Network (IP) Protocols #5

Outline

- Principles behind Internet protocols
- IP
    - Addressing
    - Forwarding
    - Tunneling
- IP Protocols
    - ICMP
    - DNS
    - ARP
    - DHCP
    - NAT
- Routing
- IPv6

Internetworking

TCP/IP

- Born out of the ARPA net in the late 1960's
- IP → Internet Protocol
- Transport Protocols
  - TCP → Transmission Control Protocol
  - UDP → User Datagram Protocol
  - Many others……..
- Open standard, runs on tablets, Smartphones, PC's to supercomputers and others….

More Later

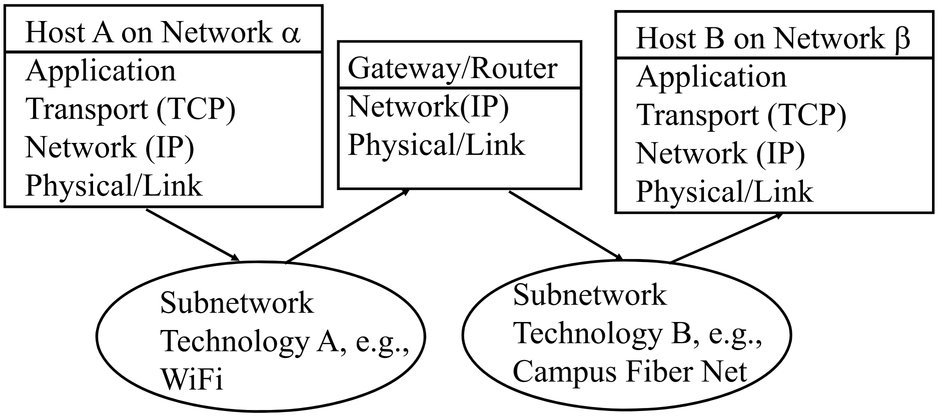
The Internet is more than IP

- A suite of protocols enable today's Internet
  - IP
  - ARP
  - DHCP
  - DNS
  - ICMP
  - NAT
  - Routing
    - IGP's (BGP)
    - EGP's (OSPF)

Internetworking:

- Application, e.g., Web/HTTP, e-mail-Simple Mail Transfer Protocol, (SMTP), streaming services
- Service Provider, end-to-end communications (TCP, UDP or other)
- Internetwork, functions to connect networks and routers (previously called gateways) into a total system, (IP)
- Subnetwork, e.g., WiFi, LTE, Ethernet, Bluetooth, Wireless, others…

Internetworking



Host A on Network α
Application
Transport (TCP)
Network (IP)
Physical/Link

Gateway/Router
Network(IP)
Physical/Link

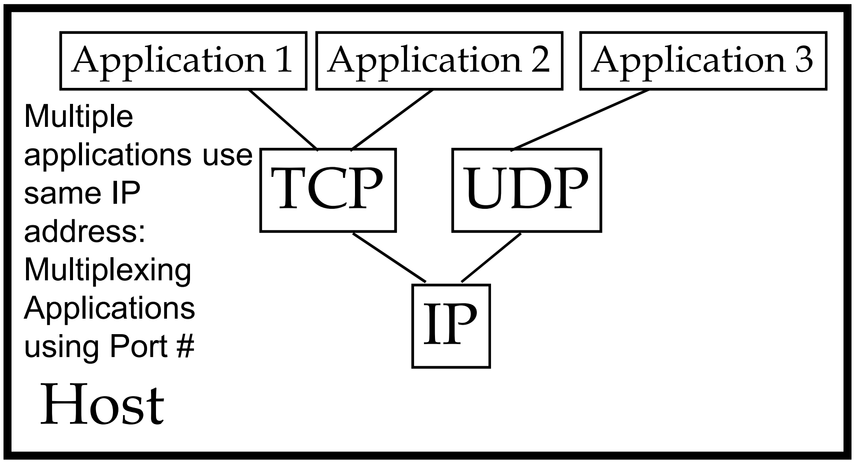
Host B on Network β
Application
Transport (TCP)
Network (IP)
Physical/Link

Subnetwork
Technology A, e.g.,
WiFi

Subnetwork
Technology B, e.g.,
Campus Fiber Net

IP Message units are called→Datagrams

Internetworking



Multiple applications use same IP address: Multiplexing Applications using Port #

Internet Design Principles

- Make sure it works
  - Do prototypes
  - Do not wait until standard documents are completed
- Keep it Simple
  - Best effort service model
  - End-to-end reliability
- Make clear choices→ goal to avoid multiple ways of accomplishing the same thing
- Exploit Modularity → protocol layers (Layered architecture)
- Expect Heterogeneity
  - Hardware
  - OSs
  - Transmission facilities
  - Applications

Modified from: "Computer Networks, 4rd Edition, A.S. Tanenbaum. Prentice Hall, 2002

Internet Design Principles

- Avoid static options and parameters→ best to negotiate or adapt
- Look for “good” design not optimum
- Be strict when sending and tolerant when receiving
- Scalability
  - **# users**
  - **Geographic scope**
  - **Transmission speeds**
- Consider performance and cost

Modified from: “Computer Networks, 4rd Edition, A.S. Tanenbaum. Prentice Hall, 2002

Network service model

Q: What *service model* for “channel” transporting datagrams from sender to receiver?

example services for *individual* datagrams:

- guaranteed delivery
- guaranteed delivery with less than 40 msec delay

example services for a *flow* of datagrams:

- in-order datagram delivery
- guaranteed minimum bandwidth to flow
- restrictions on changes in inter-packet spacing

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Network-layer service model

Network Architecture	Service Model	Quality of Service (QoS) Guarantees ?			
		Bandwidth	Loss	Order	Timing
Internet	best effort	none	no	no	no

Internet "best effort" service model

No guarantees on*:*

i. successful datagram delivery to destination
ii. timing or order of delivery
iii. bandwidth available to end-end flow

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Network-layer service model

Network Architecture	Service Model	Quality of Service (QoS) Guarantees ?			
		Bandwidth	Loss	Order	Timing
Internet	best effort	none	no	no	no
ATM	Constant Bit Rate	Constant rate	yes	yes	yes
ATM	Available Bit Rate	Guaranteed min	no	yes	no
Internet	Intserv Guaranteed (RFC 1633)	yes	yes	yes	yes
Internet	Diffserv (RFC 2475)	possible	possibly	no	no

ATM=Asynchronous transfer mode, now a legacy technology. ATM was developed in the 1990s and was used in telecommunications networks, but it has since been largely replaced by newer technologies like IP/MPLS and Ethernet

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

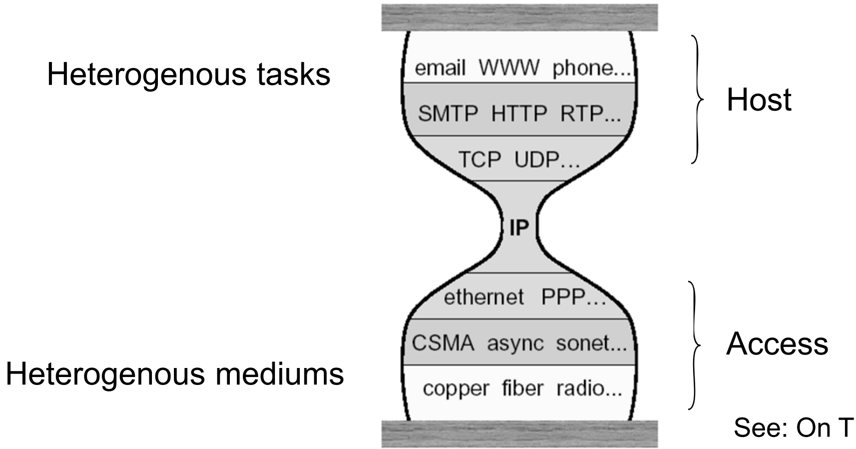
Reflections on best-effort service:

- simplicity of mechanism has allowed Internet to be widely deployed adopted
- sufficient provisioning of bandwidth allows performance of real-time applications (e.g., interactive voice, video) to be “good enough” for “most of the time”
- Link and transport layers often provides error recovery (discussed later)
- replicated, application-layer distributed services (datacenters, content distribution networks) connecting close to clients’ networks, allow services to be provided from multiple locations
- congestion control of “elastic” services helps

It’s hard to argue with success of best-effort service model for IP

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

IP Hourglass Architecture



See: On The Hourglass Model
https://vimeo.com/339192746

Modified from: Steve Deering
http://www.iab.org/Documents/hourglass-london-ietf.pdf

Problems with IP architecture

- End host assumptions
  - Not mobile
  - Address Binding → Coupling between IP address and end-device
- Security
  - Assumed friendly environment but in reality it is adversarial
- Economic model
  - Original architecture did not have an economic mode

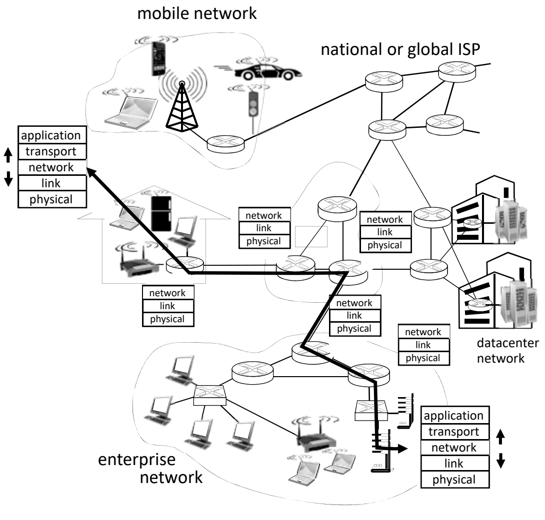
    → Causes inter-carrier problems with providing QoS
- Narrow hourglass model prevents applications awareness

  → new applications placing demands for core functionality
- These are currently addressed via point solutions → Middle boxes

---

Network-layer services and protocols

- transport segment from sending to receiving host
  - sender: encapsulates segments into datagrams, passes to link layer
  - receiver: delivers segments to transport layer protocol
- network layer protocols in *every Internet device*: hosts, routers
- routers:
  - examines header fields in all IP datagrams passing through it
  - moves datagrams from input ports to output ports to transfer datagrams along end-end path



Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

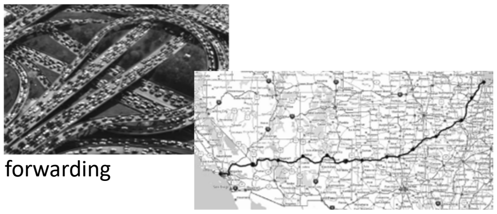
Two key network-layer functions

network-layer functions:

- *forwarding:* move packets from a router's input link to appropriate router output link
- *routing:* determine route taken by packets from source to destination
  - *routing algorithms*

analogy: taking a trip

- *forwarding:* process of getting through single interchange
- *routing:* process of planning trip from source to destination



forwarding

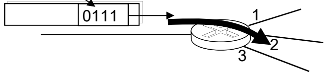
routing

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Network layer: data plane, control plane

Data plane:

- *local*, per-router function
- determines how datagram arriving on router input port is forwarded to router output port



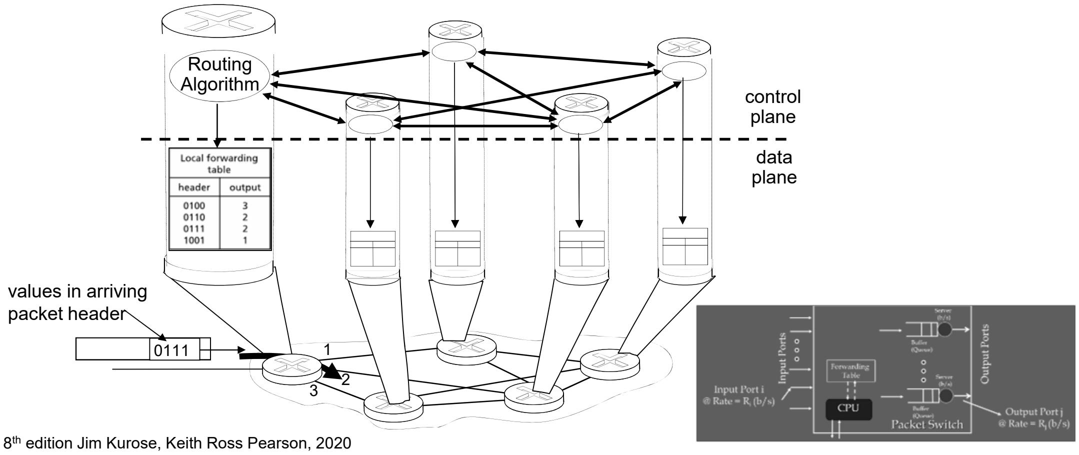
Control plane

- *network-wide* logic
- determines how datagram is routed among routers along end-end path from source host to destination host
- two control-plane approaches:
  - *traditional routing algorithms:* implemented in routers
  - *software-defined networking (SDN)*: implemented in (remote) servers

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Per-router control plane

Individual routing algorithm components *in each and every router* interact in the control plane



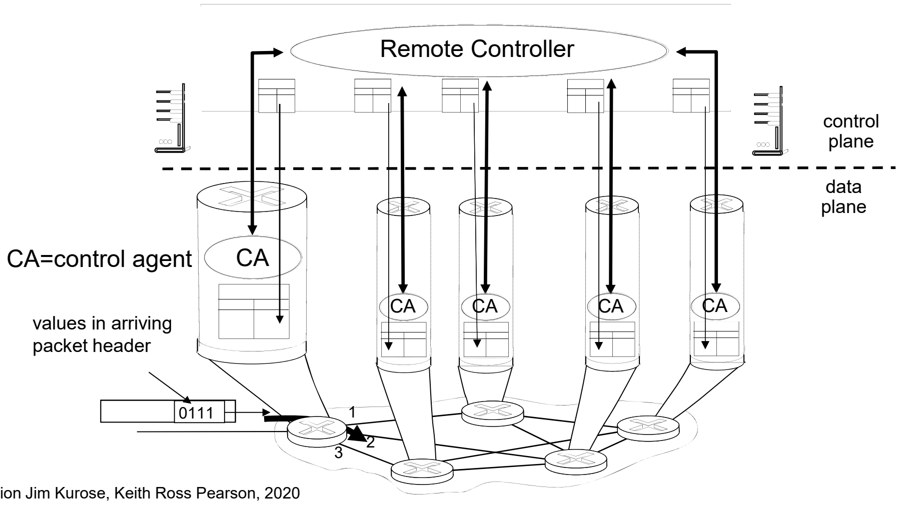
Local forwarding table

header	output
0100	3
0110	2
0111	2
1001	1

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Software-Defined Networking (SDN) control plane

Remote controller computes, installs forwarding tables in routers



Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

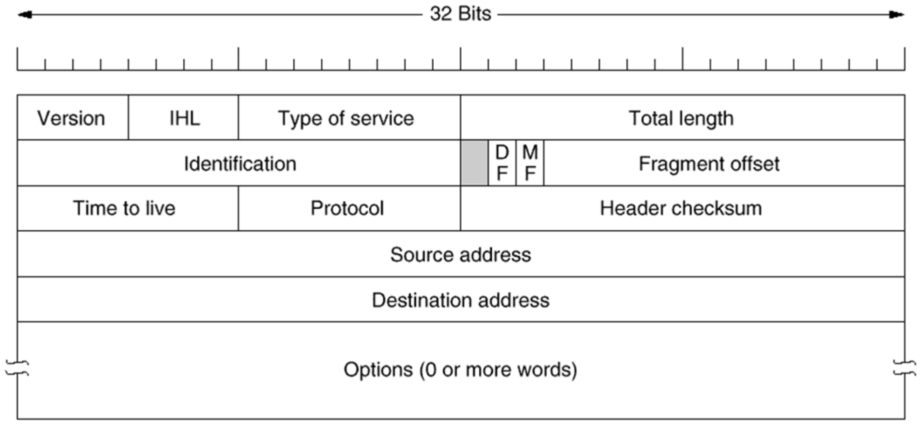
Internetworking: IP

IP is connectionless

- No call set up
- Each datagram treated independently
- Datagrams may be lost
- Hides the subnet technology from the application to allow the use of many different subnet technologies
- IP addresses identify device (e.g., host)

Internetworking: IPv4

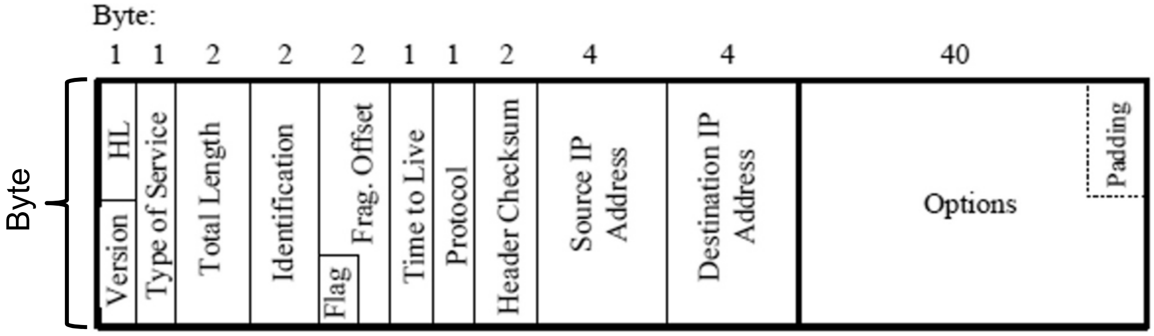
IP packet header



If no options then routers use “fast path” through hardware

From: “Computer Networks, 3rd Edition, A.S. Tanenbaum. Prentice Hall, 1996

IP packet header (byte-by-byte)



Examine IP by looking at the role of each field in the packet header

---

Internetworking: IP

IP packet header-IPv4

- Version, enable transition between different versions of IP datagrams, e.g., IPv4 and IPv6.
- IHL= Number of 32 bit words in the header
- ToS= Type of Service, enables the use of priority queuing, basis for IP DiffServ
- Maximum length of IP datagram (including header) = 65,535 bytes
- TTL field decremented at each **hop** if 0 then drop packet. Why? Time-to-live is NOT a time.
- Header Checksum verifies header only, Why?
- Identification and flag fields deal with fragmentation & reassembly
- Protocol field, identifies the associated transport protocol

TTL Field (Not a time!)

- The TTL field specifies the maximum number of hops (routers) that a packet can traverse before it is discarded. Each router that forwards a packet decrements the TTL field by one, and when the TTL field reaches zero, the packet is dropped and an ICMP (Internet Control Message Protocol) message is sent back to the source host.
- The TTL prevents IP packets from circulating indefinitely in a network due to routing loops or other errors. If a packet's TTL value is too high, it could end up consuming network resources indefinitely, causing congestion and potentially disrupting other network traffic.
- The TTL is also useful for troubleshooting network problems, optimizing routing policies, and improving network performance. TTL is used in traceroute (discussed later).

Header Checksum

- IP header uses check bits to detect errors in the ***header***
- A checksum is calculated for header contents
- Checksum recalculated at every router, so algorithm selected for ease of implementation Let header consist of L, 16-bit words,

  $\mathbf{b}_0, \mathbf{b}_1, \mathbf{b}_2, \ldots, \mathbf{b}_{L-1}$
- The algorithm appends a 16-bit *checksum* $\mathbf{b}_L$

From: Communication Networks:
Fundamentals Concepts and Key Architectures
Authors: A. Leon-Garcia and I. Widjaja

Checksum Field

The checksum $\mathbf{b}_L$ is calculated as follows:

- Treating each 16-bit word as an integer, find

  $\mathbf{x} = \mathbf{b}_0 + \mathbf{b}_1 + \mathbf{b}_2 + \ldots + \mathbf{b}_{L-1}$ modulo $2^{16}-1$
- The checksum is then given by:

  $\mathbf{b}_L = -\mathbf{x}$ modulo $2^{15}-1$
- This is the 16-bit 1's complement sum of the **b's**
- If checksum is 0, use all 1's representation (all zeros reserved to indicate checksum was not calculated)
- *Thus, the headers must satisfy the following **pattern***:

  $\mathbf{0} = \mathbf{b}_0 + \mathbf{b}_1 + \mathbf{b}_2 + \ldots + \mathbf{b}_{L-1} + \mathbf{b}_L$ modulo $2^{15}-1$

Check Sum

In IPv4 Routers need to recalculate the check sum because the header changes.
Why does the header change at each router?

Link to: How to Calculate IP Header Checksum (With an Example)

Modified from: Communication Networks:
Fundamentals Concepts and Key Architectures
Authors: A. Leon-Garcia and I. Widjaja

Differentiated Services:
Concept IP DiffServ-ToS Field

- Provide scalable service discrimination in the Internet
- No need to maintain **per flow** state or doing per hop signaling.
- Employs a small set of building blocks from which a variety of services can be built.
- These services can be either end-to-end or intra domain.

Differentiated Services:
Concept IP DiffServ ToS Field

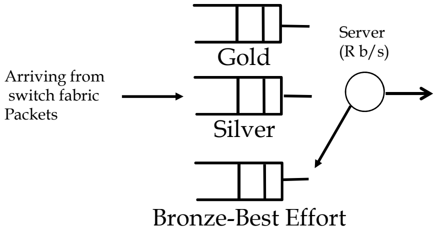
- Differentiated Services provide a wide range of services through:
  - Setting bits in the ToS at network edges and administrative boundaries,
  - Using those bits to determine how packets are treated (Queued/Served) by the routers inside the network, and
  - Conditioning the marked packets at network boundaries in accordance with the requirements of each service.
- Enable CoS in the Internet
- However, needs agreement across all networks to provide consistent performance.

IP DiffServ ToS Field

- Expedited Forwarding (EF): This service is used for time-sensitive and delay-sensitive traffic, such as voice or video. It provides a low-latency, low-jitter, and high-bandwidth service with a minimum delay and packet loss rate.
- Assured Forwarding (AF): This service is used for applications that require a certain level of service, but not necessarily low-latency or high-bandwidth. AF offers four classes of service with different priority levels and different levels of drop probability.
- Network Control (NC)—This class is typically high priority because it supports protocol control
- DiffServ enables network administrators the ability to ensure that critical traffic receives the necessary CoS while optimizing the use of network resources.

IP DiffServ ToS Field

ToS field is used to place IP packets in associated queue in the output ports of routers

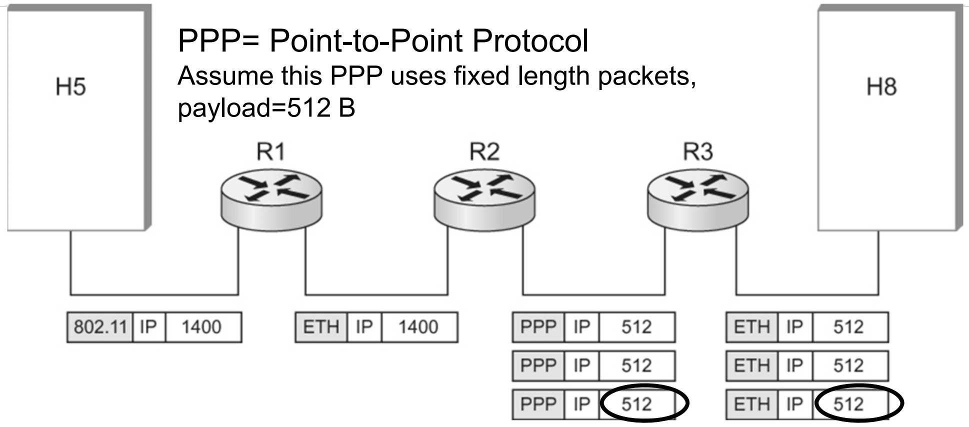


Packet to be clocked out selected from queue as per policy, e.g.,
Serve Gold packets,
if Gold queue empty the serve Silver packets,
if Gold and Silver Queues empty the serve BE packets

Fragmentation: Fragmentation offset

- Use of Flags, Fragment Offset, Identification
- Fragmentation and reassembly
  - At routers, within the network
  - If PDU size > Maximum Transfer Unit(MTU) for a subnet in the path the IP must fragment the PDU and reassemble at the destination
    - Ethernet ~1500 byte PDU's
    - PPP uses 512 bytes PDU's
  - IPv6 does not allow fragmentation
    - Not done at routers
    - Performed end-to-end, using an end-to-end MTU discovery process

Fragmentation and reassembly (IPv4)



PPP= Point-to-Point Protocol

Assume this PPP uses fixed length packets, payload=512 B

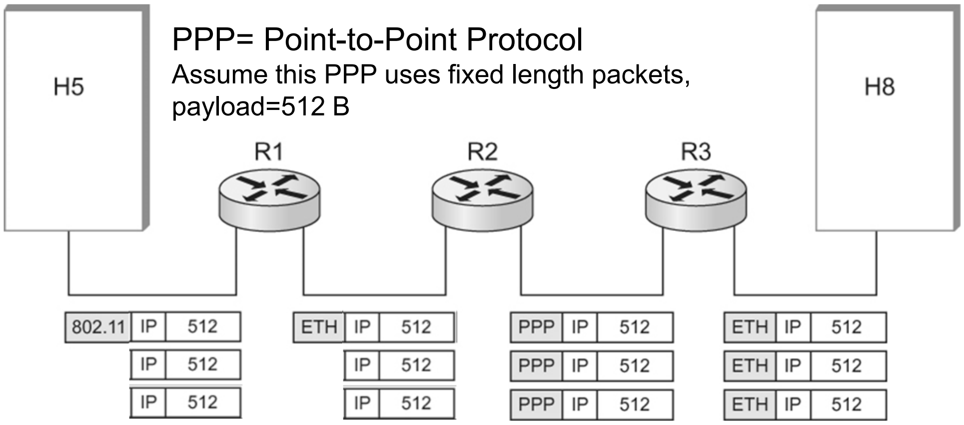
512+512+376=1400

in last PPP packet: Payload(376) +136 pad

Modified from: Computers Networks, Peterson and Davie,

In IPv6

- H5 and H8 discover the value of the MTU
- Discovery process uses ICMPv6 and algorithm similar to traceroute (discussed later)



PPP= Point-to-Point Protocol

Assume this PPP uses fixed length packets, payload=512 B

Modified from: Computers Networks, Peterson and Davie,

An address

- An address is a unique identifier that is used to locate or communicate with a particular location or entity. An address provides a way to identify a specific location or recipient for the purpose of delivering goods or information.
- Addresses can take many forms depending on the context in which they are used.
    - A postal address typically includes a street name, house number or apartment number, city, state or province, and zip or postal code.
    - Email addresses consist of a username followed by an @ symbol and the domain name of the email provider.
    - Internet Protocol (IP) addresses are a set of numerical values that identify devices on a network.
    - Socket address identifies an application on a host
    - Uniform Resource Locator (URL) is the address of a given unique resource on the Web
- Addresses are essential in facilitating communication and transactions. They allow for information to be delivered accurately and efficiently to the intended recipient. Accurate and up-to-date addresses are particularly important.

Hierarchy of Addresses

- Addresses are organized in a hierarchy to provide a structured and systematic way of identifying locations. This hierarchical structure allows for the easy and efficient management of addresses, as well as providing a way to scale up the system as more and more end-points (devices) are added to the network.
- The hierarchical structure typically starts with the largest entity (e.g., geographic), such as a country or state/province, and then progresses down to smaller entities such as cities, towns, neighborhoods, and finally specific buildings or units.
- In networks addresses are organized in a hierarchy to enable efficient routing of data between devices on a network, the hierarchy provides a systematic and organized way to identify and locate specific devices.
- This hierarchy makes it easy to locate and identify a specific location by using a series of progressively more specific identifiers.

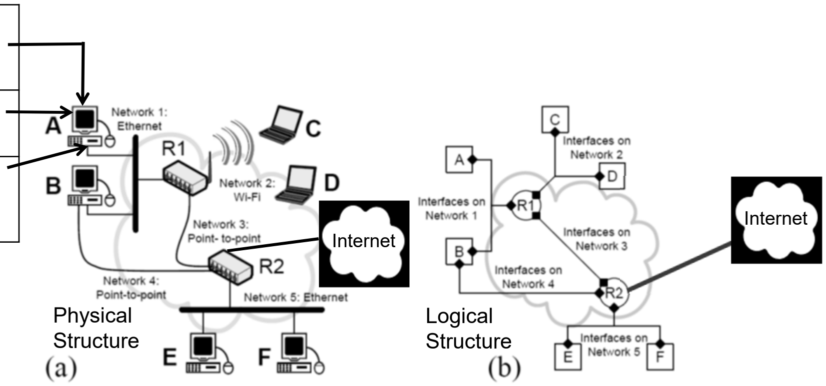
Addressing/Naming: Questions

- How are IP addresses are organized?
- How are IP addresses used to construct hierarchy?
- How to translate from names people use to addresses used in the network.
- How to get an address?
- PHY (Link layer) interfaces have addresses, typically PHY addresses are known locally. How associate an network address (known globally) to a PHY address?
- How are applications identified on the host that use the same IP address (more on this in the section on the transport layer)

Addressing

- Different layers contain different addresses:
  - Link Layer (Medium Access Control - MAC) address
  - Network Address (IP address)
  - Transport address (socket)

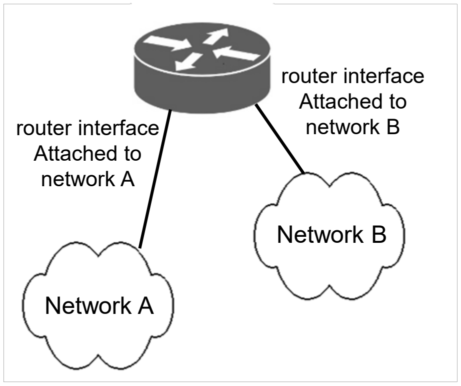
Application Interface (TCP/UDP/Other)	Socket Address (Known Locally)
Internetwork Interface	IP address (Known globally)
Physical Interface (Layer 2 address)	MAC address (Known Locally)

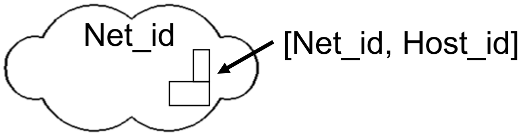


Modified from: Computer Networks: Performance and Quality of Service,
Ivan Marsic, Rutgers University, http://www.ece.rutgers.edu/~marsic/books/CN/

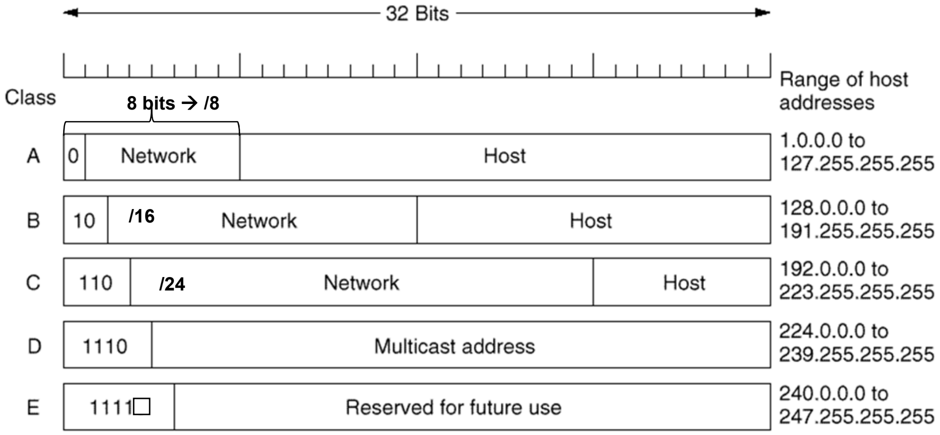
Internetworking: IP Addressing

- Every host (device) and router interface has an IP address
- 32 bits/address ➔ 4.295x10^9 addresses (IPv4)
  - Last of the IPv4 addresses allocated → in 2011
- 128 bits/address ➔ 3.4x10^{34} addresses (IPv6)
- Addresses contains
  - Host ID
    - Identifies a unique host on a network
  - Network ID
    - Identifies the network that the host is connected to
  - Initially five formats for IP addresses (Classfull IP Addressing)





Internetworking: Classfull IP Addressing & /x notation



From: "Computer Networks, 3rd Edition, A.S. Tanenbaum. Prentice Hall, 1996

Internetworking:
Classfull IP Addressing

- Class A addresses /8
  - 127 Class A addresses
  - 2^{24} hosts(16.77 Million)/Class A addresses
- Class B networks /16
  - 16383 Class B addresses (address '0' is reserved)
  - 2^{16} (65K) hosts/addresses
    KU has a class B address
- Class C addresses /24
  - 2,097,152 Class C addresses ('0' and '2,907,151' reserved)
    - 256 hosts/network.
- Class D is used for multicasting

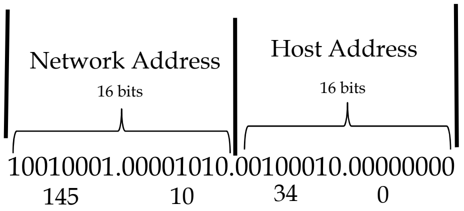
Internetworking:
IP Addressing Notation

- 32 bits = 4 bytes
- Represent each byte by a decimal
- Example: 11.55.31.84
  - 00001011 . 00110111 . 00011111 . 1010100
  - 11 55 31 84
- Example: 129.237.125.27 is a KU address
- Some tools will show the IP address in Hex, e.g., 129.237.125.27 is 81 ED 7D 1B, e.g., in wireshark the bits on the wire are shown in Hex

Internetworking: IP Addressing Notation

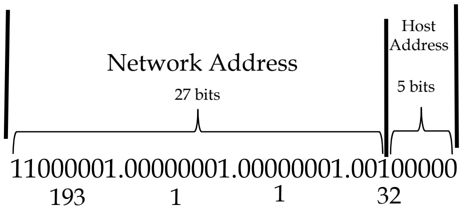
145.10.34.0/16

Prefix = 16 bits so this is a /16 Network



193.1.1.32/27

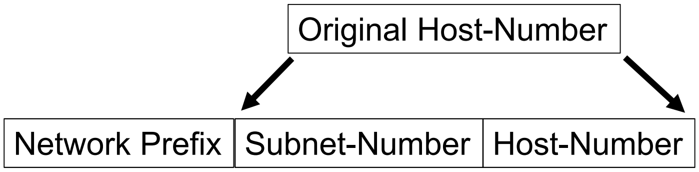
Prefix = 27 bits so this is a /27 Network



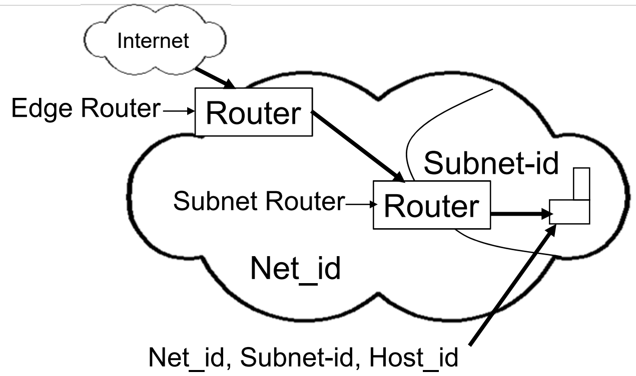
Internetworking: Subnetting

Subnetting divides the standard host number into:

- Subnet number
- Host – number



Internetworking: Subnetting



The "Internet" gets the packet to the Network, inside the network the packet is delivered to the Subnet router and then to the host.

Assumes final network is "broadcast"

Special addresses (Can not use for host ID):

- Address with host ID=all 0s refers to the network
- Address with host ID=all 1s refers to a broadcast packet, i.e., it goes to all host on the network

---

Internetworking: Subnetting

- Base Net: $\overbrace{11000001}^{193}.\overbrace{00000001}^{1}.\overbrace{00000001}^{1}$ .00000000 = 193.1.1.0/24
- Subnet #0: 11000001.00000001.00000001.000 00000 = 193.1.1.0/27
- Subnet #1: 11000001.00000001.00000001.001 00000 = 193.1.1.32/27
- Subnet #2: 11000001.00000001.00000001.010 00000 = 193.1.1.64/27
- Subnet #3: 11000001.00000001.00000001.011 00000 = 193.1.1.96/27
- Subnet #4: 11000001.00000001.00000001.100 00000 = 193.1.1.128/27
- Subnet #5: 11000001.00000001.00000001.101 00000 = 193.1.1.160/27
- Subnet #6: 11000001.00000001.00000001.110 00000 = 193.1.1.192/27
- Subnet #7: 11000001.00000001.00000001.111 00000 = 193.1.1.224/27

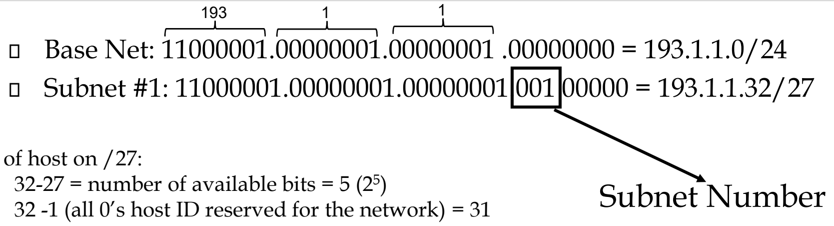
Subnet Number

Number of host on /27:

32-27 = number of available bits = 5 (2^5)
32 -1 (all 0's host ID reserved for the network) = 31
31 – 1 (all 1's host ID reserved for broadcast) = 30

Number of host on /27=30

Internetworking: Subnetting



- Base Net: 11000001.00000001.00000001 .00000000 = 193.1.1.0/24
- Subnet #1: 11000001.00000001.00000001 001 00000 = 193.1.1.32/27

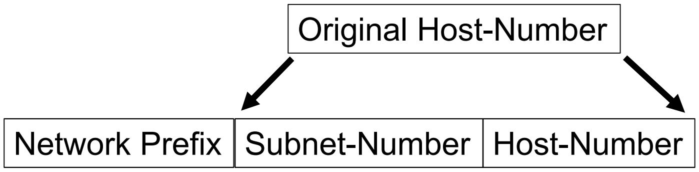
Subnet Number

Number of host on /27:
    32-27 = number of available bits = 5 (2^5)
    32 -1 (all 0's host ID reserved for the network) = 31
    31 – 1 (all 1's host ID reserved for broadcast) = 30
Number of host on /27=30

Subnet #1:
    Network address: 001 00000 or 193.1.1.32
    Broadcast address 001 11111 or 193.1.1.63
    Host addresses: 001 00001 to 001 11110 or 193.1.1.33 to 193.1.1.62

Internetworking: Subnetting

Subnetting divides the standard host number into:
- Subnet number
- Host – number

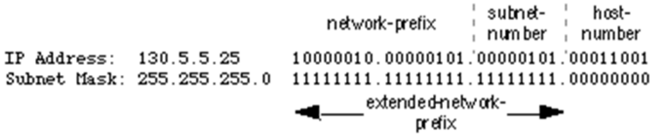


11000001.00000001.00000001.			101	01100
193	1	1	5	12

193.1.1.172/27

Internetworking: Subnetting

To identify the Subnet the router uses a “subnet mask”
Subnet mask has a “1” in each bit position of the address
except the host ID



```
130.5.5.25      10000010.00000101.00000101.00011001
255.255.255.0   11111111.11111111.11111111.00000000
                10000010.00000101.00000101 00000000
```

}AND

Logical AND to find network prefix

network prefix 130.5.5.0/24

From: Understanding IP Addressing: Everything You Ever Wanted To Know By Chuck Semeria http://www.3com.com/nsc/501302.html

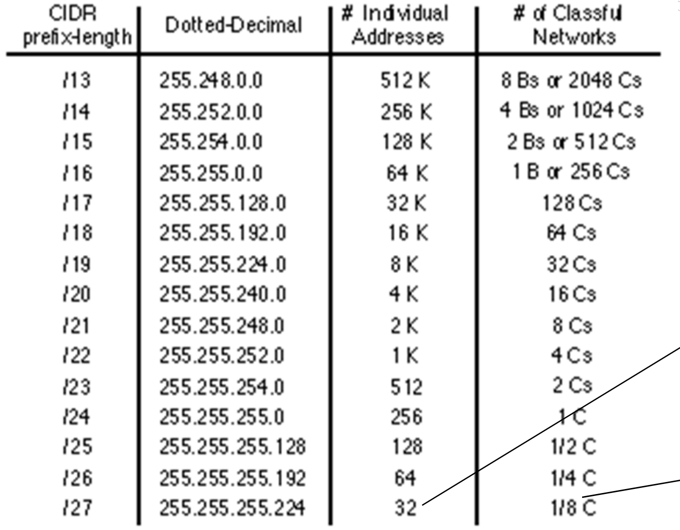
---

Internetworking: CIDR

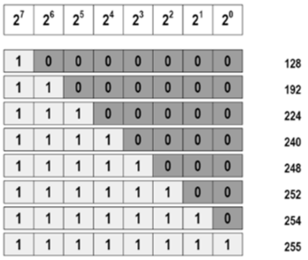
- Classless Interdomian Routing (CIDR)
- Removes the classful address restriction
- Extends the concept of subnetting to routers inside the Internet
- Partially relieves address exhaustion, allows more efficient use of IPv4 address space
- Supports deployment of arbitrarily sized networks
- Aggregation allows reduction in the size of routing tables

From: Understanding IP Addressing: Everything You Ever Wanted To Know By Chuck Semeria http://www.3com.com/nsc/501302.html

Internetworking: CIDR :



CIDR prefix-length	Dotted-Decimal	# Individual Addresses	# of Classful Networks
/13	255.248.0.0	512 K	8 Bs or 2048 Cs
/14	255.252.0.0	256 K	4 Bs or 1024 Cs
/15	255.254.0.0	128 K	2 Bs or 512 Cs
/16	255.255.0.0	64 K	1 B or 256 Cs
/17	255.255.128.0	32 K	128 Cs
/18	255.255.192.0	16 K	64 Cs
/19	255.255.224.0	8 K	32 Cs
/20	255.255.240.0	4 K	16 Cs
/21	255.255.248.0	2 K	8 Cs
/22	255.255.252.0	1 K	4 Cs
/23	255.255.254.0	512	2 Cs
/24	255.255.255.0	256	1 C
/25	255.255.255.128	128	1/2 C
/26	255.255.255.192	64	1/4 C
/27	255.255.255.224	32	1/8 C



2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
1	0	0	0	0	0	0	0	128
1	1	0	0	0	0	0	0	192
1	1	1	0	0	0	0	0	224
1	1	1	1	0	0	0	0	240
1	1	1	1	1	0	0	0	248
1	1	1	1	1	1	0	0	252
1	1	1	1	1	1	1	0	254
1	1	1	1	1	1	1	1	255

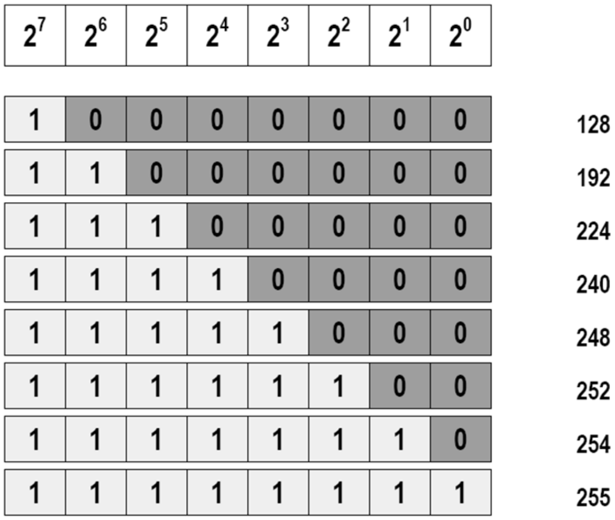
hosts/
Subnet-2

of
subnets

Modified from: Understanding IP Addressing: Everything You Ever Wanted To Know By Chuck Semeria http://www.3com.com/nsc/501302.html

---

Possible Subnet Mask Values



2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	
1	0	0	0	0	0	0	0	128
1	1	0	0	0	0	0	0	192
1	1	1	0	0	0	0	0	224
1	1	1	1	0	0	0	0	240
1	1	1	1	1	0	0	0	248
1	1	1	1	1	1	0	0	252
1	1	1	1	1	1	1	0	254
1	1	1	1	1	1	1	1	255

Examples of
subnet masks:
255.254.0.0
255.128.0.0
255.255.192.0

Modified from: https://www.ict.tuwien.ac.at/skripten/datenkomm/infobase/L30-IP_Technology_Basics_v4-6.pdf

Domain Name System: DNS

- Domain Name System (DNS)
  Names ←→ IP translation
- Non-numeric form for IP addresses host naming
  - host.department.institution.domain
- Names are long and human understandable
  - Wastes space to carry them in packet headers
  - Hard to parse
- Numeric addresses are shorter and machine understandable
  - If fixed size, easy to carry in headers and parse
- DNS distributed database implemented in hierarchy of many name servers

---

Domain Name System: DNS

- IP Addressing ->Example gauss.eecs.ku.edu=> 129.237.125.220
- A different IP address can be assigned to each physical interface on a host, note a physical interface will have a unique physical address, for IEEE 802.3 this is a 48-bit number
- A host can have multiple IP addresses: multihomed
- See https://who.is/
  - www.ku.edu
  - 129.237.11.76

Domain Name System: DNS

- Names also constructed in hierarchy
- DNS services
  - Domain name system (DNS) contain tables to convert:
    - host.department.institution.domain to a 32-bit address
  - Indirection
    - Multiple names may point to same address
    - Can move a machine and just update the resolution table
  - Host aliasing
  - Mail server aliasing
  - Load distribution
    - replicated Web servers: many IP addresses correspond to one name

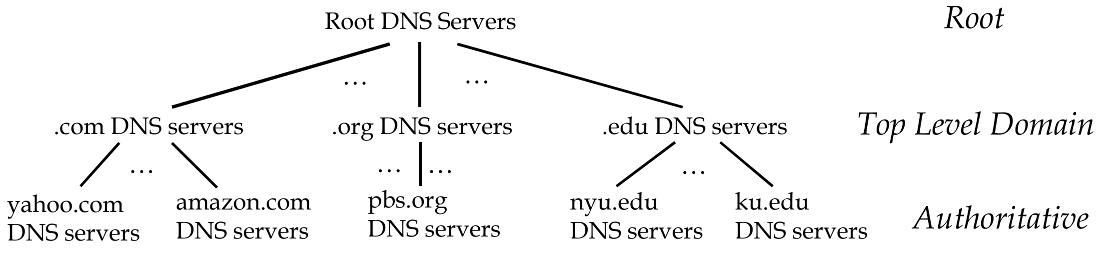
---

Domain Name System: DNS

- DNS is a real time distributed data base
  (maybe the worlds largest)
- Records in the DNS database include:
  - A → Maps name to IP
  - PTR (pointer) → Maps name to name (alias)
  - MX (mail exchange) → Maps name to name of mail server
- DNS is a critical infrastructure for the Internet.

Modified from: *Computer Networking: A Top-Down Approach*
8th edition Jim Kurose, Keith Ross Pearson, 2020

DNS: a distributed, hierarchical database



Root DNS Servers

Root

.com DNS servers

.org DNS servers

.edu DNS servers

Top Level Domain

yahoo.com DNS servers

amazon.com DNS servers

pbs.org DNS servers

nyu.edu DNS servers

ku.edu DNS servers

Authoritative

Client wants IP address for www.amazon.com; 1st approximation:

- client queries root server to find .com DNS server
- client queries .com DNS server to get amazon.com DNS server
- client queries amazon.com DNS server to get IP address for www.amazon.com

Not Efficient

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

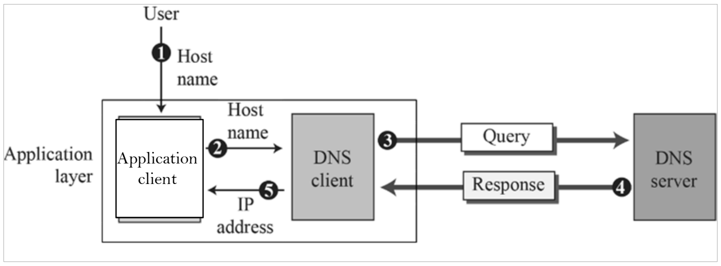
Local and Authoritative DNS name servers

- does not strictly belong to hierarchy
- each ISP (residential ISP, company, university) has one
  - also called "default name server"
  - default name servers resolve DNS queries by finding and retrieving information from authoritative DNS servers.
- when host makes DNS query, query is sent to its local DNS server
  - has local cache of recent name-to-address translation pairs (but may be out of date!)
  - acts as proxy, forwards query into hierarchy
- Authoritative DNS servers:
  - organization's own DNS server(s), providing authoritative hostname to IP mappings for organization's named hosts
  - can be maintained by organization or service provider
  - default name servers resolve DNS queries by finding and retrieving information from authoritative DNS servers.

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Domain Name System: DNS

1. User gives name to application client
2. Application client passes name to local DNS client
3. At boot time the local host is configured with the IP address of at least one DNS server. The DNS client sends a query to the DNS server to get the IP address associated with the name.
4. The DNS server responds with the IP address
5. The local DNS client passes the IP address to the application
6. The application now associates that name with an IP address
7. The local DNS client caches results



See:

a. DNS servers using ipconfig /all
b. DNS cache using ipconfig /displaydns

From: Data Communications and Networking 5th Edition by Behrouz A. Forouzan
and Computer and Communication Networks, 2nd Edition. Nader F. Mir. Prentice Hall

---

DNS: root name servers

- official, contact-of-last-resort by name servers that can not resolve name
- *incredibly important* Internet function
  - Internet couldn't function without it!
  - DNSSEC – provides security (authentication and message integrity)
- ICANN (Internet Corporation for Assigned Names and Numbers) manages root DNS domain

13 logical root name "servers" worldwide each "server" replicated many times (~200 servers in US)

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

DNS: a distributed, hierarchical database

Very large distributed database:
- ~ billion records, each simple

Handles many *trillions* of queries/day:
- *many* more reads than writes
- *performance matters:* almost every Internet transaction interacts with DNS - msecs count!

Organizationally, physically decentralized:
- millions of different organizations responsible for their records

“bulletproof”: reliability, security

Top level domains to naming authorities (see Internet Corporations for Assigned Names and Numbers- ICANN; http://www.icann.org)

.edu
.com
.mil
.org
.gov
.net
.biz
.{country} .il, .uk, .au
More…

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

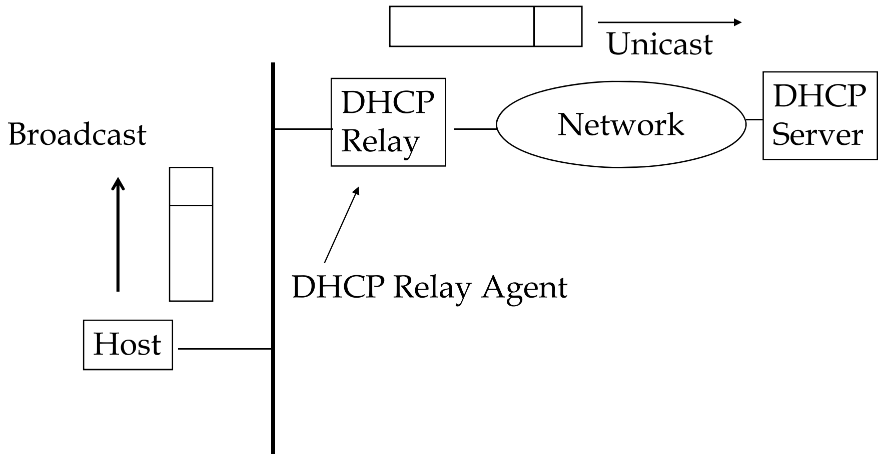
Host Configuration: Dynamic Host Configuration Protocol (DHCP)

- Every host needs an IP address
- Initial approach: System Administrators manually configure host IP information (static)
- Management nightmare for large enterprise networks
- Management nightmare for “always on” public networks, e.g., cable modem systems

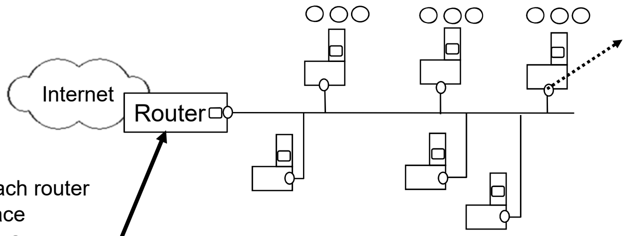
Host Configuration: Dynamic Host Configuration Protocol (DHCP)

- Solution: DHCP
- DHCP server maintains pool of IP addressed that are distributed on demand.
- The protocol governs the distribution of addresses
- DHCP enables the scaling of network management

Host Configuration: Dynamic Host Configuration Protocol (DHCP)



PHY/Layer 2/MAC and IP Addresses



Interface PHY “listens”
for its address
(in a broadcast message)

1. Typically each router PHY interface connects to a different network
2. Each router PHY interface has and MAC and IP address

o Network interface PHY address (PHY or MAC)

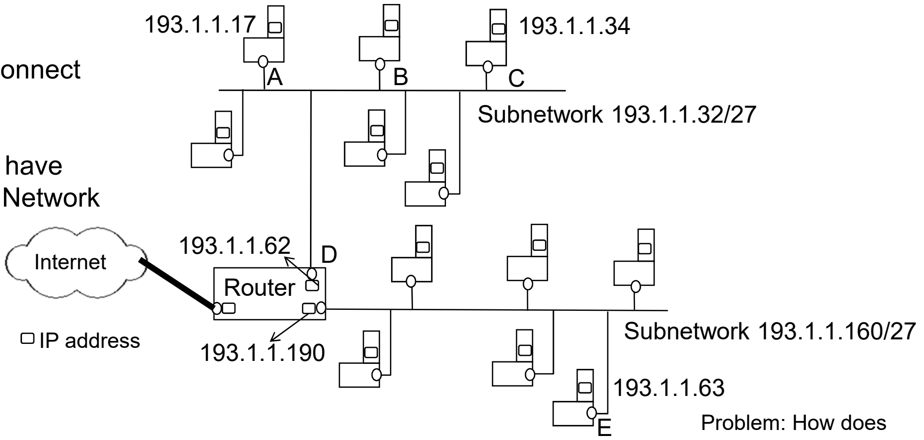
▭ IP address

○ Socket (e.g., port # identifying applications in the host)

PHY and IP Addresses and Networks

Topology:
Routers connect networks.

Networks have their own Network address.



▭ IP address

Problem: How does
host with IP 193.1.1.17
send a packet to 193.1.1.34?

o Network interface PHY address (PHY or MAC)
Here A, B, C, D, E

Internet Control Protocols: ARP (On Ethernet)

- Address Resolution Protocol (ARP)
  - Purpose: Map IP address to physical address (or link layer address)
- Want to talk to 129.237.116.75
- Send MAC ***"broadcast"*** message: **Who owns 129.237.116.75**
- 129.237.116.75 will respond: **I do and here is my physical address**
- Reverse ARP (RARP)
- Maps Physical address into IP address

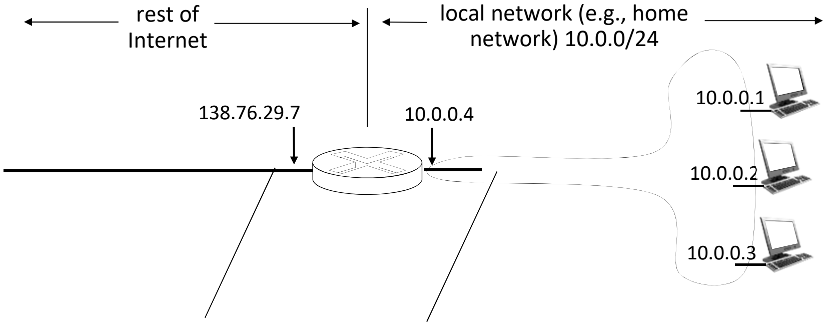
NAT: Network Address Translation

- NAT is an partial/alternate solution to IPv4 address exhaustion
- Use a private IP address internally while sharing one external IP address
- Need identifier to map private internal IP to external IP
- Look ahead; in TCP/UDP packet header there is a 16 bit field for port #, normally port # are used to identify processes in a host.
- NAT "highjacks" the port # and uses it as part of an private host identifier, so in a NAT router:
  - outgoing datagrams: replace (source IP address, port #) of every outgoing datagram to (NAT IP address, new port #)
  - . . . remote clients/servers will respond using (NAT IP address, new port #) as destination addr.
  - In the NAT translation table every (source IP address, port #) is mapped to (NAT IP address, new port #) translation pair
  - incoming datagrams: replace (NAT IP address, new port #) in dest fields of every incoming datagram with corresponding (source IP address, port #) stored in NAT table

Modified from *Computer Networking: A Top Down Approach Featuring the Internet*, 4nd edition. Jim Kurose, Keith Ross, Addison-Wesley, Copyright 1996-2002, J .F Kurose and K.W. Ross, All Rights Reserved

NAT: network address translation

NAT: all devices in local network share just one IPv4 address as far as outside world is concerned



all datagrams *leaving* local network have *same* source NAT IP address: 138.76.29.7, but *different* source port numbers

datagrams with source or destination in this network have 10.0.0/24 address for source, destination (as usual)

NAT: network address translation

- all devices in local network have 32-bit addresses in a “private” IP address space (10/8, 172.16/12, 192.168/16 prefixes) that can only be used in local network
- advantages:
  - just one IP address needed from provider ISP for *all* devices
  - can change addresses of host in local network without notifying outside world
  - can change ISP without changing addresses of devices in local network
  - security: devices inside local net not directly addressable, visible by outside world

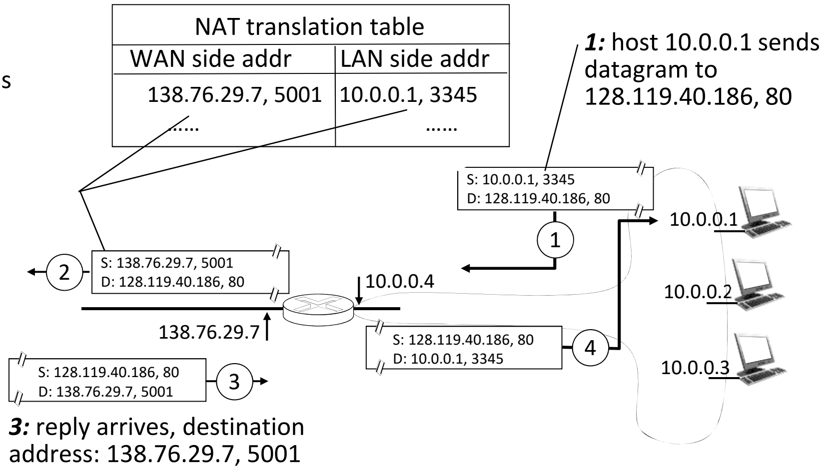
NAT: network address translation

implementation: NAT router must (transparently):

- outgoing datagrams: replace (source IP address, port #) of every outgoing datagram to (NAT IP address, new port #)
    - remote clients/servers will respond using (NAT IP address, new port #) as destination address
- remember (in NAT translation table) every (source IP address, port #) to (NAT IP address, new port #) translation pair
- incoming datagrams: replace (NAT IP address, new port #) in destination fields of every incoming datagram with corresponding (source IP address, port #) stored in NAT table

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

NAT: network address translation



NAT translation table

WAN side addr	LAN side addr
138.76.29.7, 5001	10.0.0.1, 3345
……	……

2: NAT router changes datagram source address from 10.0.0.1, 3345 to 138.76.29.7, 5001, updates table

1: host 10.0.0.1 sends datagram to 128.119.40.186, 80

3: reply arrives, destination address: 138.76.29.7, 5001

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

NAT: network address translation

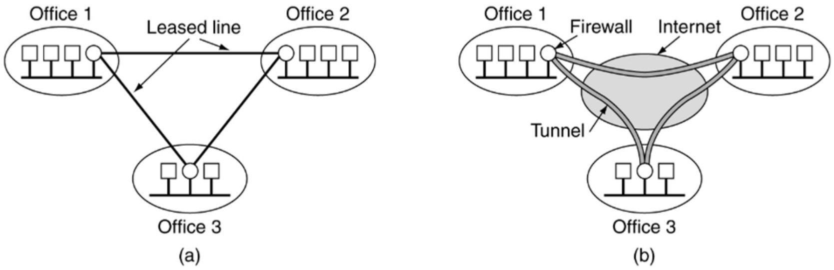
- NAT overloads the port # construct
- Violates end-to-end argument (hour glass model), applications developers now may need to take NAT into account.
- NAT has been controversial:
  - routers “should” only process up to layer 3
  - address “shortage” should be solved by IPv6
  - violates end-to-end argument (port # manipulation by network-layer device)
  - NAT traversal: what if client wants to connect to server behind NAT?
- but NAT is here to stay:
  - extensively used in home and institutional nets, 4G/5G cellular nets
  - NAT enables the Universal Plug and Play (UPnP) set of protocols, (UPnP is designed to simplify the process of connecting devices to a network and configuring them for use.)

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Tunneling

- A tunnel is a *virtual* point-to-point connection between a pair of nodes through an arbitrary number of networks
- Packet entering a tunnel is encapsulated into another packet
- Packet leaving the tunnel is de-encapsulated restoring the original packet format

VPN (virtual private network)

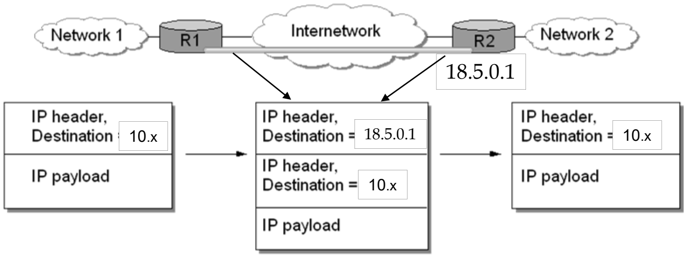


(a) A leased-line private network. (b) A virtual private network.

From: Computer Networks, A. S. Tanenbaum, Prentice Hall, 2003

Tunneling: Example of IP-IP tunnel

Goals:
- Enable the use Private Addressing Scheme inside enterprises
- Enable security, i.e., secure tunnel forming a Virtual Private Network (VPN)

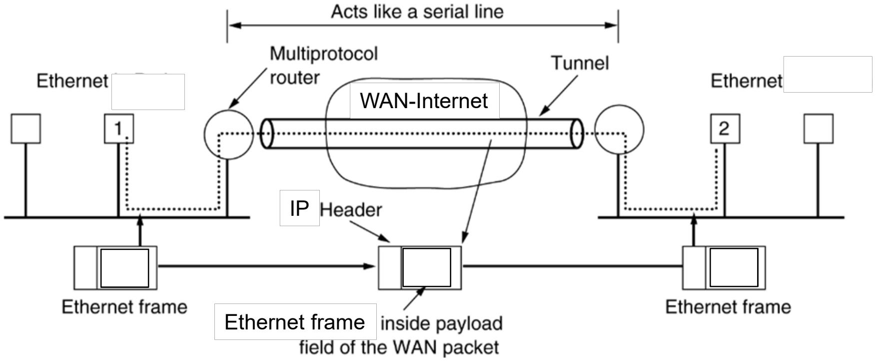


Private (internal) addresses are not routed on the Internet and no traffic can be sent to them from the Internet, they only supposed to work within the local network.
Example of private IP addresses:
Range from 10.0.0.0 to 10.255.255.255 — a 10.0.0.0 network with a 255.0.0.0 or /8 (an 8-bit) mask

Modified from: "Computer Networks", L. Peterson and B. Davie, Morgan Kaufman, 2000

Tunneling

Example of Ethernet ove IP tunnel



Modified from: "Computer Networks, 4rd Edition, A.S. Tanenbaum. Prentice Hall, 2003

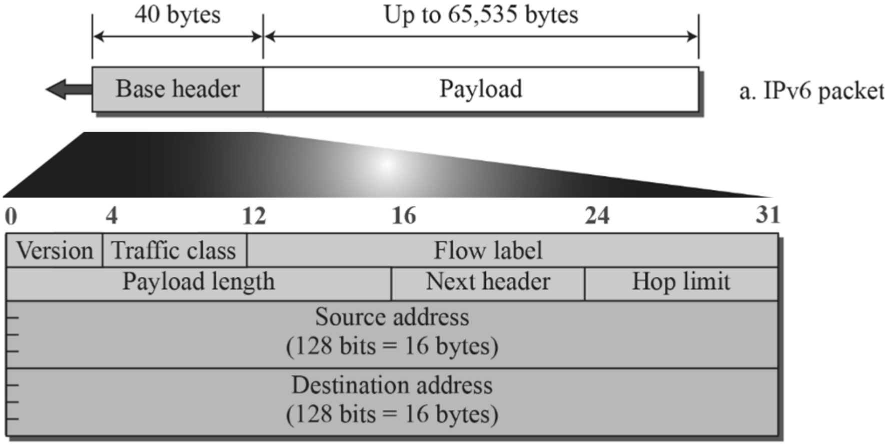
Tunneling: Benefits & Penalties

- Benefits
  - Enables *"virtual private networks"*
  - Allows address independence in the enterprise
  - Enhances security (with encryption)
  - Enables gateway functionality, carry other PDUs formats (protocols) across an IP network
- Penalties
  - Increased overhead: packets are longer
  - Performance of edge routers: routers must add and remove encapsulation
  - Management: tunnel set up

IPv6

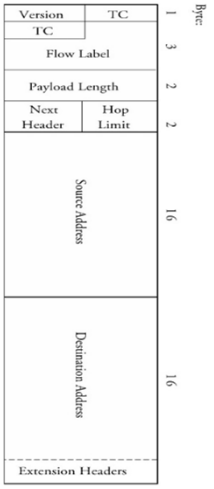
- In 2011 all IPv4 addresses were assigned
- IPv6 →Longer addresses
  - 128 bits/address (16 bytes) → 3.4x10^{38} addresses
  - Valid IPv6 address: 1002:DB78:7DF0:D5E9:976C:74ED:0FA1:89C1 (in hexadecimal)
  - IPv6 can use \n CIDR notation to identify network prefix
- Simplified header –
  - 64 bit aligned
  - Longer but fewer fields
  - All fields are of fixed size
  - Easier to process at high speeds.
- Better options support → encoded in optional extension headers
- Flow label to support differentiated services

IPv6 Header



a. IPv6 packet

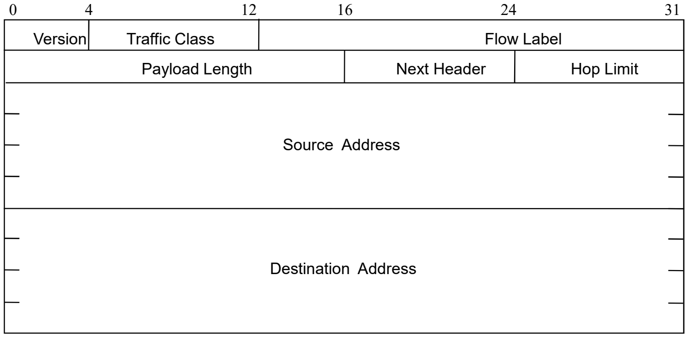
b. Base header



From: Data Communications and Networking 5th Edition by Behrouz A. Forouzan
and Computer and Communication Networks, 2nd Edition. Nader F. Mir. Prentice Hall

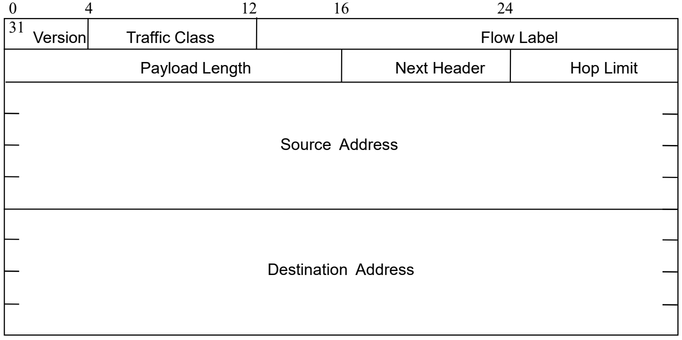
IPv6 Header Format

- Version field same size, same location as IPv4
- Traffic class to support differentiated services
- Flow: sequence of packets from particular source to particular destination for which source requires special handling



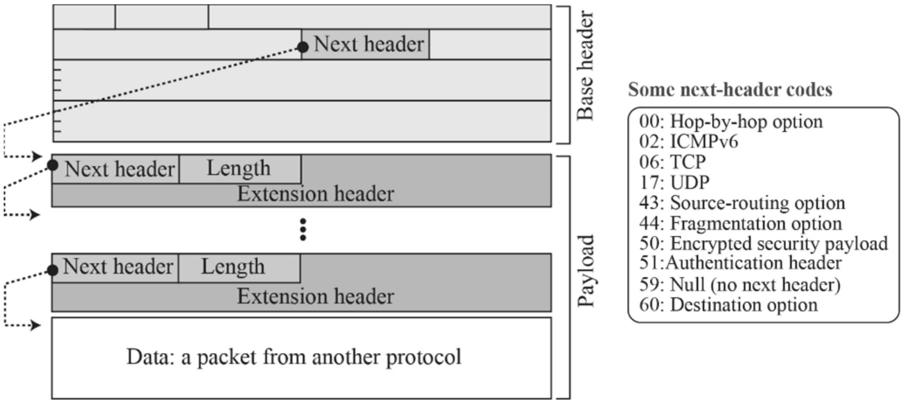
IPv6 Header Format

- Payload length: length of data excluding header, up to 65535 B
- Next header: type of extension header that follows basic header, e.g., TCP or UDP or options
- Hop limit: # hops packet can travel before being dropped by a router



Note: No CheckSum !!!

Figure 22.7: Payload in an IPv6 datagram



From: Data Communications and Networking 5th Edition by Behrouz A. Forouzan

---

IPv6 Addressing

- Address Categories
  - Unicast: single network interface
  - Multicast: group of network interfaces, typically at different locations. Packet sent to all.
  - Anycast: group of network interfaces. Packet sent to only one interface in group, e.g. nearest.
- Hexadecimal notation
  - Groups of 16 bits represented by 4 hex digits
  - Separated by colons
    - 4BF5:AA12:0216:FEBC:BA5F:039A:BE9A:2176
  - Shortened forms:
    - 4BF5:0000:0000:0000:BA5F:039A:000A:2176
    - To 4BF5:0:0:0:BA5F:39A:A:2176
    - To 4BF5::BA5F:39A:A:2176
  - Mixed notation:
    - ::FFFF:128.155.12.198
    - IPv4-mapped address, 0:0:0:0:0:FFFF:w.x.y.z or ::FFFF:w.x.y.z

From: Communication Networks:
Fundamentals Concepts and Key Architectures
Authors: A. Leon-Garcia and I. Widjaja

IPv6

- No checksum (assumes other layers take care of it)
  - Lowers router processing, no longer have to recompute header checksum at each hop since TTL decremented.
  - Relieves resource burden on very fast links
- No fragmentation in the network – source must perform PATH MTU discovery
  - Send ICMPv6 with requested MTU to destination, if get MTU to big response, decrement and retry. When destination replies, you have it.
  - Lowers router overhead – pushes complexity to edge

IPv6

- No broadcasts, replaced by multicasts
- ARPs, and ICMP combined/replaced with similar ICMPv6 functions.
- Security – IPsec available for v4, but is required to be available with IPv6 stack.
- Better support for mobility, auto configuration
  - No need for NAT, but NAT not going away
  - Hosts have multiple addresses, can dynamically reconfigure without impact→ easier plug-and-play
  - Router Solicitation, Router Advertisement – replaces DHCP, also includes duplicate address support
  - Enables stateless autoconfiguration → IPv6 address using a prefix obtained from a local router using an anycast message, eliminating the need for DHCP servers

Address Types based on Prefixes

Binary prefix	Types	Percentage of address space
0000 0000	Reserved	0.39
0000 0001	Unassigned	0.39
0000 001	ISO network addresses	0.78
0000 010	IPX network addresses	0.78
0000 011	Unassigned	0.78
0000 1	Unassigned	3.12
0001	Unassigned	6.25
001	Aggregatable global unicast addresses	12.5
010	Provider-based unicast addresses	12.5
011	Unassigned	12.5
100	Geographic-based unicast addresses	12.5
101	Unassigned	12.5
110	Unassigned	12.5
1110	Unassigned	6.25
1111 0	Unassigned	3.12
1111 10	Unassigned	1.56
1111 110	Unassigned	0.78
1111 1110 0	Unassigned	0.2
1111 1110 10	Link local use addresses	0.098
1111 1110 11	Site local use addresses	0.098
1111 1111	Multicast addresses	0.39

Aggregatable global unicast addresses

- Identified by the Format Prefix (FP) of 001
- Same as public IPv4 addresses.
- Globally routable and reachable on the IPv6 Internet.
- Aggregatable global unicast addresses are also known as global addresses.
- For more details see:

http://technet.microsoft.com/en-us/library/cc759208%28v=ws.10%29.aspx

Modified From: http://technet.microsoft.com/en-us/library/cc759208%28v=ws.10%29.aspx

Special Purpose Addresses

	n bits	m bits	o bits	p bits	(125-m-n-o-p) bits
010	Registry ID	Provider ID	Subscriber ID	Subnet ID	Interface ID

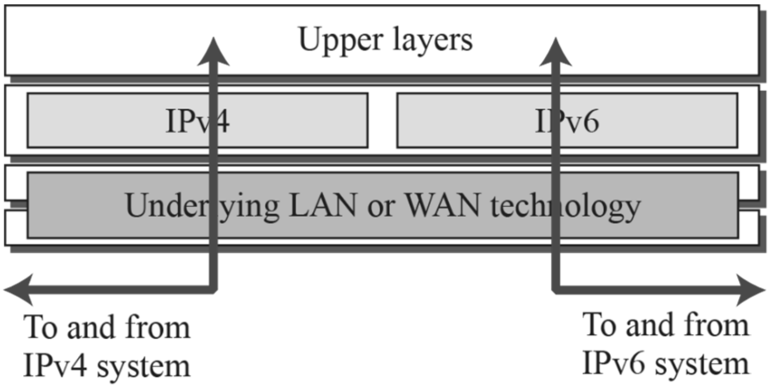
- *Provider-based Addresses*: 010 prefix
  - Assigned by providers to their customers
  - Hierarchical structure promotes aggregation
    - Registry ID: ARIN, RIPE, APNIC
    - ISP
    - Subscriber ID: subnet ID & interface ID
- IPv6 enables different hierarchical address structures to promote flexibility

---

Transition Mechanisms

- IPv6 Adoption https://www.akamai.com/visualizations/state-of-the-internet-report/ipv6-adoption-visualization
- Dual stacks
  - Network elements running IPv4 and IPv6 at the same time
  - With translation between protocols
  - Some routers already doing this.
- Tunneling

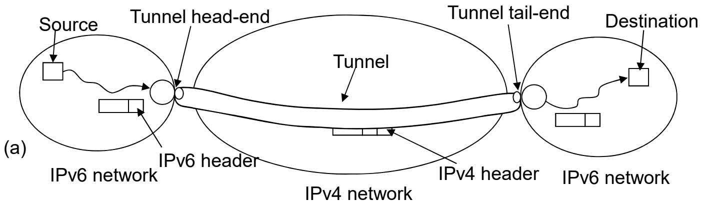
Dual Stack

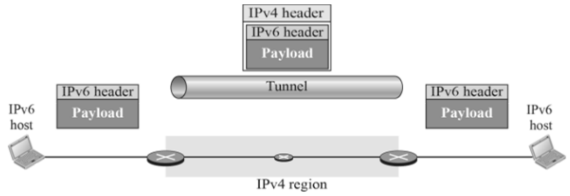


From: Data Communications and Networking 5th Edition by Behrouz A. Forouzan

---

Migration from IPv4 to IPv6
IPv6 over IPv4



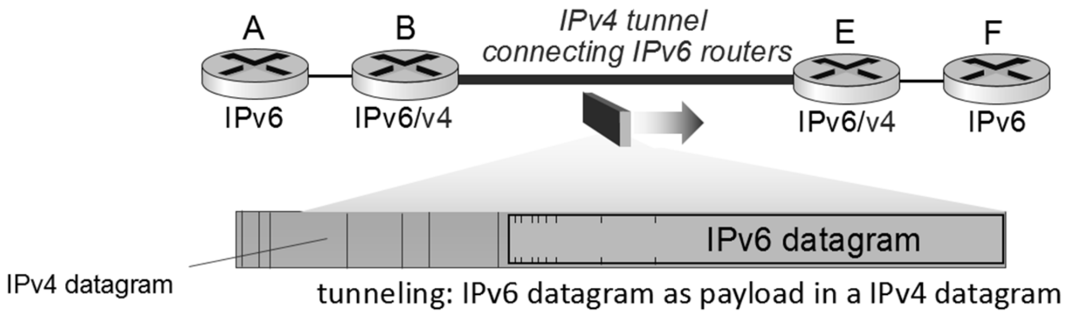


Modified From: Communication Networks: Fundamentals Concepts and Key Architectures Authors: A. Leon-Garcia and I. Widjaja

From: Data Communications and Networking 5th Edition by Behrouz A. Forouzan

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

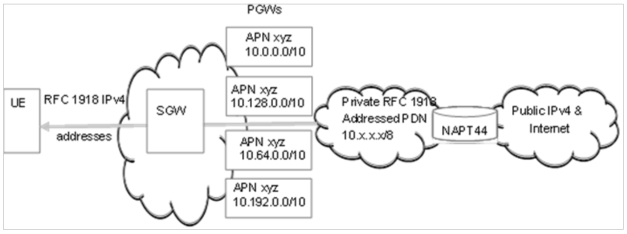
Migration from IPv4 to IPv6
IPv4 over IPv6



Modified from: 8th edition Jim
Kurose, Keith Ross Pearson, 2020

IP addressing in LTE (4G/5G)

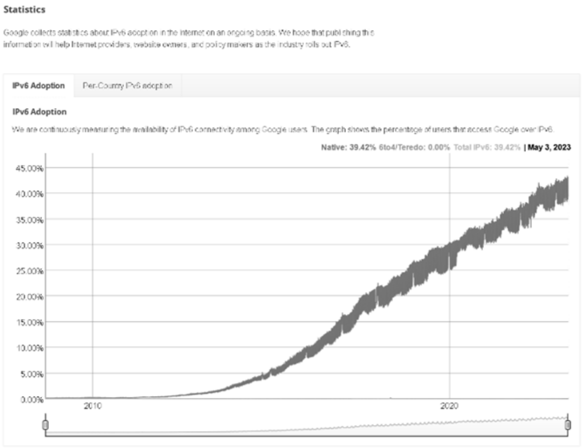
- Very large number of users, eventual transition to IPv6
- Scenario 1: Dual-stack (IPv6/IPv4) connectivity with Limited Public IPv4 Address Pools
- Scenario 2: Dual Stack (IPv6/IPv4) connectivity with Limited Private IPv4 Address Pools
- Scenario 3: UEs with IPv6-only connection and applications using IPv6
- Scenario 4: IPv4 applications running on a Dual-stack host with an assigned IPv6 prefix and a shared IPv4 address and having to access IPv4 services



LTE terminology:
UE= User Equipment, aka smartphone
NAPT = Network Address Port Translation-there is NAPT44 and NATP64, aka NAT
SGW = Serving Gateway
PGW = Packet Data Network Gateway
APN = Access Point Name

See: Digital cellular telecommunications system (Phase 2+) (GSM); Universal Mobile Telecommunications System (UMTS); LTE; IPv6 migration guidelines (3GPP TR 23.975 version 14.1.0 Release 14) , 2017

IPv6 deployment



Statistics

Google collects statistics about IPv6 adoption in the Internet on an ongoing basis. We hope that publishing this information will help Internet providers, website owners, and policy makers as the industry rolls out IPv6.

IPv6 Adoption

We are continuously measuring the availability of IPv6 connectivity among Google users. The graph shows the percentage of users that access Google over IPv6.

Native: 39.42% 6to4/Teredo: 0.00% Total IPv6: 39.42% | May 3, 2023

Go to https://www.google.com/intl/en/ipv6/statistics.html to get the latest data

---

IPv6

- For iOS, "Starting June 1, 2016, all apps submitted to the App Store must support IPv6-only networking. A majority of apps will not require any changes as IPv6 is already supported by NSURLSession and CFNetwork APIs. However, if your app utilizes IPv4-specific APIs or hard-coded IP addresses, you will need to make changes. Be sure to test for IPv6 compatibility before submitting your app to the App Store for review." From: https://developer.apple.com/support/ipv6/

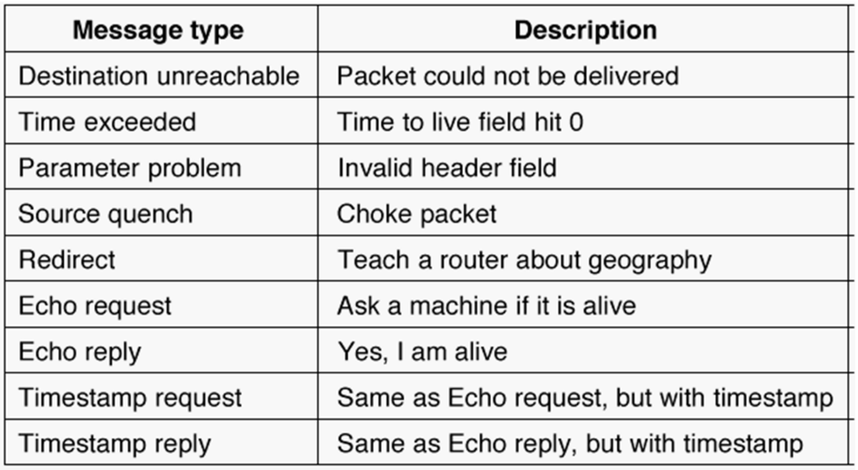
- Android uses dual-stack IPv4/IPv6.

- For more information on supporting IPv6 networks, review Supporting IPv6 DNS64/NAT64 Networks: https://developer.apple.com/library/ios/documentation/NetworkingInternetWeb/Conceptual/NetworkingOverview/UnderstandingandPreparingfortheIPv6Transition/UnderstandingandPreparingfortheIPv6Transition.html#//apple_ref/doc/uid/TP40010220-CH213-SW1

Internet Control Message Protocol: ICMP

- Purpose: Report unexpected events & test
- Used by hosts and routers to communicate network-level information
  - error reporting: unreachable host, network, port, protocol
  - echo request/reply (used by ping)
- network-layer "above" IP:
  - ICMP messages carried in IP datagrams
- *ICMP message:* type, code plus first 8 bytes of IP datagram causing error
- ICMPv6

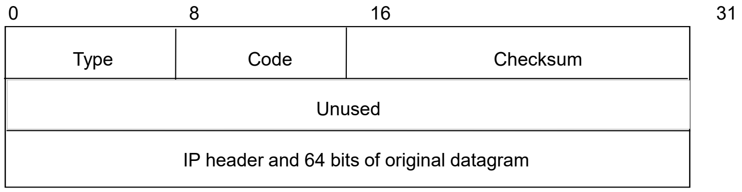
Principal ICMP message types



Message type	Description
Destination unreachable	Packet could not be delivered
Time exceeded	Time to live field hit 0
Parameter problem	Invalid header field
Source quench	Choke packet
Redirect	Teach a router about geography
Echo request	Ask a machine if it is alive
Echo reply	Yes, I am alive
Timestamp request	Same as Echo request, but with timestamp
Timestamp reply	Same as Echo reply, but with timestamp

From: "Computer Networks, 3rd Edition, A.S. Tanenbaum. Prentice Hall, 1996

ICMP Basic Error Message Format

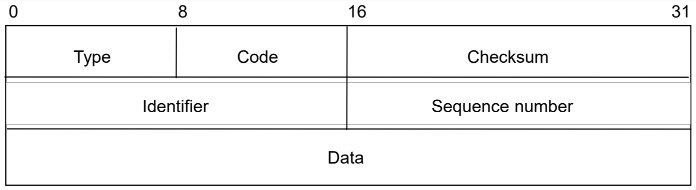


- *Type* of message: some examples
  - 0 Network Unreachable; 3 Port Unreachable
  - 1 Host Unreachable 4 Fragmentation needed
  - 2 Protocol Unreachable 5 Source route failed
  - 11 Time-exceeded, code=0 if TTL exceeded
- Code: purpose of message
- IP header & 64 bits of original datagram
  - To match ICMP message with original data in IP packet

From: Communication Networks:
Fundamentals Concepts and Key Architectures Authors: A. Leon Garcia and I. Widjaja

---

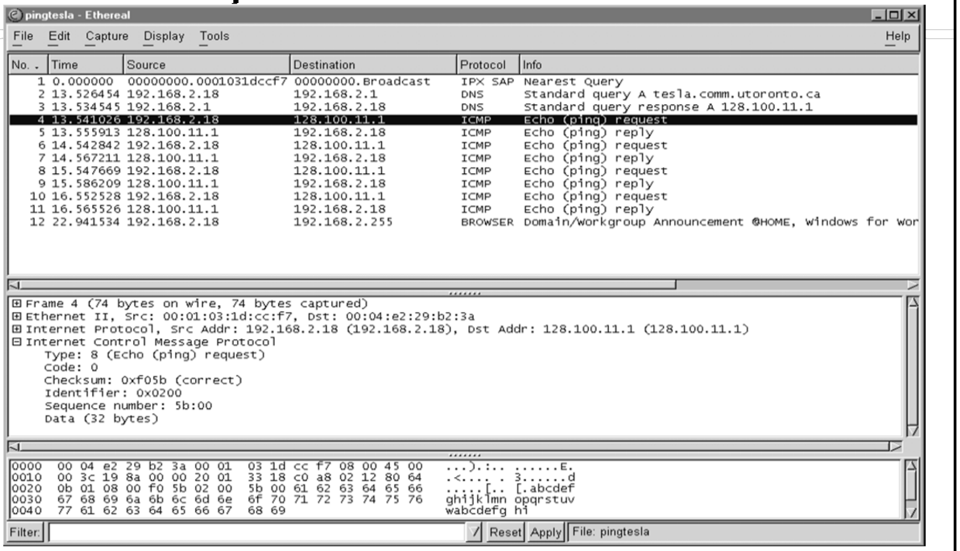
Echo Request & Echo Reply Message Format



- Echo request: type=8; Echo reply: type=0
  - Destination replies with echo reply by copying data in request onto reply message
- Sequence number to match reply to request
- ID to distinguish between different sessions using echo services
- Used in PING

From: Communication Networks:
Fundamentals Concepts and Key Architectures Authors: A. Leon-Garcia and I. Widjaja

Example – Echo request



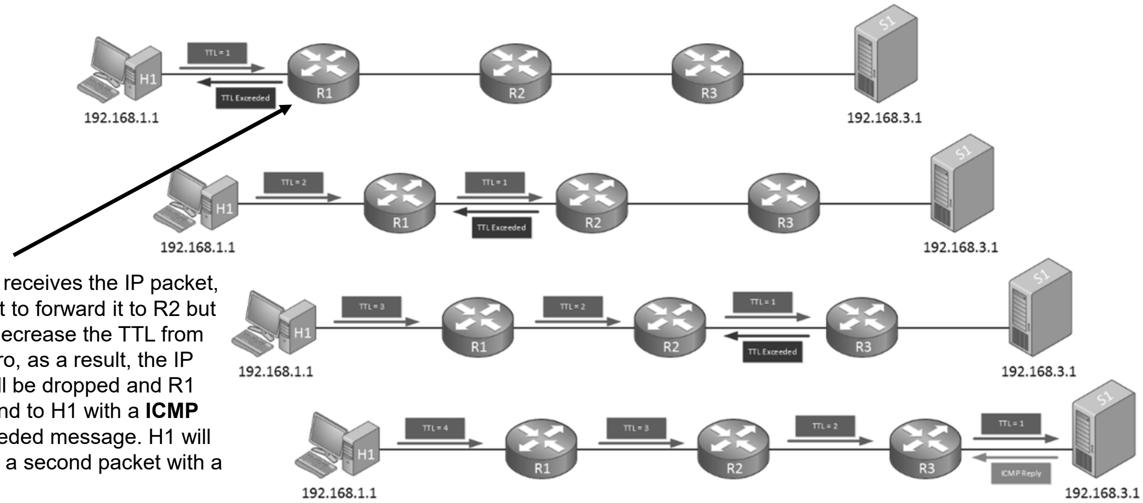
```
No. . Time      Source                 Destination         Protocol Info
    1 0.000000  00000000.0001031dccf7  00000000.Broadcast  IPX SAP  Nearest Query
    2 13.526454 192.168.2.18           192.168.2.1         DNS      Standard query A tesla.comm.utoronto.ca
    3 13.534545 192.168.2.1            192.168.2.18        DNS      Standard query response A 128.100.11.1
    4 13.541026 192.168.2.18           128.100.11.1        ICMP     Echo (ping) request
    5 13.555913 128.100.11.1           192.168.2.18        ICMP     Echo (ping) reply
    6 14.542842 192.168.2.18           128.100.11.1        ICMP     Echo (ping) request
    7 14.567211 128.100.11.1           192.168.2.18        ICMP     Echo (ping) reply
    8 15.547669 192.168.2.18           128.100.11.1        ICMP     Echo (ping) request
    9 15.586209 128.100.11.1           192.168.2.18        ICMP     Echo (ping) reply
   10 16.552528 192.168.2.18           128.100.11.1        ICMP     Echo (ping) request
   11 16.565526 128.100.11.1           192.168.2.18        ICMP     Echo (ping) reply
   12 22.941534 192.168.2.18           192.168.2.255       BROWSER  Domain/Workgroup Announcement @HOME, Windows for Wor
```

```
⊞ Frame 4 (74 bytes on wire, 74 bytes captured)
⊞ Ethernet II, Src: 00:01:03:1d:cc:f7, Dst: 00:04:e2:29:b2:3a
⊞ Internet Protocol, Src Addr: 192.168.2.18 (192.168.2.18), Dst Addr: 128.100.11.1 (128.100.11.1)
⊟ Internet Control Message Protocol
    Type: 8 (Echo (ping) request)
    Code: 0
    Checksum: 0xf05b (correct)
    Identifier: 0x0200
    Sequence number: 5b:00
    Data (32 bytes)
```

```
0000  00 04 e2 29 b2 3a 00 01  03 1d cc f7 08 00 45 00   ...).:.. ......E.
0010  00 3c 19 8a 00 00 20 01  33 18 c0 a8 02 12 80 64   .<.... . 3......d
0020  0b 01 08 00 f0 5b 02 00  5b 00 61 62 63 64 65 66   .....[.. [.abcdef
0030  67 68 69 6a 6b 6c 6d 6e  6f 70 71 72 73 74 75 76   ghijklmn opqrstuv
0040  77 61 62 63 64 65 66 67  68 69                     wabcdefg hi
```

From: Communication Networks: Fundamentals Concepts and Key Architectures
Authors: A. Leon-Garcia and I. Widjaja

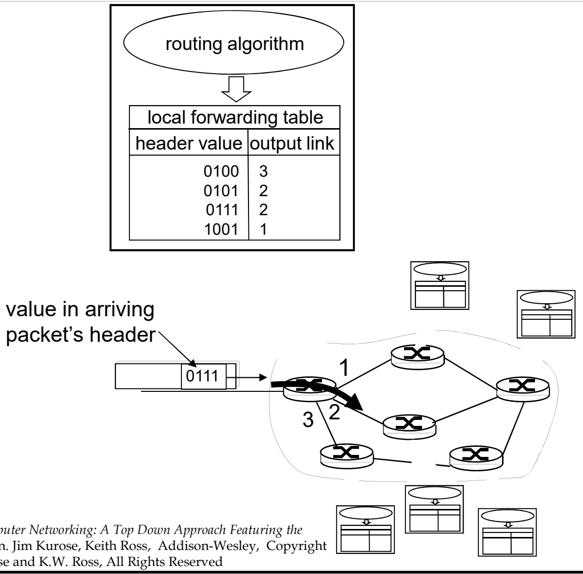
ICMP and Traceroute



When R1 receives the IP packet, it will want to forward it to R2 but it has to decrease the TTL from one to zero, as a result, the IP packet will be dropped and R1 will respond to H1 with a **ICMP** TTL exceeded message. H1 will now send a second packet with a TTL of 2:

Modified From:
https://networklessons.com/cisco/ccna-routing-switching-icnd1-100-105/traceroute

Routing vs. Forwarding



routing algorithm

local forwarding table	
header value	output link
0100	3
0101	2
0111	2
1001	1

value in arriving packet's header

Modified from *Computer Networking: A Top Down Approach Featuring the Internet*, 4nd edition. Jim Kurose, Keith Ross, Addison-Wesley, Copyright 1996-2002, J .F Kurose and K.W. Ross, All Rights Reserved

---

Routing vs. Forwarding

- Forwarding:
  Process of reading packet header, getting the destination address, looking up output hardware port in forwarding table and send packet on its way
- Routing: Process of building the forwarding table
- Forwarding is local

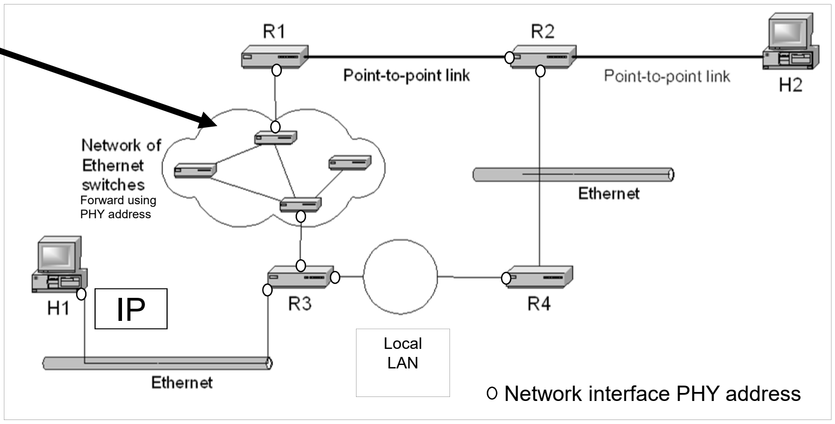
Routing vs. Forwarding

- Routing is
  - Distributed (routers communicate using a routing protocol)
  - "learns" the network topology
  - Finds "shortest" path
- Routing is like exploring,
  - Send explorers packets
  - They return with information of possible paths
  - Then calculate the best way to get from "here to there"

Forwarding: Delivery of an IP datagram

- View at the data link layer (the physical interconnections):
  - Internetwork is a collection of LANs or point-to-point links or switched networks that are connected by routers

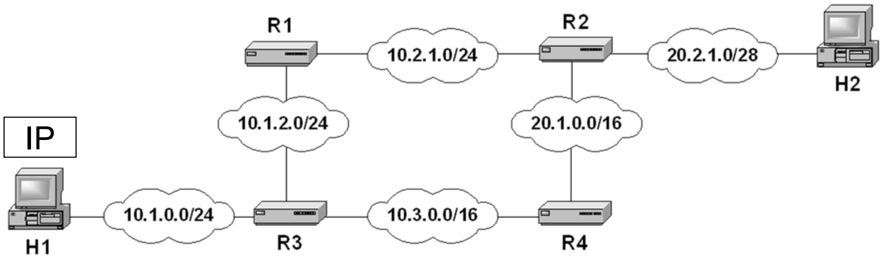
Ethernet Switches forward packets based on MAC (PHY) Address



Modified from: www.cs.virginia.edu/~itlab/book/slides/module09-ipforwV3.ppt

Forwarding: Delivery of an IP datagram

- View at the IP layer:
  - An IP network is a logical entity with a network number
  - We represent an IP network as a "cloud"
  - The IP delivery service takes the view of clouds, and ignores the data link layer view



Modified from: www.cs.virginia.edu/~itlab/book/slides/module09-ipforwV3.ppt

---

Tenets of end-to-end delivery of datagrams

The following conditions must hold so that an IP datagram can be successfully delivered

- The network prefix of an IP destination address must correspond to a unique data link layer network (=LAN or point-to-point link or switched network).
- Routers and hosts that have a common network prefix must be able to **directly** exchange IP datagrams using a data link protocol (e.g., broadcast, MAC, Ethernet, PPP)
- Every data link layer (Layer 2) network must be connected to at least one other data link layer network via a router.

Modified from: www.cs.virginia.edu/~itlab/book/slides/module09-ipforwV3.ppt

Forwarding Tables

- Each router and each host keeps a **forwarding table** which tells the router how to process an incoming packet
- Main columns:
    - **Destination address:** includes network where is the IP datagram going to?
    - **Next hop:** how to send the IP datagram?
    - **Interface:** what is the output hardware port?
- Next hop and interface column can often be summarized as one column
- Forwarding tables are set so that datagrams gets closer to the its destination

Modified from: www.cs.virginia.edu/~itlab/book/slides/module09-ipforwV3.ppt

---

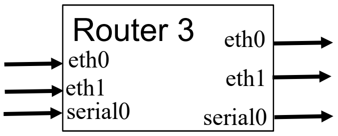
Forwarding Tables

Forwarding table of a host or router

IP datagrams can be directly delivered ("direct") or is sent to a router ("R4")

Will also include a default Next Hop, interface

Destination	Next Hop	interface
10.1.0.0/24	direct	eth0
10.1.2.0/24	direct	eth1
10.2.1.0/24	R4	serial0
10.3.1.0/24	direct	eth1
20.1.0.0/16	R4	serial0
20.2.1.0/28	R4	serial0



3 PHY Interfaces
eth0
eth1
serial0

Modified from: www.cs.virginia.edu/~itlab/book/slides/module09-ipforwV3.ppt

Forwarding Table Router 3

10.1.0.0/24

00001010 00000001 00000000 00000000
To
00001010 00000001 00000000 11111111

10.1.2.0/24

00001010 00000001 00000010 00000000
To
00001010 00000001 00000010 11111111

10.2.1.0/24

00001010 00000010 00000001 00000000
To
00001010 00000010 00000001 11111111

10.3.1.0/24

00001010 00000011 00000001 00000000
To
00001010 00000011 00000001 11111111

20.1.0.0/16

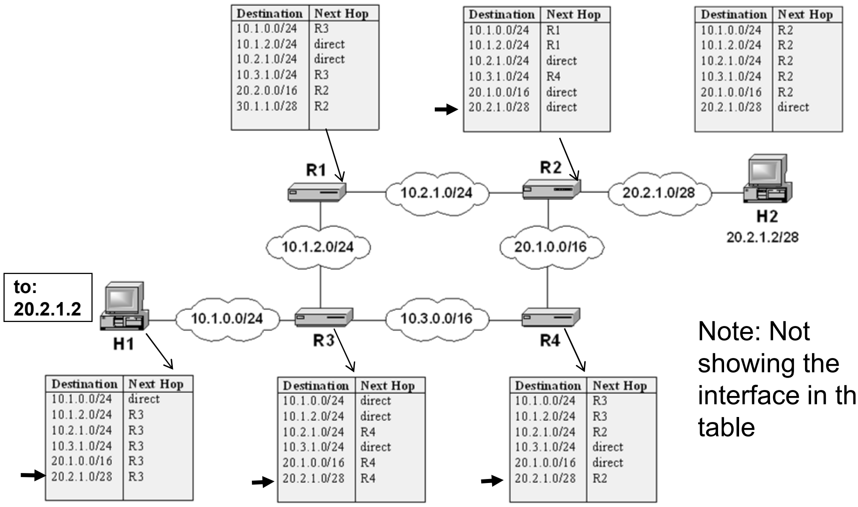
00010100 00000001 00000000 00000000
To
00010100 00000001 11111111 11111111

20.2.1.0/28

00010100 00000010 00000001 00000000
To
00010100 00000010 00000001 00001111

Remember that:
a) all 0's host ID reserved for the network
b) all 1's host ID reserved for broadcast

Delivery with forwarding tables



Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	direct
10.2.1.0/24	direct
10.3.1.0/24	R3
20.2.0.0/16	R2
30.1.1.0/28	R2

Destination	Next Hop
10.1.0.0/24	R1
10.1.2.0/24	R1
10.2.1.0/24	direct
10.3.1.0/24	R4
20.1.0.0/16	direct
20.2.1.0/28	direct

Destination	Next Hop
10.1.0.0/24	R2
10.1.2.0/24	R2
10.2.1.0/24	R2
10.3.1.0/24	R2
20.1.0.0/16	R2
20.2.1.0/28	direct

Destination	Next Hop
10.1.0.0/24	direct
10.1.2.0/24	R3
10.2.1.0/24	R3
10.3.1.0/24	R3
20.1.0.0/16	R3
20.2.1.0/28	R3

Destination	Next Hop
10.1.0.0/24	direct
10.1.2.0/24	direct
10.2.1.0/24	R4
10.3.1.0/24	direct
20.1.0.0/16	R4
20.2.1.0/28	R4

Destination	Next Hop
10.1.0.0/24	R3
10.1.2.0/24	R3
10.2.1.0/24	R2
10.3.1.0/24	direct
20.1.0.0/16	direct
20.2.1.0/28	R2

Note: Not showing the interface in the table

Modified from: www.cs.virginia.edu/~itlab/book/slides/module09-ipforwV3.ppt

Forwarding Tables – Router 3

IP address of arriving packet 20.2.1.2
00010100 00000010 00000001 00000010

Start with longest prefix known /28 (20.2.1.0/28)

Net mask 255.255.255.240
111111111 111111111 111111111 11110000

Logical AND incoming IP address with net mask
00010100 00000010 00000001 00000010
AND
111111111 111111111 111111111 11110000
=
00010100 00000010 00000001 00000000
20.2.1.0/28 is in the table so output Serial0 which is connected to R4

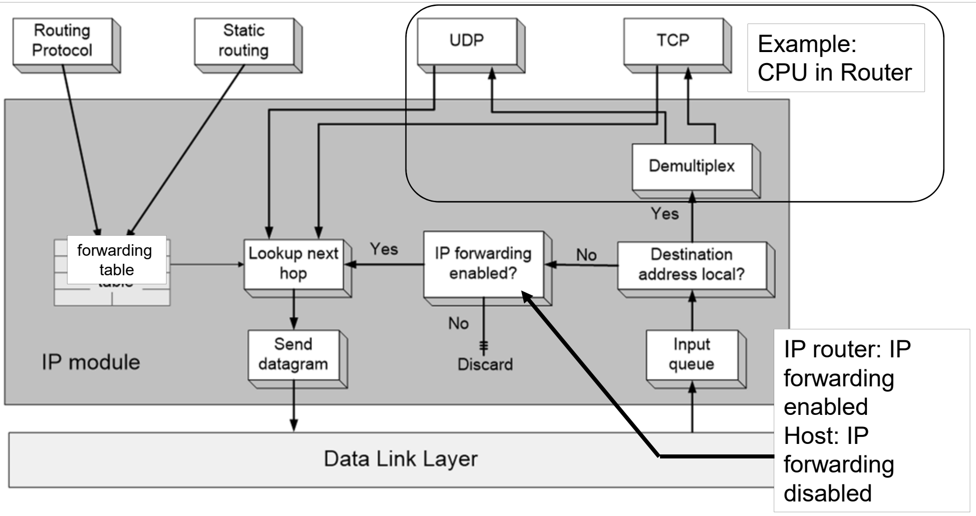
Delivery of IP datagrams

- There are two distinct processes to delivering IP datagrams:
  1. **Forwarding:** How to pass a packet from an input interface to the output interface?
  2. **Routing:** How to find and setup the forwarding tables?

- Forwarding must be done as fast as possible:
  - on routers, is often done with support of hardware
  - on PCs, is done in kernel of the operating system
- Routing is less time-critical
  - Filling in the forwarding table using learned information
  - On a PC, routing is done as a background process

Modified from: www.cs.virginia.edu/~itlab/book/slides/module09-ipforwV3.ppt

Processing of an IP datagram in IP (Host or Router)



Example:
CPU in Router

IP router: IP forwarding enabled
Host: IP forwarding disabled

Modified from: www.cs.virginia.edu/~itlab/book/slides/module09-ipforwV3.ppt

---

Processing of an IP datagram in IP

- Processing of IP datagrams is very similar on an IP router and a host
- Main difference:
  "IP forwarding" is enabled on router and disabled on host
- IP forwarding enabled
  → if a datagram is received, but it is not for the local system, the datagram will be sent to a different system
- IP forwarding disabled
  → if a datagram is received, but it is not for the local system, the datagram will be dropped
- The CPU in a router can be the local system, e.g., the destination for routing messages is the CPU in the router

Modified from: www.cs.virginia.edu/~itlab/book/slides/module09-ipforwV3.ppt

Processing of an IP datagram at a router

Receive an IP datagram →

1. IP header validation (Header checksum)
2. Process options in IP header
3. Parse the destination IP address
4. Forwarding table lookup
5. Decrement TTL
6. Perform fragmentation (if necessary)
7. Calculate checksum
8. Transmit to next hop
9. Send ICMP packet (if necessary)
    - If TTL = 0 after decrement then drop packet and send ICMP message

Modified from: www.cs.virginia.edu/~itlab/book/slides/module09-ipforwV3.ppt

---

Type of forwarding table entries

- **Network route**
    - Destination addresses is a network address (e.g., 10.0.2.0/24)
    - Most entries are network routes
- **Host route**
    - Destination address is an interface address (e.g., 10.0.1.2/32)
    - Used to specify a separate route for certain hosts
- **Default route**
    - Used when no network or host route matches
    - The router that is listed as the next hop of the default route is the **default gateway (for Cisco: "gateway of last resort")**
- **Loopback address**
    - Routing table for the loopback address (127.0.0.1)
    - The next hop lists the loopback (lo0) interface as outgoing interface

Modified from: www.cs.virginia.edu/~itlab/book/slides/module09-ipforwV3.ppt

Forwarding table lookup: Longest Prefix Match

Network Address/mask	Next Hop
10.0.0.0/8	R1
128.143.0.0/16	R2
128.143.64.0/20	R3
128.143.192.0/20	R3
128.143.71.0/24	R4
128.143.71.55/32	R3
default	R5

Forward table with IP & prefix defined with /n

	Leftmost bits in destination address-network prefix	Next Hop
shortest prefix	00001010 (/8)	R1
	10000000 10001111 (/16)	R2
	10000000 10001111 0100 (/20)	R3
	10000000 10001111 1100 (/20)	R3
	10000000 10001111 01000111 (/24)	R4
Longest prefix	10000000 10001111 01000111 00110111 (/32)	R3
	Default	R5

Forward table in bits with IP & prefix defined with /n

Forwarding table lookup: Longest Prefix Match

- **Longest Prefix Match:** Search for the forwarding table entry that has the longest match with the prefix of the destination IP address

1. Search for a match on all 32 bits
2. Search for a match for 24 bits
3. Search for a match for 20 bits
4. Search for a match for 16 bits
5. Search for a match for 8 bits
6. No match send out default -> R

Host route, loopback entry
→ 32-bit prefix match

Default route is represented as 0.0.0.0/0
→ 0-bit prefix match

How to forward → **128.143.71.21**

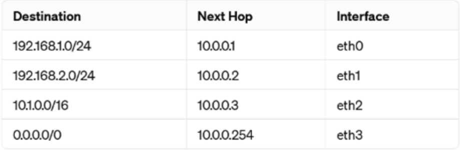
10000000 10001111 01000111 00010101
AND
11111111 11111111 11111111 00000000
=
10000000 10001111 01000111 – in table

	Leftmost bits in destination address-network prefix	Next Hop
shortest prefix	00001010 (/8)	R1
	10000000 10001111 (/16)	R2
	10000000 10001111 0100 (/20)	R3
	10000000 10001111 1100 (/20)	R3
	10000000 10001111 01000111 (/24)	R4
Longest prefix	10000000 10001111 01000111 00110111 (/32)	R3
	Default	R5

The longest prefix match for 128.143.71.21 is for 24 bits with entry 128.143.71.0/24
Datagram will be sent to R4

Modified from: www.cs.virginia.edu/~itlab/book/slides/module09-ipforwV3.ppt

Forwarding table lookup: Longest Prefix Match



Destination	Next Hop	Interface
192.168.1.0/24	10.0.0.1	eth0
192.168.2.0/24	10.0.0.2	eth1
10.1.0.0/16	10.0.0.3	eth2
0.0.0.0/0	10.0.0.254	eth3

The first entry indicates that any packet destined for the IP addresses in the range of 192.168.1.0 to 192.168.1.255 (subnet mask /24) should be forwarded to the next hop address 10.0.0.1 via interface eth0.

The second entry specifies that packets destined for the IP addresses in the range of 192.168.2.0 to 192.168.2.255 should be forwarded to the next hop address 10.0.0.2 via interface eth1.

The third entry indicates that packets destined for the IP addresses in the range of 10.1.0.0 to 10.1.255.255 should be forwarded to the next hop address 10.0.0.3 via interface eth2.

The last entry with destination 0.0.0.0/0 serves as a default route, meaning any packet that doesn't match any specific entry in the forwarding table should be forwarded to the next hop address 10.0.0.254 via interface eth3.

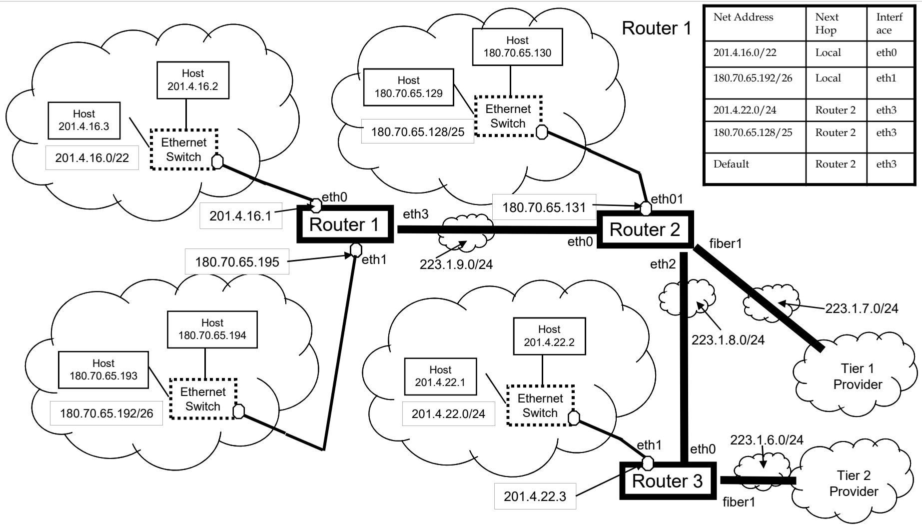
Example: Packet arrives with the destination IP address 192.168.1.100. To determine the appropriate next hop for forwarding, the router performs longest prefix matching:

It compares the destination IP address (192.168.1.100) with the entries in the forwarding table.

The router finds that the destination IP address matches the first entry (192.168.1.0/24) in the forwarding table.

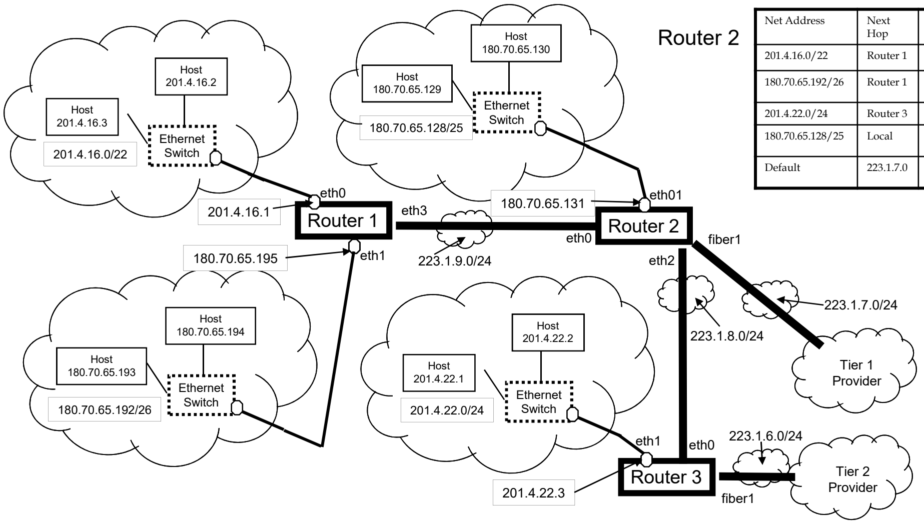
Since this entry has the longest matching prefix (/24), the router selects the corresponding next hop (10.0.0.1) and forwards the packet via the specified interface (eth0).

Example: IP Forwarding



Net Address	Next Hop	Interface
201.4.16.0/22	Local	eth0
180.70.65.192/26	Local	eth1
201.4.22.0/24	Router 2	eth3
180.70.65.128/25	Router 2	eth3
Default	Router 2	eth3

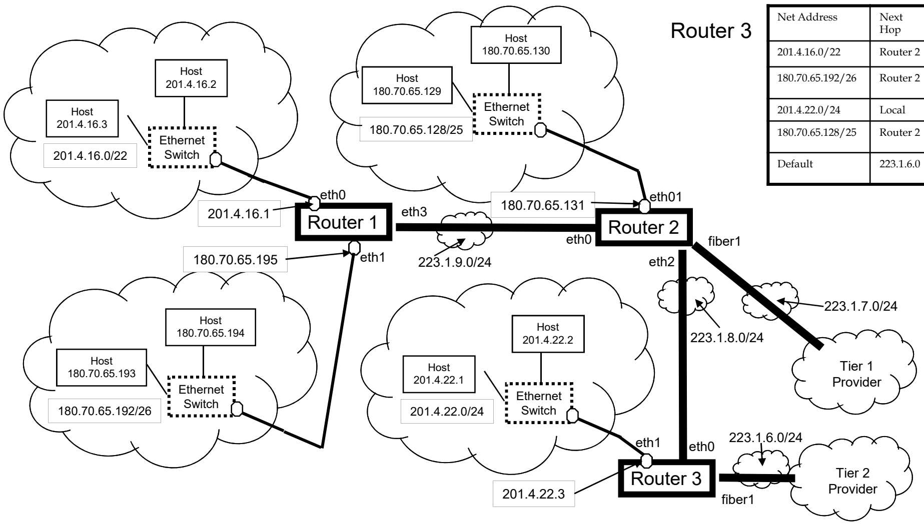
Example: IP Forwarding



Router 2

Net Address	Next Hop	Interface
201.4.16.0/22	Router 1	eth0
180.70.65.192/26	Router 1	eth0
201.4.22.0/24	Router 3	eth2
180.70.65.128/25	Local	eth1
Default	223.1.7.0	fiber1

Example: IP Forwarding

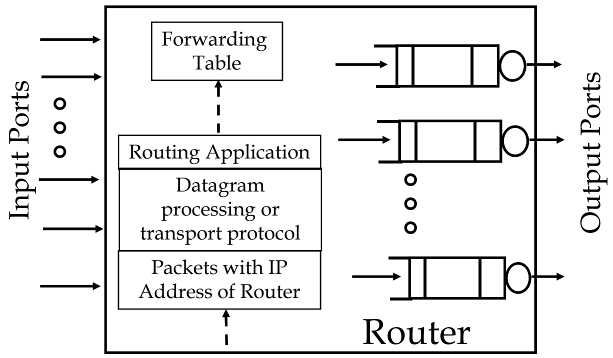


Router 3

Net Address	Next Hop	Interface
201.4.16.0/22	Router 2	eth0
180.70.65.192/26	Router 2	eth0
201.4.22.0/24	Local	eth1
180.70.65.128/25	Router 2	eth0
Default	223.1.6.0	fiber1

IP Routing

- View routing as an application running on a router's CPU communicating over IP or with or w/o a transport protocol, UDP or TCP



Making routing scalable

our routing study thus far - idealized

- all routers identical
- network "flat"

… not true in practice

scale: billions of destinations:

- can't store all destinations in routing tables!
- routing table exchange would swamp links!

administrative autonomy:

- Internet: a network of networks
- each network admin may want to control routing in its own network

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Internet approach to scalable routing

aggregate routers into regions known as “autonomous systems” (AS) (a.k.a. “domains”)

intra-AS (aka “intra-domain”): routing among *within same AS (“network”)*

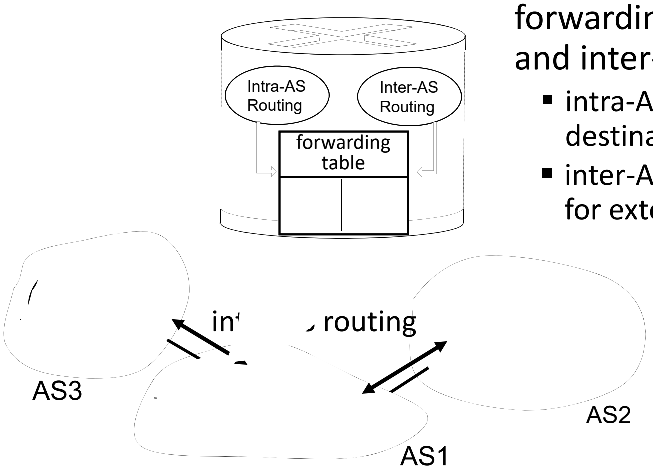
- all routers in AS must run same intra-domain protocol
- Interior Gateway Router (IGP) Protocol
- routers in different AS can run different intra-domain routing protocols (IGPs)
- gateway router: at “edge” of its own AS, has link(s) to router(s) in other AS’es

inter-AS (aka “inter-domain”): routing *among* AS’es

- gateways perform inter-domain routing (as well as intra-domain routing)
- Exterior Gateway Routing (EGP) Protocol

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Interconnected ASes



forwarding table configured by intra- and inter-AS routing algorithms

- intra-AS routing determine entries for destinations within AS
- inter-AS & intra-AS determine entries for external destinations

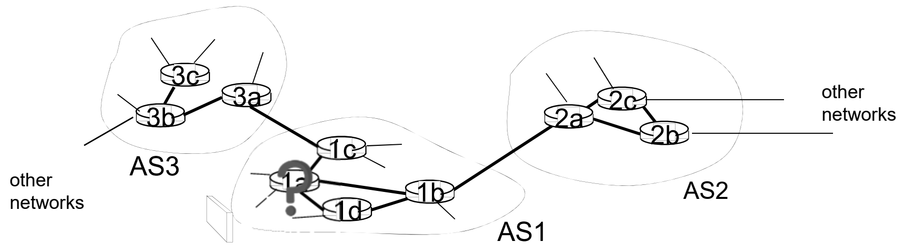
Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Inter-AS routing: a role in intradomain forwarding

- suppose router in AS1 receives datagram destined outside of AS1:
  - router should forward packet to gateway router in AS1, but which one?

AS1 inter-domain routing must:

1. learn which destinations reachable through AS2, which through AS3
2. propagate this reachability info to all routers in AS1



Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Autonomous Systems (AS)

- Global Internet viewed as collection of autonomous systems.
- **Autonomous system (AS)** is a set of routers or networks administered by a single organization
- Same routing protocol need not be run within an AS and between ASs
- But, to the outside world, an AS should present a *consistent picture of what ASs are reachable* through it
- **Stub AS:** has only a single connection to the outside world.
- **Multihomed AS:** has multiple connections to the outside world, but refuses to carry transit traffic
- **Transit AS:** has multiple connections to the outside world, and can carry transit and local traffic.

From: Communication Networks: Fundamentals Concepts and Key Architectures
Authors: A. Leon-Garcia and I. Widjaja

AS Numbers (ASN)

- In RFC 4893 AS numbers are 32 bits (AS #'s are not IP addresses)
- KU is an AS with AS # 2496
- Internet Assigned Numbers Authority (IANA) gives ASNs to regional internet registry (RIR), RIRs give ASNs to, ISPs and end-user organizations.
- RIRs
  - African Network Information Centre (AfriNIC) for Africa
  - American Registry for Internet Numbers (ARIN) for the United States, Canada, and several parts of the Caribbean region.
  - Asia-Pacific Network Information Centre (APNIC) for Asia, Australia, New Zealand, and neighboring countries
  - Latin America and Caribbean Network Information Centre (LACNIC) for Latin America and parts of the Caribbean region
  - Réseaux IP Européens Network Coordination Centre (RIPE) for Europe, the Middle East, and Central Asia

---

Routing

- Routing protocols are used to *"load"* the forwarding tables in IP routers
- Routing protocols "learn about the *"state of the network"* and communicate routing information between routers
- Routing protocols implement part of the IP, signaling for IP or the *"Control Plane"*

Internet Routing Protocols

- Interior Gateway Router (IGP) Protocol
  - Routing protocol within “autonomous” systems, e.g., KU
    - Open Shortest Path First (OSPF)
    - Router Information Protocol (RIP)
  - An AS is usually own/controlled by one organization, e.g., an ISP
- Exterior Gateway Routing (EGP) Protocol
  - Routing between “autonomous” systems
    - Border Gateway Protocol (BGP)
  - EGPs must work **BETWEEN** organizations, e.g., Level-3 and ATT
- As of March 2021 there were over 100,000 AS’s.

Routing Protocols: Issues

- Coordinate a path (route)
- Route discovery
  - What does the network look like → topology?
  - What routes are available?
- What information needs to be shared?
  - What are the characteristics of the paths, e.g., capacity, delay, loss, jitter, etc.
- How is network state information shared, e.g., flooding?
  - Flooding=send packet out all ports
- How is network state information used?

The Routing Problem

- Routing algorithms attempt to build forwarding tables to "optimally" route traffic based on some knowledge of the network topology and state (e.g., link delay and loss)
- Practical problems:
  - Which shortest path algorithm to use?
  - How to learn the topology and network state, e.g., congested routes?
  - How define an optimization metric (length or "distance")?
    - The bubble, change paths to reduce delays for some traffic may worsen performance for other traffic.
  - How to respond to:
    - Network element failures
    - Link failures
    - Changes in traffic, e.g., congestion
  - How to establish policies between AS's?
- Different routing protocols answer these questions in different ways.

Routing-Shortest Path Algorithm

- What is distance (link weight)?
  - Propagation delay ∝ Physical distance,
    e.g., terrestrial vs. satellite link
  - Number of hops, i.e., number of routers the packet hits between the source and destination
  - Other *"cost"*
    - Cost in $
    - Cost in *"congestion"*, least congested
    - Available capacity
    - Path propagation delay
    - Administratively set

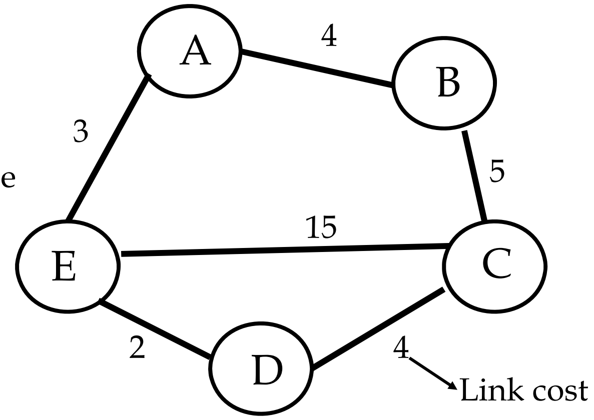
Routing-Shortest Path Algorithm

- Shortest Path Algorithm finds the minimum “distance” path between nodes
- Input
  - Topology
  - Link *“distances” (link weights)*
- Output is a forwarding table

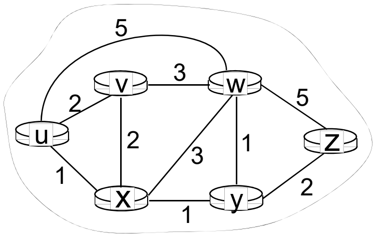
Routing-Shortest Path Algorithm

Example:

Find the shortest path routing table for all nodes



Graph abstraction: link costs



$c_{a,b}$: cost of *direct* link connecting *a* and *b*

e.g., $c_{w,z} = 5$, $c_{u,z} = \infty$

cost defined by network operator: could always be 1, or inversely related to bandwidth, or inversely related to congestion

graph: $G = (N,E)$

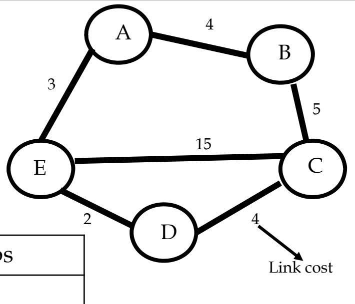
N: set of routers = { *u, v, w, x, y, z* }

E: set of links ={ *(u,v), (u,x), (v,x), (v,w), (x,w), (x,y), (w,y), (w,z), (y,z)* }

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Exhaustive Search

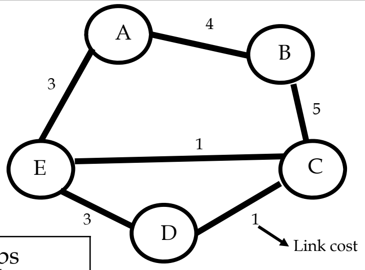
- Find Shortest path from A to D
- List all possible paths and their lengths



Path	Length	# hops
A→B→C→D	13	3
A→E→D	5	2
A→E→C→D	22	3
A→B→C→E→D	26	4

Exhaustive Search

- New link weights
- Find Shortest path from A to D
- List all possible paths and their lengths

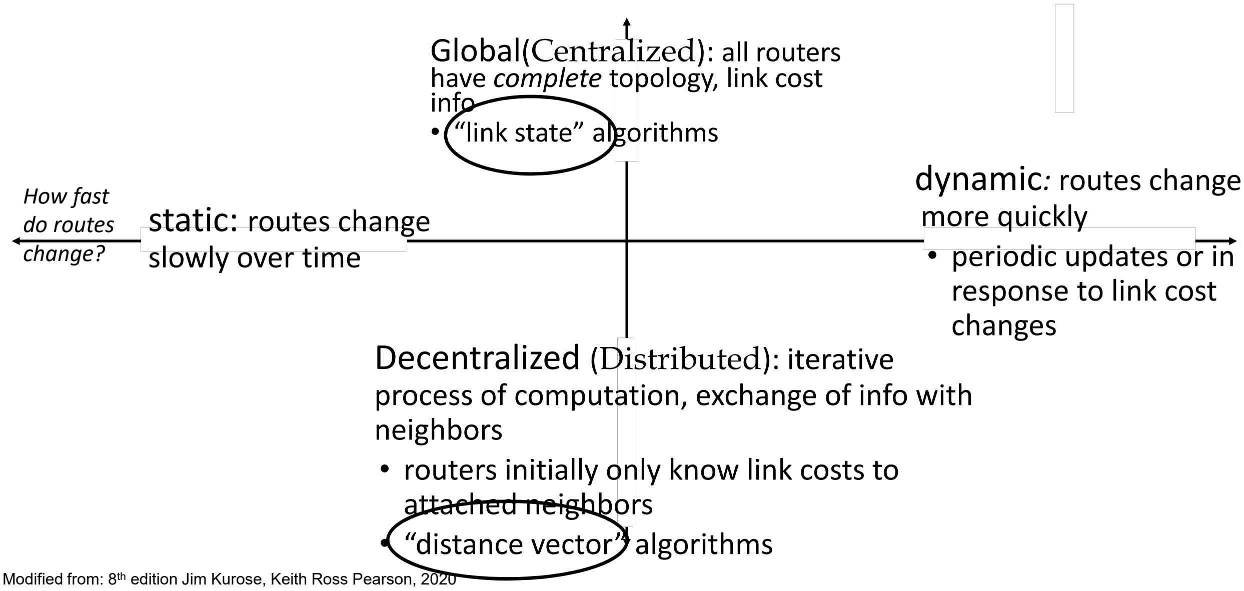


Path	Length	# hops
A→B→C→D	10	3
A→E→D	6	2
A→E→C→D	5	3
A→B→C→E→D	13	4

Routing Algorithms

- Exhaustive Search does not scale with the size of the network
- Routing is a “top-10” networking challenge
- Optimum shortest path algorithms exist to efficiently find the shortest path
- Routing Algorithms
  - Centralized
  - Distributed
- Examples:
  - Bellman-Ford Algorithm (one source/destination pair at a time)
  - Dijkstra’s Algorithm (source to all destinations)

Routing algorithm classification



Global(Centralized): all routers have *complete* topology, link cost info
- "link state" algorithms

How fast do routes change?

static: routes change slowly over time

dynamic*:* routes change more quickly
- periodic updates or in response to link cost changes

Decentralized (Distributed): iterative process of computation, exchange of info with neighbors
- routers initially only know link costs to attached neighbors
- "distance vector" algorithms

Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Shortest Path Approaches

Distance Vector Protocols
- Neighbors exchange list of distances to destinations
- Best next-hop determined for each destination
- Bellman-Ford (distributed) shortest path algorithm

Link State Protocols
- Link state information flooded to all routers
- Routers have complete topology information
- Shortest path (& hence next hop) calculated
- Dijkstra (centralized) shortest path algorithm
  - Show example of Dijkstra's Algorithm

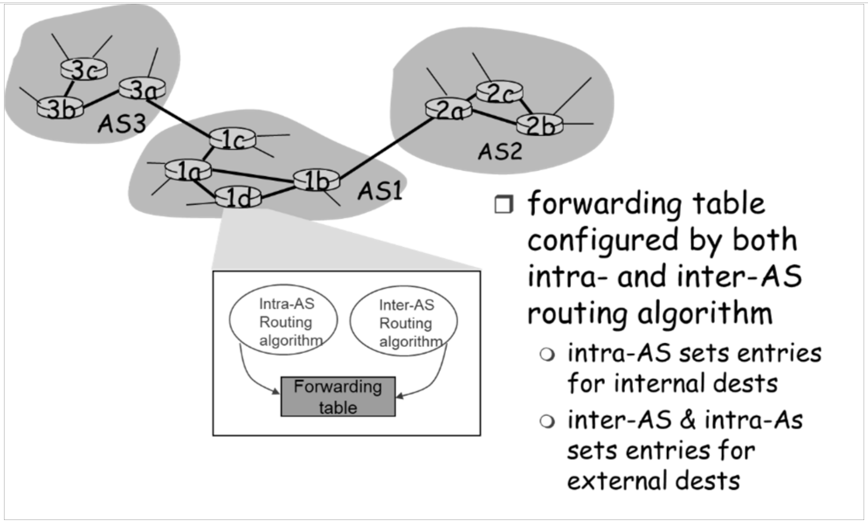
http://demonstrations.wolfram.com/ShortestPathsAndTheMinimumSpanningTreeOnAGraphWithCartesianE/

From: Communication Networks:
Fundamentals Concepts and Key Architectures
Authors: A. Leon-Garcia and I. Widjaja

More on IP Routing:
Open Shortest Path First (OSPF)

- IGP (within one AS)
- Link State routing protocol
- Routers discover
  - Their neighbors
  - The state of incident links
- Communicate state by periodically flooding the Link State Advertisements (LSA) throughout the network
- All routers converge to same map of the network topology
- Shortest path algorithm then used for routing. Distance can be more that just hop count.
- Carried directly by IP

More on IP Routing:
EGPs (Between AS's)



- forwarding table configured by both intra- and inter-AS routing algorithm
  - intra-AS sets entries for internal dests
  - inter-AS & intra-As sets entries for external dests

Modified from *Computer Networking: A Top Down Approach Featuring the Internet*, 4nd edition. Jim Kurose, Keith Ross, Addison-Wesley, Copyright 1996-2002, J .F Kurose and K.W. Ross, All Rights Reserved

More on IP Routing: Border Gateway Protocol (BGP)

- BGP (Border Gateway Protocol): *the* de facto standard
- BGP provides each AS a means to:
  1. Obtain subnet reachability information from neighboring ASs.
  2. Propagate reachability information to all AS-internal routers.
  3. Determine “good” routes to subnets based on reachability information and policy.
- allows subnet to advertise its existence to rest of Internet: *“I am here, here is who I can reach, and how”*

Modified from *Computer Networking: A Top Down Approach Featuring the Internet*, 4nd edition. Jim Kurose, Keith Ross, Addison-Wesley, Copyright 1996-2002, J .F Kurose and K.W. Ross, All Rights Reserved

More on IP Routing: Border Gateway Protocol (BGP)

- EGP (Between AS’s)
- Finds paths for source/destinations pairs that span multiple AS’s.
- Path vector protocol, BGP advertises a sequence of AS #’s to the destination
- Routing information includes complete list of networks (AS’s) between source and destination
- Path vector info used to prevent routing loops
- Allows ranking of routes based on polices
- Polices are arbitrary rules, e.g., based on business agreements
- BGP enforces policy through selection of different paths to a destination and by control of redistribution of routing information
- Currently, it is common to have these manually configured
- Business agreements are reflected in BGP policies
- BGP uses TCP as the transport protocol

Why different Intra-, Inter-AS routing ?

Policy:

- inter-AS (IGP): admin wants control over how its traffic routed, who routes through its network
- intra-AS (EGP): single admin, so policy less of an issue

Scale:

- hierarchical routing saves table size, reduced update traffic

Performance:

- intra-AS (IGP): can focus on performance
- inter-AS (EGP): policy dominates over performance

Security:

- intra-AS (IGP): OSPF messages are authenticated
- inter-AS (EGP):
  - BGP authenticates the identity of their peers
  - BGPSEC provides a mechanism for verifying the authenticity and integrity of BGP route updates

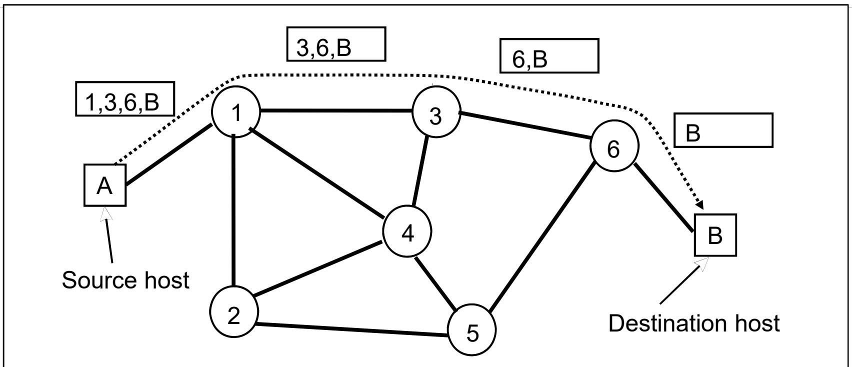
Modified from: 8th edition Jim Kurose, Keith Ross Pearson, 2020

Source Routing

- Source host selects path that is to be followed by a packet: sequence of nodes in path inserted into header
- Intermediate switches read next-hop address and remove address
- Source host needs link state information or access to a route server
- Source routing allows the host to control the paths that its information traverses in the network
- Potentially the means for customers to select what service providers they use

From: Communication Networks:
Fundamentals Concepts and Key Architectures
Authors: A. Leon-Garcia and I. Widjaja

Example of source routing



Does source routing scale?

Both IPv4 and IPv6 allow source routing

Modified From: Communication Networks:
Fundamentals Concepts and Key Architectures
Authors: A. Leon-Garcia and I. Widjaja