical addressing, the IPv4 address space,
  - Perhaps help clean up the routing tables - just use one huge chunk per ISP and one huge chunk per customer.



IPv6 Header Cleanup: Options

- 32 IPv4 options → variable length header
  - Rarely used
  - No development / many hosts/routers do not support
    - Worse than useless: Packets w/options often even get dropped!
  - Processed in "slow path".
- IPv6 options: "Next header" pointer
  - Combines "protocol" and "options" handling
    - Next header: "TCP", "UDP", etc.
  - Extensions header: Chained together
  - Makes it easy to implement host-based options
  - One value "hop-by-hop" examined by intermediate routers
    - E.g., "source route" implemented only at intermediate hops

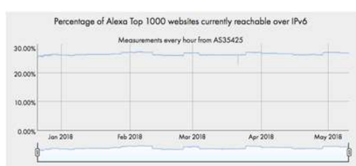
IPv6 Header Cleanup: “no”

- No checksum
  - Motivation was efficiency: If packet corrupted at hop 1, don’t waste b/w transmitting on hops 2..N.
  - Useful when corruption frequent, bandwidth expensive
  - Today: corruption is rare, bandwidth is cheap
- No fragmentation
  - Router discard packets, send ICMP “Packet Too Big” → host does MTU discovery and fragments
  - Reduced packet processing and network complexity.
  - Increased MTU a boon to application writers
  - Hosts can still fragment - using fragmentation header. Routers don’t deal with it any more.

Migration from IPv4 to IPv6

- Interoperability with IP v4 is necessary for incremental deployment.
  - No “flag day”
- Fundamentally hard because a (single) IP protocol is critical to achieving global connectivity across the internet
- Process uses a combination of mechanisms:
  - Dual stack operation: IP v6 nodes support both address types
  - Tunnel IP v6 packets through IP v4 clouds
  - IPv4-IPv6 translation at edge of network
    - NAT must not only translate addresses but also translate between IPv4 and IPv6 protocols
  - IPv6 addresses based on IPv4 – no benefit!
- 20 years later, this is still a major challenge!

Things are looking up?



Countries with IPv6 deployment greater than 15%

https://www.internetsociety.org/resources/2018/state-of-ipv6-deployment-2018/