Network Layer

October 18, 2021

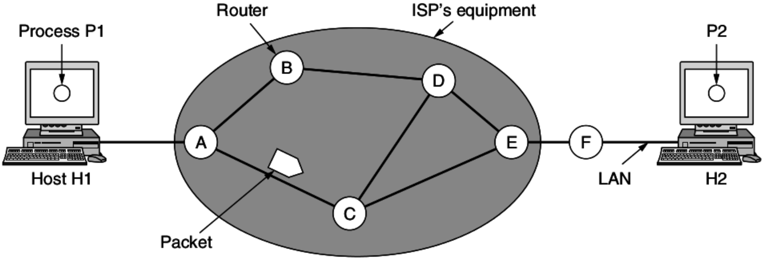
Goals

- Getting packets from the source all the way to the destination.
- Getting to the destination may require making many hops at intermediate routers along the way.

How to achieve this?

- To achieve its goals, N/W layer must know about the topology of the network (i.e., the set of all routers and links) and choose appropriate paths through it.
- Routes chosen should avoid overloading some of the communication lines and routers while leaving others idle.

Store and Forward Packet Switching



- A host with a packet to send transmits it to the nearest router, either on its own LAN or over a point-to-point link to the ISP.
- The packet is stored there until it has fully arrived and the link has finished its processing by verifying the checksum.
- Then it is forwarded to the next router along the path until it reaches the destination host, where it is delivered.

Services provided to Transport Layer

- The network layer provides services to the transport layer at the network layer/transport layer interface.
- Question/Dilemma: Should N/W layer provide connection oriented service or connectionless service.

Connection less service

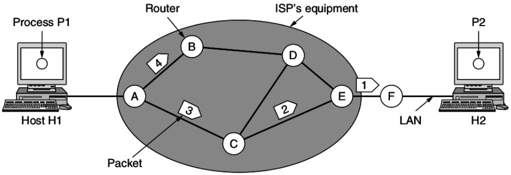
- Argument: Routers' job is moving packets around and nothing else.
- The network is inherently unreliable, no matter how it is designed. Therefore, the hosts should accept this fact and do error control (i.e., error detection and correction) and flow control themselves.
- Should comprise SEND PACKET and RECEIVE PACKET primitives, and each packet must carry the full destination address, because each packet sent is carried independently of its predecessors, if any.

Connection oriented service

- Argument: Success of telephone network. Reliable and provides QoS.
- X.25, Frame Relay (CO): ARPANET, Internet (CL).

Implementation of Connection Less Service

- Packets are injected into the network individually and routed independently of each other. No advance setup is needed.
- Packets are frequently called datagrams (in analogy with telegrams) and the network is called a datagram network.
- The algorithm that manages the tables and makes the routing decisions is called the routing algorithm.

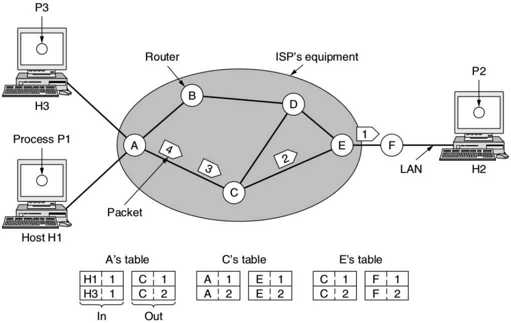


Dest.	A's table (initially)	A's table (later)	C's table	E's table
A	–	–	A	C
B	B	B	A	D
C	C	C	–	C
D	B	B	E	D
E	C	B	E	–
F	C	B	E	F

Dest. Line

Implementation of Connection Oriented Service

- Path (VC) from the source router to destination router must be established before any data packets can be sent. Each packet carries an identifier telling which virtual circuit it belongs to.
- *A* assigns a different connection identifier to the outgoing traffic for the second connection. Avoiding conflicts of this kind is why routers need the ability to replace connection identifiers in outgoing packets.



A's table

In		Out	
H1	1	C	1
H3	1	C	2

C's table

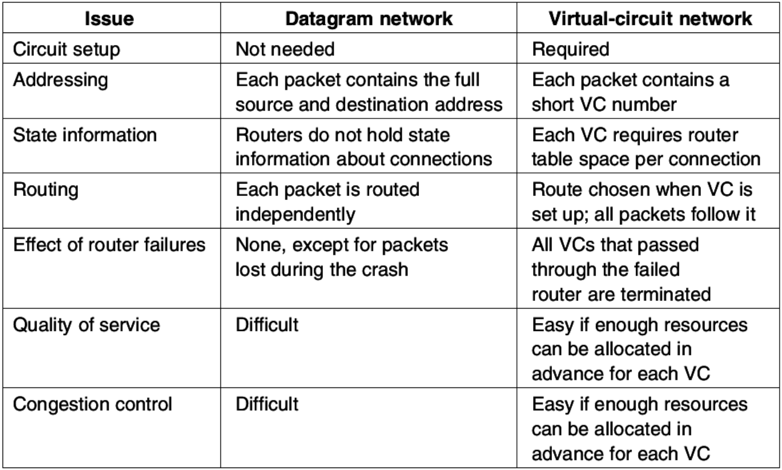
A	1	E	1
A	2	E	2

E's table

C	1	F	1
C	2	F	2

VC & Datagram Networks

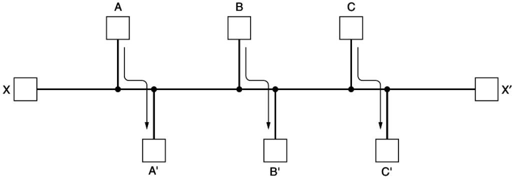
- Resources (e.g., buffers, band-width, and CPU cycles) can be reserved in advance, when the connection is established.



Issue	Datagram network	Virtual-circuit network
Circuit setup	Not needed	Required
Addressing	Each packet contains the full source and destination address	Each packet contains a short VC number
State information	Routers do not hold state information about connections	Each VC requires router table space per connection
Routing	Each packet is routed independently	Route chosen when VC is set up; all packets follow it
Effect of router failures	None, except for packets lost during the crash	All VCs that passed through the failed router are terminated
Quality of service	Difficult	Easy if enough resources can be allocated in advance for each VC
Congestion control	Difficult	Easy if enough resources can be allocated in advance for each VC

Routing Algorithms

- Routing algorithm is that part of the network layer software responsible for deciding which output line an incoming packet should be transmitted on.
- A router can be thought of a two processes inside it:
  1. Handles each packet as it arrives, **looking up the outgoing line to use** for it in the routing tables. **This process is forwarding.**
  2. Other process is responsible for filling in and updating the routing tables.
- Desirable properties of a routing algorithm: correctness, simplicity, robustness, stability, fairness, and efficiency.



Classes of Routing Algorithms

- **Non-adaptive:** Routes are computed in advance, offline, and downloaded to the routers when the network is booted(static routing).
- **Adaptive algorithms:** Change their routing decisions to reflect changes in the topology, and sometimes changes in the traffic as well.

Dynamic routing algorithms differ in following:

1. where they get their information
2. when they change the routes, and
3. what metric is used for optimization

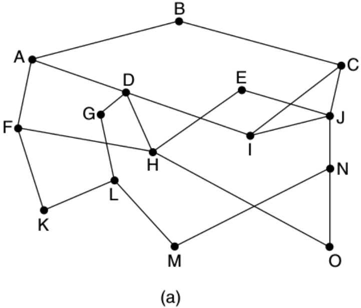
Optimality Principle

- It states that if router J is on the optimal path from router I to router K, then the optimal path from J to K also falls along the same route.

Optimality Principle

- It states that if router J is on the optimal path from router I to router K, then the optimal path from J to K also falls along the same route.
- To see this, call the part of the route from I to J r_1 and the rest of the route r_2.
- If a route better than r_2 existed from J to K, it could be concatenated with r_1 to improve the route from I to K, contradicting our statement that r_1 r_2 is optimal.

Optimality Principle



- Based on optimality principle, the set of optimal routes from all sources to a given destination form a tree rooted at the destination **(sink tree)** where the distance metric is the number of hops.
- The **goal of all routing algorithms** is to discover and use the sink trees for all routers.

Optimality Principle

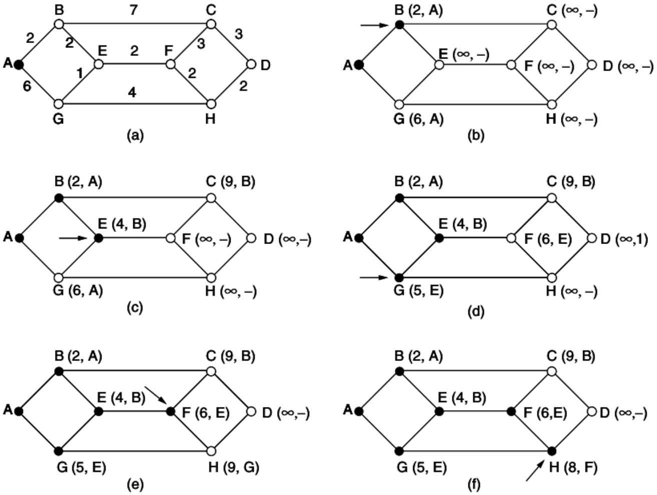
- A sink tree is not necessarily unique; other trees with the same path lengths may exist.
- Since a sink tree is indeed a tree, it does not contain any loops, so each packet will be delivered within a finite and bounded number of hops.

Shortest Path Algorithm

Lets us compute optimal paths given a complete picture of the network.

- Dijkstra's algorithm (1959) finds the shortest paths between a source and all destinations in the network. Each node is labelled with its distance from the source node along the best known path.
- Initially, no paths are known, so all nodes are labelled with infinity.
- As the algorithm proceeds and paths are found, the labels may change, reflecting better paths.
- A label may be either tentative or permanent. Initially, all labels are tentative.
- When it is discovered that a label represents the shortest possible path from the source to that node, it is made permanent and never changed thereafter.

Shortest Path Algorithm



Flooding

When a routing algorithm is implemented, each router must make decisions based on local knowledge, not the complete picture of the network.

- Flooding is the technique in which every incoming packet is sent out on every outgoing line except the one it arrived on.
- Flooding generates vast numbers of duplicate packets, in fact, an infinite number that can be limited by using hop-counter.
- A better technique for damming the flood is to have routers keep track of which packets have been flooded, to avoid sending them out a second time.

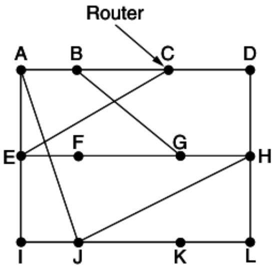
Distance Vector Routing

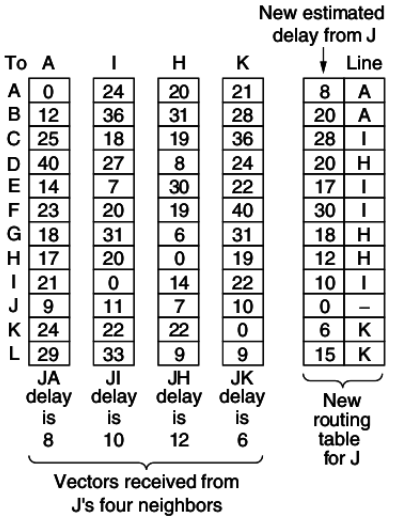
A distance vector routing algorithm operates by having each router maintain a table (i.e., a vector) giving the best known distance to each destination and which link to use to get there.

- These tables are updated by exchanging information with the neighbors. Eventually, every router knows the best link to reach each destination.
- It was the original ARPANET routing algorithm and was also used in the Internet under the name RIP.

DV Routing

Suppose that J has measured or estimated its delay to its neighbors, A, I, H, and K, as 8, 10, 12, and 6 msec, respectively.



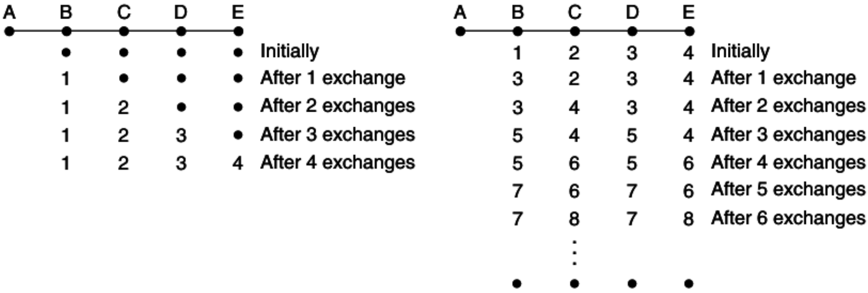


To	A	I	H	K	New estimated delay from J	Line
A	0	24	20	21	8	A
B	12	36	31	28	20	A
C	25	18	19	36	28	I
D	40	27	8	24	20	H
E	14	7	30	22	17	I
F	23	20	19	40	30	I
G	18	31	6	31	18	H
H	17	20	0	19	12	H
I	21	0	14	22	10	I
J	9	11	7	10	0	–
K	24	22	22	0	6	K
L	29	33	9	9	15	K
	JA delay is 8	JI delay is 10	JH delay is 12	JK delay is 6	New routing table for J	

Vectors received from J's four neighbors

Count to Infinity Problem

- The settling of routes to best paths across the network is called convergence.
- DV reacts rapidly to good news, but leisurely to bad news.



A	B	C	D	E	
	•	•	•	•	Initially
	1	•	•	•	After 1 exchange
	1	2	•	•	After 2 exchanges
	1	2	3	•	After 3 exchanges
	1	2	3	4	After 4 exchanges

A	B	C	D	E	
	1	2	3	4	Initially
	3	2	3	4	After 1 exchange
	3	4	3	4	After 2 exchanges
	5	4	5	4	After 3 exchanges
	5	6	5	6	After 4 exchanges
	7	6	7	6	After 5 exchanges
	7	8	7	8	After 6 exchanges
		⋮			
	•	•	•	•	

Figure: **A** was down then up, **A** was up then down

Link State Routing

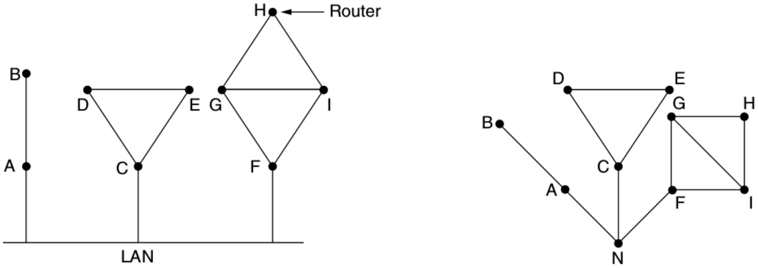
In link state routing each router must do the following:

- **Discover its neighbors** and learn their network addresses.
- **Set the distance or cost metric** to each of its neighbors.
- **Construct a packet** telling all it has just learned.
- Send this packet to and receive packets from **all other routers**.
- **Compute the shortest path** to every other router.

In effect, the complete topology is distributed to every router and Dijkstra's algorithm can be run at each router to find the shortest path to every other router.

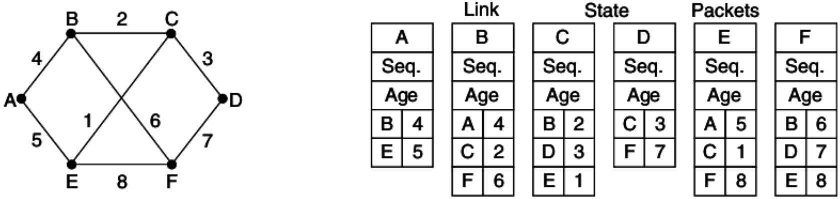
Link State Routing: Discovering Neighbors

- Sends a special **HELLO** packet on each point-to-point line. The router on the other end is expected to send back a reply giving its name.
- The router names must be globally unique.



(a) (b)

Link State Routing: Building Link State Packets



(a)

Link State Packets

A	
Seq.	
Age	
B	4
E	5

B	
Seq.	
Age	
A	4
C	2
F	6

C	
Seq.	
Age	
B	2
D	3
E	1

D	
Seq.	
Age	
C	3
F	7

E	
Seq.	
Age	
A	5
C	1
F	8

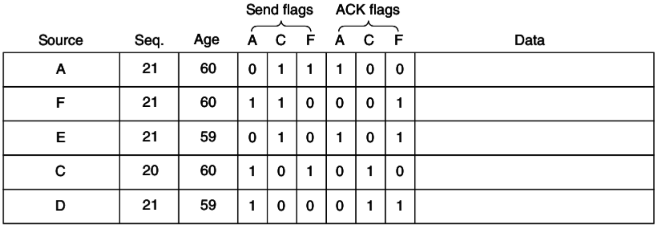
F	
Seq.	
Age	
B	6
D	7
E	8

(b)

Link State Routing: Distributing Link State Packets

The fundamental idea is to use flooding to distribute the link state packets to all routers. To keep the flood in check, each packet contains a sequence number that is incremented for each new packet sent.

The Age field is decremented by each router during the initial flooding process, to make sure no packet can get lost and live for an indefinite period of time.



			Send flags			ACK flags			
Source	Seq.	Age	A	C	F	A	C	F	Data
A	21	60	0	1	1	1	0	0	
F	21	60	1	1	0	0	0	1	
E	21	59	0	1	0	1	0	1	
C	20	60	1	0	1	0	1	0	
D	21	59	1	0	0	0	1	1	

Figure: The packet buffer for router B

Link State Routing: Computing Routes

Once a router has accumulated a full set of link state packets, it can construct the entire network graph because every link is represented.

Dijkstra's algorithm can be run locally to construct the shortest paths to all possible destinations.

OSPF and IS-IS are some examples.

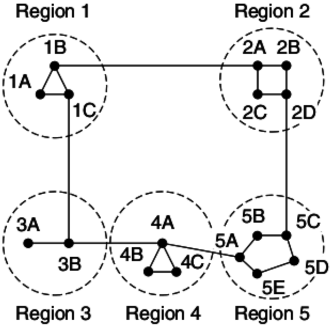
Hierarchical Routing

Can you guess why hierarchical routing?

Hierarchical Routing

- As networks grow in size, the router routing tables grow proportionally.
- More router memory, more CPU time and more bandwidth is needed to send status reports about them.
- The network may grow to the point where it is no longer feasible for every router to have an entry for every other router, so the routing will have to be done hierarchically, similar to telephone network.

Hierarchical Routing



(a)

Full table for 1A

Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2A	1B	2
2B	1B	3
2C	1B	3
2D	1B	4
3A	1C	3
3B	1C	2
4A	1C	3
4B	1C	4
4C	1C	4
5A	1C	4
5B	1C	5
5C	1B	5
5D	1C	6
5E	1C	5

(b)

Hierarchical table for 1A

Dest.	Line	Hops
1A	–	–
1B	1B	1
1C	1C	1
2	1B	2
3	1C	2
4	1C	3
5	1C	4

(c)

Broadcast Routing

- Sending a packet to all destinations simultaneously is called broadcasting.
- Simple **broadcast** and **multi-destination** broadcasting are naive techniques.
- Flooding is one of the better broadcasting techniques.
- Flooding can be bettered once the shortest path routes for regular packets have been computed (**Reverse path forwarding:**)

Reverse path forwarding:

- When a broadcast packet arrives at a router, the router checks to see if the packet arrived on the link that is **normally used for sending packets toward the source of the broadcast.**
- If so, there is an excellent chance that the **broadcast packet itself followed the best route from the router** and is therefore the first copy to arrive at the router.
- This being the case, the router forwards copies of it onto all links except the one it arrived on.
- If, however, the broadcast packet arrived on a link other than the preferred one for reaching the source, the packet is discarded as a likely duplicate.

Reverse Path Forwarding

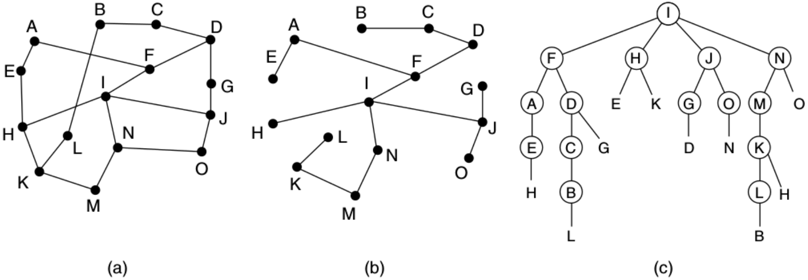


Figure 5-15. Reverse path forwarding. (a) A network. (b) A sink tree. (c) The tree built by reverse path forwarding.

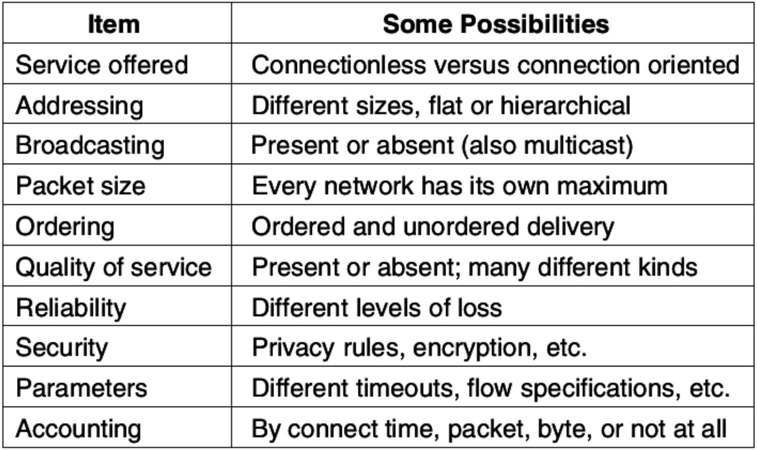
Reverse path forwarding:

- The principal advantage of reverse path forwarding is that it is efficient while **being easy to implement**.
- It sends the broadcast packet over each link only once in each direction, **just as in flooding**, yet it requires only that routers know how to reach all destinations, without needing to remember sequence numbers (or use other mechanisms to stop the flood) or list all destinations in the packet.

Internetworking

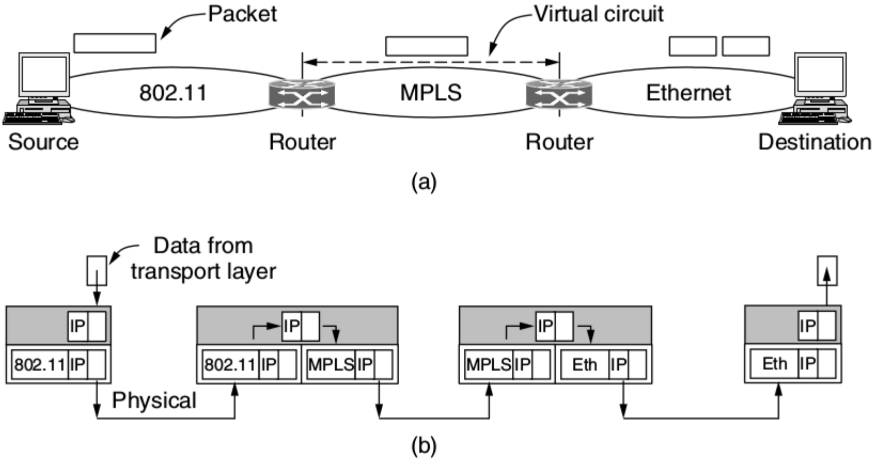
- What issues arise when two or more networks are connected to form an internetwork, or more simply an internet?

How networks differ?



Item	Some Possibilities
Service offered	Connectionless versus connection oriented
Addressing	Different sizes, flat or hierarchical
Broadcasting	Present or absent (also multicast)
Packet size	Every network has its own maximum
Ordering	Ordered and unordered delivery
Quality of service	Present or absent; many different kinds
Reliability	Different levels of loss
Security	Privacy rules, encryption, etc.
Parameters	Different timeouts, flow specifications, etc.
Accounting	By connect time, packet, byte, or not at all

How networks can be connected?



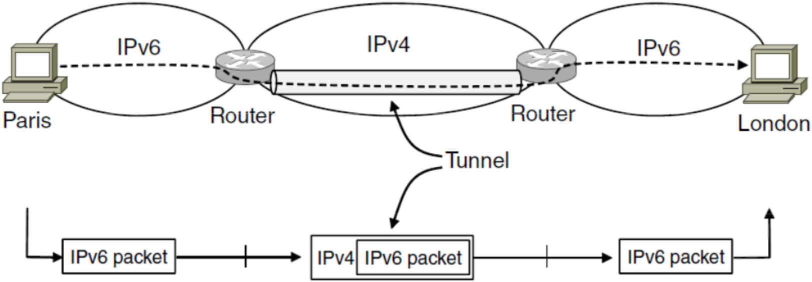
(a)

(b)

Tunneling

Connects two networks through a middle one

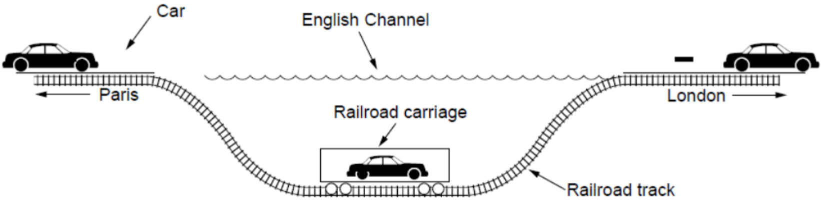
- Packets are encapsulates over the middle



Tunneling

Tunneling analogy:

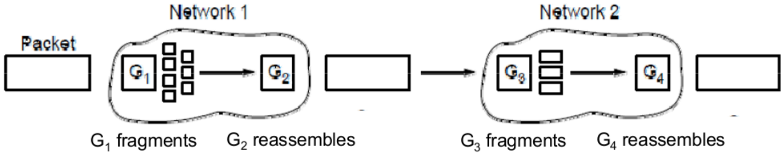
- tunnel is a link; packet can only enter/exit at ends



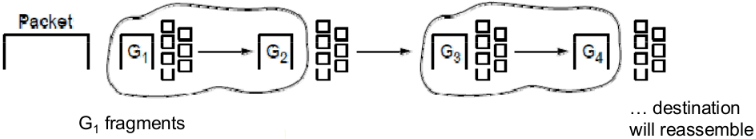
Packet Fragmentation

Networks have different packet size limits for many reasons

- Large packets sent with fragmentation & reassembly



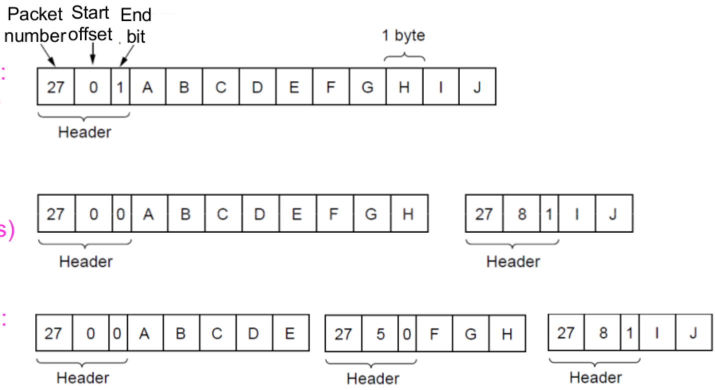
Transparent – packets fragmented / reassembled in each network



Non-transparent – fragments are reassembled at destination

Packet Fragmentation

Example of IP-style fragmentation:



Original packet:
(10 data bytes)

27 | 0 | 1 | A | B | C | D | E | F | G | H | I | J

Fragmented:
(to 8 data bytes)

27 | 0 | 0 | A | B | C | D | E | F | G | H

27 | 8 | 1 | I | J

Re-fragmented:
(to 5 bytes)

27 | 0 | 0 | A | B | C | D | E

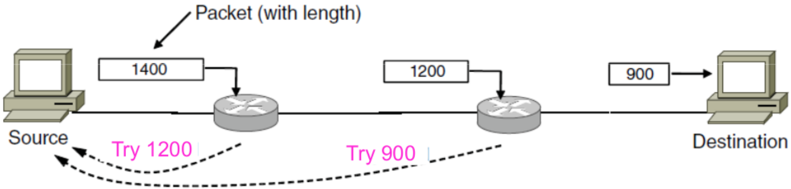
27 | 5 | 0 | F | G | H

27 | 8 | 1 | I | J

Path MTU Discovery

Path MTU Discovery avoids network fragmentation

- Routers return MTU (Max. Transmission Unit) to source and discard large packets



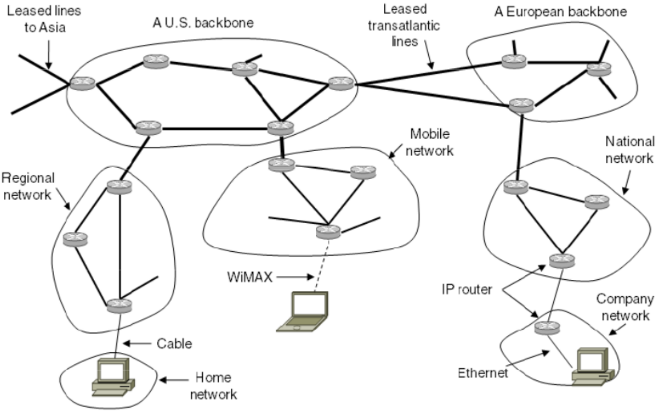
Network Layer in the Internet

IP has been shaped by guiding principles:

- Make sure it works
- Keep it simple
- Make clear choices
- Exploit modularity
- Expect heterogeneity
- Avoid static options and parameters
- Look for good design (not perfect)
- Strict sending, tolerant receiving
- Think about scalability
- Consider performance and cost

Network Layer in the Internet

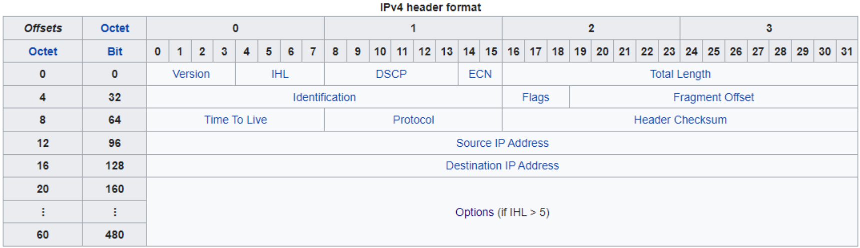
Internet is an interconnected collection of many networks that is held together by the IP protocol



Network Layer in the Internet

- The glue that holds the whole Internet together is the network layer protocol, IP (Internet Protocol).
- Unlike most older network layer protocols, IP was designed from the beginning with internetworking in mind.
- Network layer job is to provide a best-effort (i.e., not guaranteed) way to transport packets from source to destination, without regard to whether these machines are on the same network or whether there are other networks in between them.

IPv4 Protocol



IPv4 header format

Offsets	Octet	0								1								2								3							
Octet	Bit	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
0	0	Version				IHL				DSCP						ECN		Total Length															
4	32	Identification																Flags			Fragment Offset												
8	64	Time To Live								Protocol								Header Checksum															
12	96	Source IP Address																															
16	128	Destination IP Address																															
20	160	Options (if IHL > 5)																															
⋮	⋮																																
60	480																																

- It has a 20-byte fixed part and a variable-length optional part.
- The bits are transmitted from left to right and top to bottom, with the high-order bit of the Version field going first.

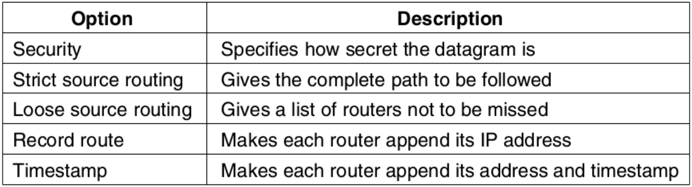
IPv4 Header

- The **Version** field keeps track of which version of the protocol the datagram belongs to.
- **IHL** represents Header length in 32-bit words.
- In **Differentiated Services** top 6 bits are used to mark the packet with its service class;(ex: expedited and assured services). The bottom 2 bits are used to carry explicit congestion notification information, such as whether the packet has experienced congestion.
- The **Total length** includes everything in the datagram- both header and data. The maximum length is 65,535 bytes.
- The **Identification field** is needed to allow the destination host to determine which packet a newly arrived fragment belongs to. All the fragments of a packet contain the same Identification value.

IPv4 Header

- **DF** stands for Don't Fragment. It is an order to the routers not to fragment the packet.
- **MF** stands for More Fragments. All fragments except the last one have this bit set. It is needed to know when all fragments of a datagram have arrived.
- The **Fragment offset** tells where in the current packet this fragment belongs. All fragments except the last one in a datagram must be a multiple of 8 bytes, the elementary fragment unit.
- The **TtL** (Time to live) field is a counter used to limit packet lifetimes. It just counts hops. When it hits zero, the packet is discarded and a warning packet is sent back to the source host.
- **Protocol field** tells which transport process to give the packet to.
- As the header carries vital information such as addresses, it rates its own checksum for protection, using **Header checksum** field.

IPv4 Header Options

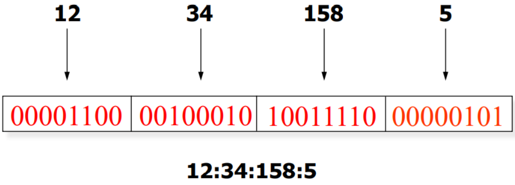


Option	Description
Security	Specifies how secret the datagram is
Strict source routing	Gives the complete path to be followed
Loose source routing	Gives a list of routers not to be missed
Record route	Makes each router append its IP address
Timestamp	Makes each router append its address and timestamp

Figure 5-47. Some of the IP options.

IP Addresses (IPv4)

- Every host and router on the Internet has an IP address: **a unique 32-bit number,** that can be used in the **SA** and **DA** fields of IP packets.
- IP address refers to a network interface, so if a host is on two networks, it must have two IP addresses (**routers have multiple interfaces**).



12 34 158 5

00001100 00100010 10011110 00000101

12:34:158:5

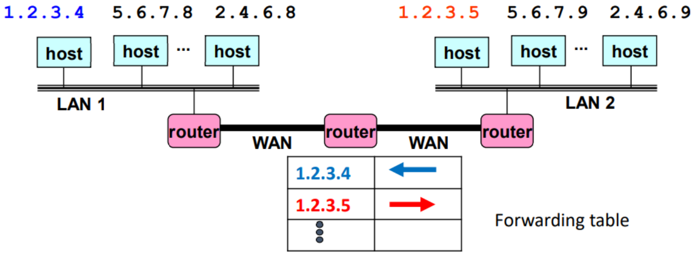
- Represented in **dotted decimal** notation.

IP Addresses (IPv4)

The Internet is an "inter-network"

- Used to connect networks together, not hosts
- Need to address a network (i.e., group of hosts)

Suppose every host has an arbitrary address – what could be the problem?

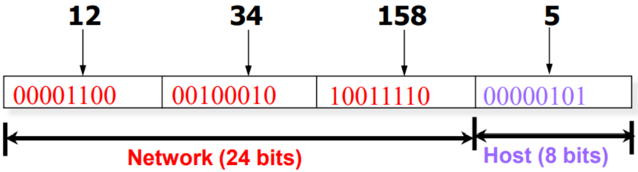


1.2.3.4	←
1.2.3.5	→
⋮	

Forwarding table

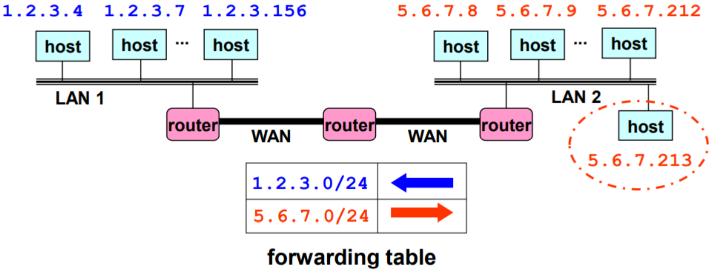
IP Addresses:Prefix

- IP addresses are hierarchical, unlike Ethernet addresses.
- Each 32-bit address is comprised of a **variable-length network portion in the top bits** and **a host portion in the bottom bits.**
- The network portion has the same value for all hosts on a single network, such as an Ethernet LAN.
- This means that a **network** corresponds to a **contiguous block of IP address space.** This block is called a **prefix.**



IP Addresses:Prefix

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



1.2.3.0/24	←
5.6.7.0/24	→

forwarding table

IP Addresses:Prefix

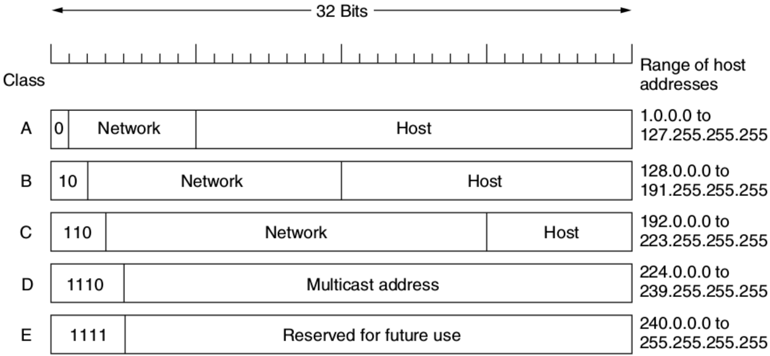
- Prefixes are written by giving the **lowest IP address in the block** and the **size of the block**.
- The size is determined by the number of bits in the network portion; the remaining bits in the host portion can vary.
- It is written after the prefix IP address as a **slash(/)** followed by the length in bits of the network portion.
- Eg: If the prefix contains 2^8 addresses, it leaves 24 bits for the **network portion**, it is written as 128.208.2.0/24.
- Since the prefix length cannot be inferred from the IP address alone, routing protocols **must carry the prefixes** to routers.
- The length of the prefix corresponds to a binary mask of **1s** in the network portion. When written out this way, it is called a **(sub)net mask**.
- It can be ANDed with the IP address to extract only the network portion. For our example, the (sub)net mask is 255.255.255.0.

IP Addresses

- **Address** - The unique number ID assigned to one host or interface in a network.
- A **network** corresponds to a **contiguous block of IP address space.** This block is called a **prefix.**
- **Subnet** - A portion of a network that shares a particular subnet address.
- **Subnet mask** - A 32-bit combination used to describe which portion of an address refers to the subnet and which part refers to the host.
- **Interface** - A network connection.

Classful IP Addresses

Before 1993, IP addresses were divided into the five categories.



Class				Range of host addresses
A	0	Network	Host	1.0.0.0 to 127.255.255.255
B	10	Network	Host	128.0.0.0 to 191.255.255.255
C	110	Network	Host	192.0.0.0 to 223.255.255.255
D	1110	Multicast address		224.0.0.0 to 239.255.255.255
E	1111	Reserved for future use		240.0.0.0 to 255.255.255.255

Figure 5-53. IP address formats.

How to identify a class – use the first few bits
0 – Class A, 10 – Class B, 110 – Class C, 1110 – Class D, 1111 – Class E

Classful IP Addresses

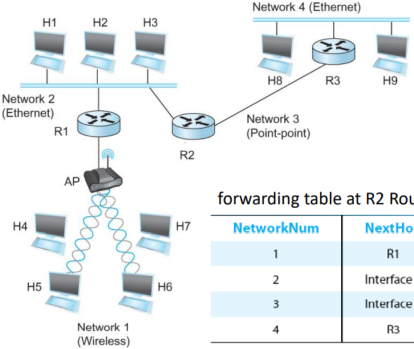
- Class A address: the 1st octet is the network portion. So the Class A has a major network address of 1.0.0.0 - 127.255.255.255. (Network 0 and 127 are reserved).
- Used for networks that have more than 65,536 hosts (actually, up to 16777214 hosts!).
- In all network classes, **host numbers** 0 and 255 are reserved.
- IP address with all **host bits set to zero** identifies the network itself.
- **Eg:** 26.0.0.0 refers to network 26, & 128.66.0.0 refers to network 128.66.
- Addresses in this form are used in routing table listings to refer to entire networks.
- IP address with all bits set to one is a broadcast address that is used to simultaneously address every host on a network.
- **Eg:** The broadcast address for network 128.66 is 128.66.255.255.

Classful IP Addresses

- In a Class B address, the first two octets are the network portion(2^{14} networks) and has a major network address of 128.0.0.0 - 191.255.255.255.
- Class B addresses are used for networks that have between 256 and 65534 hosts.
- The Class C address has a major network address of 192.0.0.0 - 223.255.255.255. Octet 4 (8 bits) is for local sub-nets and hosts - perfect for networks with less than 254 hosts.

IP datagram forwarding

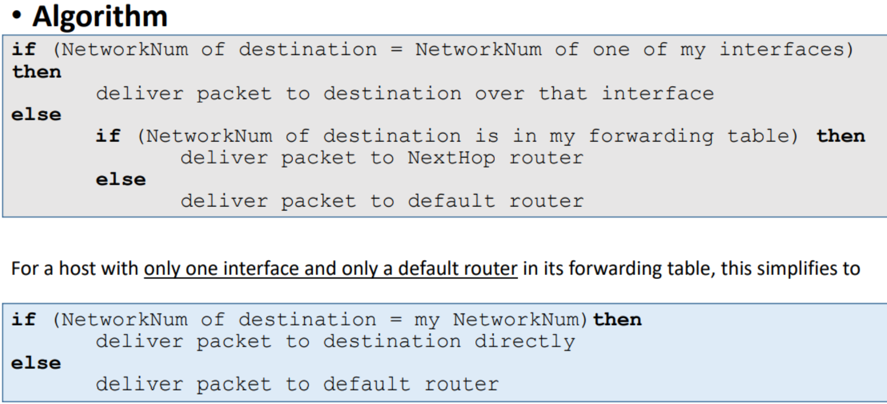
- Strategy
  - every datagram contains destination's address
  - if directly connected to destination network, then forward to host
  - if not directly connected to destination network, then forward to some router
  - forwarding table maps network number into next hop
  - each host has a default router
  - each router maintains a forwarding table



forwarding table at R2 Router

NetworkNum	NextHop
1	R1
2	Interface 1
3	Interface 0
4	R3

IP datagram forwarding



- **Algorithm**

```
if (NetworkNum of destination = NetworkNum of one of my interfaces)
then
      deliver packet to destination over that interface
else
      if (NetworkNum of destination is in my forwarding table) then
            deliver packet to NextHop router
      else
            deliver packet to default router
```

For a host with only one interface and only a default router in its forwarding table, this simplifies to

```
if (NetworkNum of destination = my NetworkNum) then
      deliver packet to destination directly
else
      deliver packet to default router
```

Obtaining IP Addresses

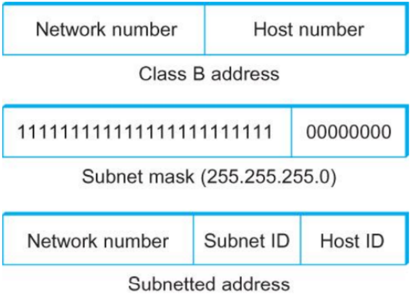
- Network numbers are managed by a non-profit corporation called **ICANN (Internet Corporation for Assigned Names and Numbers)**, to avoid conflicts.
- In turn, **ICANN** has delegated parts of the address space to various **regional authorities**, which dole out IP addresses to ISPs and other companies.
- Internet Service Providers (ISPs) allocate address blocks to their customers, who may, in turn, allocate to their customers..

Issues with Classful addressing

- You have 255 hosts in a network. Which IPv4 address class will you use – Class C or Class B ?
  - Class C – not possible
  - Class B – huge address space is lost (using only 255 addresses out of possible 2^{16}-2 addresses)
- Major problems are wastage of addresses and fewer number of available addresses.

Sub-netting

- Add another level to address/routing hierarchy: subnet
- Subnetting: divide a large network into multiple small networks for efficient use of address space
- Subnet masks denote the number of bits in the **network address** field



Network number | Host number

Class B address

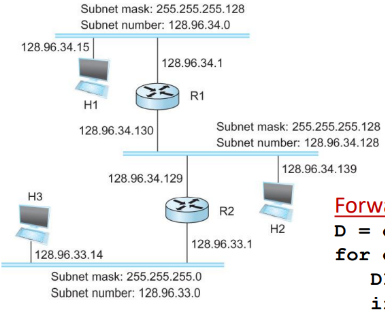
111111111111111111111111 | 00000000

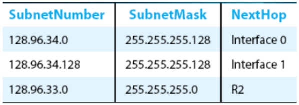
Subnet mask (255.255.255.0)

Network number | Subnet ID | Host ID

Subnetted address

Sub-netting


Subnet mask: 255.255.255.128
Subnet number: 128.96.34.0
128.96.34.15
128.96.34.1
H1
R1
128.96.34.130
Subnet mask: 255.255.255.128
Subnet number: 128.96.34.128
128.96.34.129
128.96.34.139
H3
R2
H2
128.96.33.1
128.96.33.14
Subnet mask: 255.255.255.0
Subnet number: 128.96.33.0



SubnetNumber	SubnetMask	NextHop
128.96.34.0	255.255.255.128	Interface 0
128.96.34.128	255.255.255.128	Interface 1
128.96.33.0	255.255.255.0	R2

Forwarding Algorithm

```
D = destination IP address
for each entry < SubnetNum, SubnetMask, NextHop
    D1 = SubnetMask & D
    if D1 = SubnetNum
        if NextHop is an interface
            deliver datagram directly to destination
        else
            deliver datagram to NextHop (a router)
```

sub-nets

- Routing by prefix requires all the hosts in a network to have the same network number. This property can cause problems as networks grow.
- For example, consider a university that started out with **/16** prefix for use by the Computer Science Dept. for the computers on its Ethernet.
- A year later, the Electrical Engineering Dept. wants to get on the Internet. The Art Dept. soon follows suit. **What IP addresses should these departments use?**

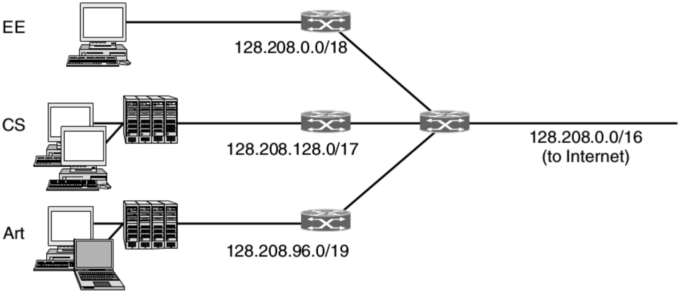
subnets

- Getting further blocks requires going outside the university and may be expensive or inconvenient.
- Moreover, the /16 already allocated has enough addresses for over **60,000** hosts.
- It might be intended to allow for significant growth, but until that happens, it is wasteful to allocate further blocks of IP addresses to the same university. **A different organization is required.**

subnets

- The solution is to allow the block of addresses to be split into several parts for internal use as multiple networks, while still acting like a single network to the outside world.
- This is called **subnetting** and the networks (such as Ethernet LANs) that result from dividing up a larger network are called subnets.

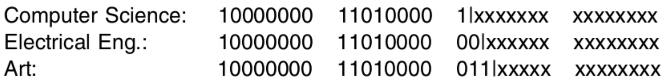
subnets



- The single **/16** has been split into pieces. This split does not need to be even, but each piece must be aligned so that any bits can be used in the lower host portion.
- In this case, half of the block **(a/17)** is allocated to the Computer Science Dept, a quarter is allocated to the Electrical Engineering Dept. (**a /18)**, and one eighth **(a/19)** to the Art Dept. The remaining eighth is unallocated

subnets

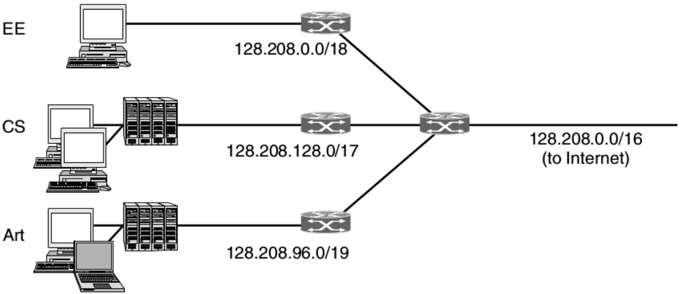
- A different way to see how the block was divided is to look at the resulting prefixes when written in binary notation:
- Here, the vertical bar (|) shows the boundary between the subnet number and the host portion.



Computer Science:	10000000	11010000	1|xxxxxxx	xxxxxxxx
Electrical Eng.:	10000000	11010000	00|xxxxxx	xxxxxxxx
Art:	10000000	11010000	011|xxxxx	xxxxxxxx

subnets

- When a packet comes into the main router, how does the router know which subnet to give it to?
- One way would be for each router to have a table with 65,536 entries telling it which outgoing line to use for each host on campus.
- But this would undermine the main scaling benefit we get from using a hierarchy. Instead, the routers simply need to know the subnet masks for the networks on campus.



subnets

- When a packet arrives, the router looks at the destination address of the packet and checks which subnet it belongs to.
- The router can do this by **ANDing** the **destination address** with the **mask for each subnet** and checking to see if the result is the corresponding prefix.
- For example, consider a packet destined for IP address 128.208.2.151. To see if it is for the Computer Science Dept., we AND with 255.255.128.0 to take the first 17 bits (which is 128.208.0.0) and see if they match the prefix address (which is 128.208.128.0). They do not match.

subnets

- Checking the first 18 bits for the Electrical Engineering Dept., we get 128.208.0.0 when ANDing with the subnet mask. This does match the prefix address, so the packet is forwarded onto the interface which leads to the Electrical Engineering network.
- The subnet divisions can be changed later if necessary, by updating all subnet masks at routers inside the university.

Understanding Subnetting

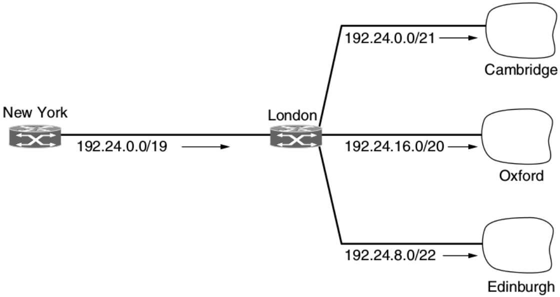
If you have network 172.16.0.0, then you know that its natural mask is 255.255.0.0 or 172.16.0.0/16. Extending the mask to anything beyond 255.255.0.0 means you are subnetting.

If you use a mask of 255.255.248.0 (/21), how many subnets and hosts per subnet does this allow for?

CIDR: Classless InterDomain Routing

Sub-netting allows addresses to be used efficiently, but there is still a problem that remains: routing table explosion.

Example:



CIDR: Classless InterDomain Routing

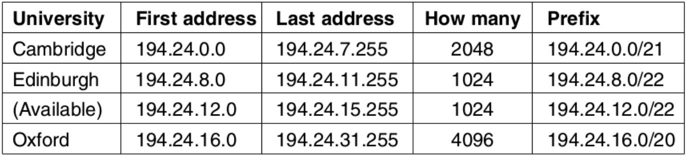
- Routers in organizations at the edge of a network, such as a university, need to have an entry for each of their sub-nets, telling the router which line to use to get to that network.
- For routes to destinations outside of the organization, they can use the simple default rule of sending the packets on the line toward the ISP that connects the organization to the rest of the Internet.
- Routers in ISPs and backbones in the middle of the Internet must know which way to go to get to every network and no simple default will work.
- These core routers are said to be in the **default-free zone** of the Internet.
- **Challenges:** Large tables, large lookup time, routing table population.

CIDR: Classless InterDomain Routing

- **Solution:** Combine multiple small prefixes into a single larger prefix. This process is called route aggregation.
- This design works with sub-netting and is called CIDR.

CIDR: Example

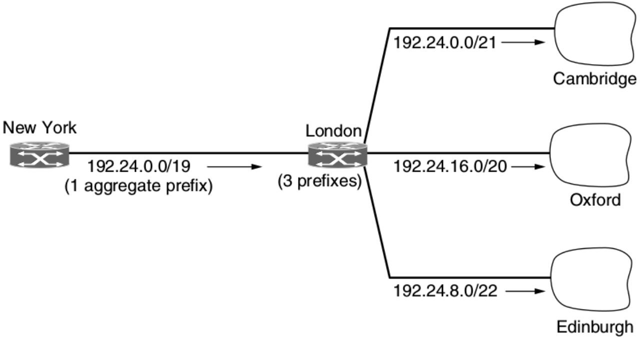
- Let us consider an example in which a block of 8192 IP addresses is available starting at 194.24.0.0.
- Suppose, Cambridge University needs 2048 addresses. Next, Oxford University asks for 4096 addresses. Finally, the University of Edinburgh asks for 1024 addresses.



University	First address	Last address	How many	Prefix
Cambridge	194.24.0.0	194.24.7.255	2048	194.24.0.0/21
Edinburgh	194.24.8.0	194.24.11.255	1024	194.24.8.0/22
(Available)	194.24.12.0	194.24.15.255	1024	194.24.12.0/22
Oxford	194.24.16.0	194.24.31.255	4096	194.24.16.0/20

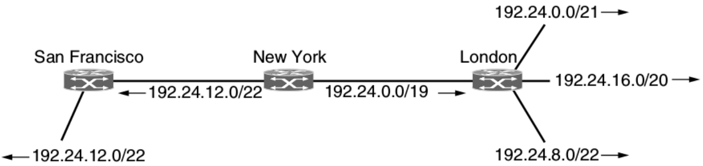
CIDR

- All of the routers in the default-free zone are now told about the IP addresses in the three networks.
- All of the IP addresses in the three prefixes should be sent from New York (or the U.S. in general) to London.



CIDR

- Prefixes are allowed to overlap. The rule is that packets are sent in the direction of the most specific route, or the longest matching prefix.

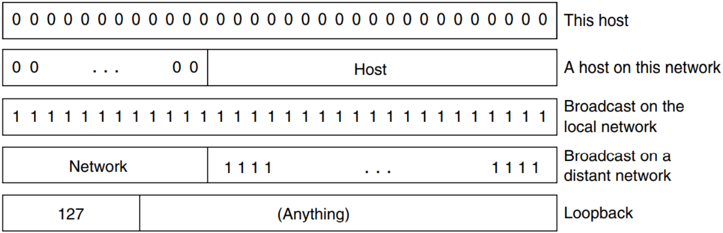


Summary of IP addressing

- Classful addressing
- sub-nets
- CIDR was added to reduce the size of the global routing table.
- Today, the bits that indicate whether an IP address belongs to class A, B, or C network are no longer used.

Special Addresses

- The IP address 0.0.0.0, the lowest address, is used by hosts when they are being booted. It means "this network" or "this host."



Address	Meaning
0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0	This host
0 0 . . . 0 0 | Host	A host on this network
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	Broadcast on the local network
Network | 1 1 1 1 . . . 1 1 1 1	Broadcast on a distant network
127 | (Anything)	Loopback

Network Address Translation

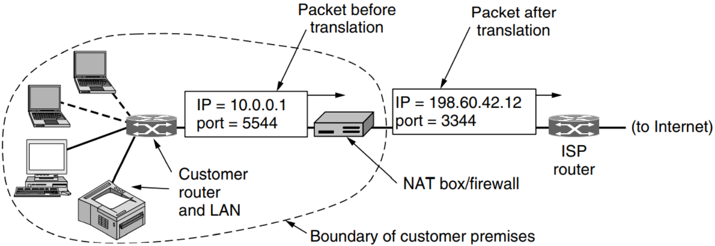
- IP addresses are scarce.
- An ISP might have a /16 address, giving it 65,534 usable host numbers. If it has more customers than that, it has a problem.
- This scarcity has led to techniques to use IP addresses sparingly.
- One approach is to dynamically assign an IP address to a computer when it is on and using the network.
- With the techniques we have seen so far, **each computer must have its own IP address all day long.**

Network Address Translation

- The basic idea behind NAT is for the ISP to assign each home or business a single IP address (or at most, a small number of them) for Internet traffic.
- Within the customer network, every computer gets a unique IP address, which is used for routing intramural traffic.
- However, just before a packet exits the customer network and goes to the ISP, an address translation from the unique internal IP address to the shared public IP address takes place.
- This translation makes use of three ranges of IP addresses that have been declared as private.
- **No packets containing these addresses may appear on the Internet**. The three reserved ranges are:

10.0.0.0 – 10.255.255.255/8 (16,777,216 hosts)
172.16.0.0 – 172.31.255.255/12 (1,048,576 hosts)
192.168.0.0 – 192.168.255.255/16 (65,536 hosts)

NAT operation



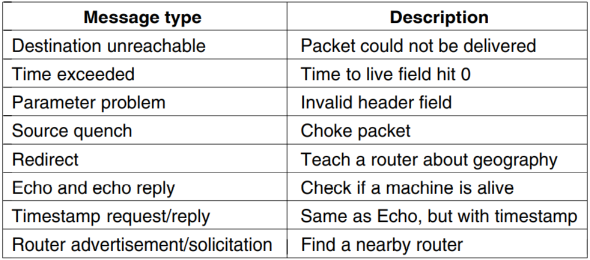
- NAT designers observed that most IP packets carry either TCP or UDP payloads.
- Whenever an outgoing packet enters the NAT box, the 10.x.y.z source address is replaced by the customer's true IP address.
- The TCP Source port field is replaced by an index into the NAT box's 65,536-entry translation table.
- This table entry contains the original IP address and the original source port.

Internet Control Protocols

- IP is used for data transfer. Internet has several companion control protocols that are used in the network layer.
- They include ICMP, ARP, and DHCP.

Internet Control Message Protocol

- When something unexpected occurs during packet processing at a router, the event is reported to the sender by ICMP.
- ICMP is also used to test the Internet.
- Each ICMP message type is carried encapsulated in an IP packet.

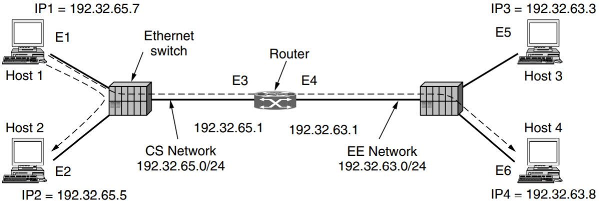


Message type	Description
Destination unreachable	Packet could not be delivered
Time exceeded	Time to live field hit 0
Parameter problem	Invalid header field
Source quench	Choke packet
Redirect	Teach a router about geography
Echo and echo reply	Check if a machine is alive
Timestamp request/reply	Same as Echo, but with timestamp
Router advertisement/solicitation	Find a nearby router

All messages can be found at:
www.iana.org/assignments/icmp-parameters.

Address Resolution Protocol

- IP addresses are not sufficient for sending packets.
- Data link layer NICs such as Ethernet cards do not understand Internet addresses.
- How do IP addresses get mapped onto data link layer addresses, such as Ethernet?



Frame	Source IP	Source Eth.	Destination IP	Destination Eth.
Host 1 to 2, on CS net	IP1	E1	IP2	E2
Host 1 to 4, on CS net	IP1	E1	IP4	E3
Host 1 to 4, on EE net	IP1	E4	IP4	E6

Address Resolution Protocol

Let us assume the sender(H1) knows the name of the intended receiver(H2):

- Step 1: To find the IP address for host 2. This lookup is done by DNS.
- Step 2: Build a packet with 192.32.65.5 in the Destination address field and gives it to the IP software to transmit.
- Step 3:Host 1 outputs a broadcast packet onto the Ethernet asking who owns IP address 192.32.65.5(H2).
- Step 4: Host 2 alone will respond with its Ethernet address (E2)

Optimizations: Caching, gratuitous ARP.

Address Resolution Protocol

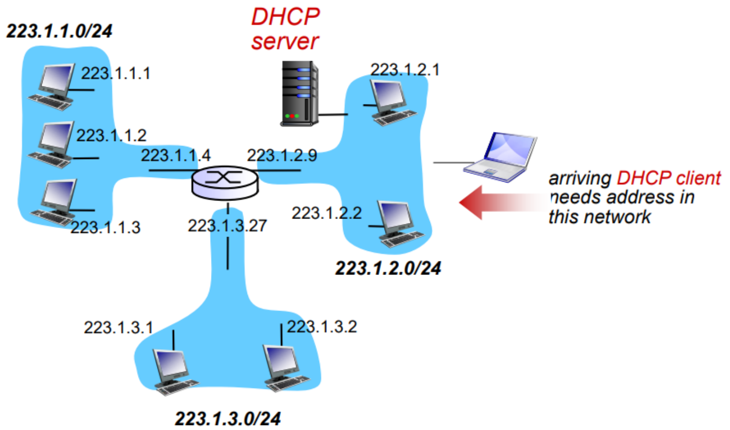
- How to deal when a host in on another network?
- Use the default gateway or proxy ARP.

Dynamic Host Configuration Protocol

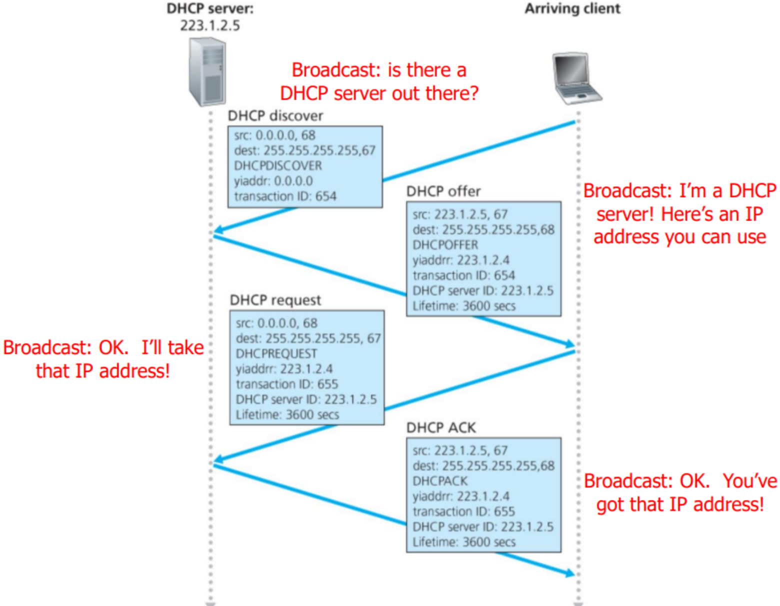
- ARP (as well as other Internet protocols) makes the assumption that hosts are configured with some basic information, such as their own IP addresses. How do hosts get this information?

DHCP overview:
host broadcasts “DHCP discover” msg [optional]
DHCP server responds with “DHCP offer” msg [optional]
host requests IP address: “DHCP request” msg
DHCP server sends address: “DHCP ack” msg

DHCP Client-Server Scenario



DHCP Operation



DHCP server:
223.1.2.5
Arriving client
Broadcast: is there a DHCP server out there?
DHCP discover
DHCP offer
Broadcast: I'm a DHCP server! Here's an IP address you can use
DHCP request
Broadcast: OK. I'll take that IP address!
DHCP ACK
Broadcast: OK. You've got that IP address!

DHCP discover

src: 0.0.0.0, 68
dest: 255.255.255.255,67
DHCPDISCOVER
yiaddr: 0.0.0.0
transaction ID: 654

DHCP offer

src: 223.1.2.5, 67
dest: 255.255.255.255,68
DHCPOFFER
yiaddrr: 223.1.2.4
transaction ID: 654
DHCP server ID: 223.1.2.5
Lifetime: 3600 secs

DHCP request

src: 0.0.0.0, 68
dest: 255.255.255.255, 67
DHCPREQUEST
yiaddrr: 223.1.2.4
transaction ID: 655
DHCP server ID: 223.1.2.5
Lifetime: 3600 secs

DHCP ACK

src: 223.1.2.5, 67
dest: 255.255.255.255,68
DHCPACK
yiaddrr: 223.1.2.4
transaction ID: 655
DHCP server ID: 223.1.2.5
Lifetime: 3600 secs

DHCP Operation

- DHCP server is responsible for providing configuration information to hosts
- There is at least one DHCP server for an administrative domain
- DHCP server maintains a pool of available addresses
- Newly booted or attached host sends DHCPDISCOVER message to a special IP address (255.255.255.255)
- DHCP relay agent unicasts the message to DHCP server and waits for the response