Congestion Control

- When too many packets present in (a part of) the network causes packet delay and loss that degrades performance. This situation is called **congestion**.
- The network and transport layers share the responsibility for handling congestion. Since congestion occurs within the network, it is the network layer that directly experiences it and must ultimately determine what to do with the excess packets.
- However, the most effective way to control congestion is to reduce the load that the transport layer is placing on the network.

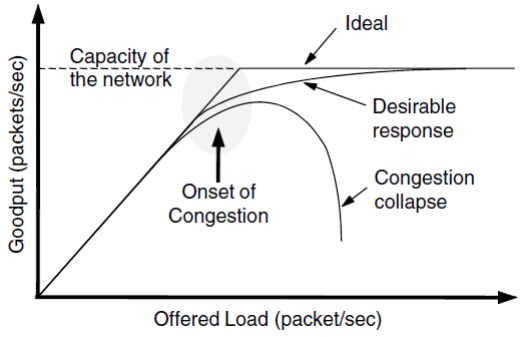
63

Network Congestion

- What is network congestion?
  - Too many packets in the network.
  - Router queues are always full.
    - Routers start dropping packets.
  - Congestion can fuel itself.
    - Packet drops lead to retransmissions.
    - More traffic!
  - May result in congestion collapse!
    - Close to 0 throughput!

64

Congestion Control



When too much traffic is offered, congestion sets in and performance degrades sharply.

65

Congestion Control

- When the number of packets dumped into the subnet by the hosts is within its carrying capacity, they are all delivered (except for a few that are afflicted with transmission errors) and the number delivered is proportional to the number sent.
- However, as traffic increases too far, the routers are no longer able to cope and they begin losing packets. This tends to make matters worse.
- At very high traffic, performance collapses completely and almost no packets are delivered.

66

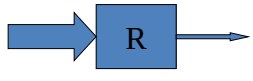
Infinite-Buffer Routers

- Intuition says add more memory to routers and that'll avoid congestion.
  - Nagle (1987) showed that infinite buffers actually make congestion worse.
  - More packets enqueued for long time; they time out and are retransmitted; but still transmitted by router.
  - Therefore, more traffic.

67

Causes of Congestion

- Mismatch in capacity among different parts of the system.
  - Mismatch in link speeds.
  - Mismatch in router processing capability.
    - Table lookup and update.
    - Queue management.



- Congestion in one point of network tends to propagate backwards toward sender.

68

Congestion versus Flow Control

- Congestion control tries to ensure the network is able to carry offered traffic.
  - It is a global issue, involving the behavior of all the hosts and intermediate routers.
- Flow control ensures that the communication end-points are able to keep up with one another.
  - Involves only the end-points.

69

Congestion and Flow Control

- Often mixed because tend to use same feedback mechanisms.
  - Example: “slow down” message received at host may be caused by receiver not being able to keep up with sender host or by network not being able to handle additional traffic.

70

General Principles of Congestion Control

- From control theory point of view:
  - Open and closed loop solutions.
- Open loop solutions:
  - Avoidance approach.
    - Tries to make sure problem doesn't happen.
    - Doesn't take current network state into account.
- Closed loop solutions:
  - Feedback loop.

71

Closed Loop Solutions

1. Monitor the system.
   - detect when and where congestion occurs.
2. Pass information to where action can be taken.
3. Adjust system operation to correct the problem.

72

Closed Loop Solutions

- 3 components:
  - Monitoring.
  - Feedback generation.
  - Operation adjustment.
- Monitoring metrics:
  - Packet loss.
  - Average queue length.
  - Number of retransmitted packets.
  - Average packet delay.

73

Feedback

- Send information about the problem once it's detected.
  - Router that detects problem sends packet to traffic source(s).
  - Special-purpose bit in every packet that router sets when it detects congestion above certain level to warn neighbors.
  - Special probe messages to detect congested areas so they can be avoided.

74

Congestion Control Taxonomy

- Open loop algorithms:
  - Act at source.
  - Act at destination.
- Closed loop algorithms:
  - Explicit feedback.
  - Implicit feedback.

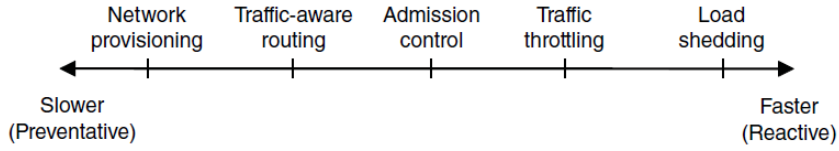
75

Approaches to Congestion Control

- The presence of congestion means that the load is (temporarily) greater than the resources (in a part of the network) can handle.
- Two solutions come to mind: increase the resources or decrease the load.
- Figure on next slide shows that, these solutions are usually applied on different time scales to either prevent congestion or react to it once it has occurred.

76

Approaches to Congestion Control



Timescales of approaches to congestion control

77

Network Provisioning

- The most basic way to avoid congestion is to build a network that is well matched to the traffic that it carries.
- If there is a low-bandwidth link on the path along which most traffic is directed, congestion is likely.
- Sometimes resources can be added dynamically when there is serious congestion, for example, turning on spare routers or enabling lines that are normally used only as backups (to make the system fault tolerant) or purchasing bandwidth on the open market.
- More often, links and routers that are regularly heavily utilized are upgraded at the earliest opportunity.
- This is called **provisioning** and happens on a time scale of months, driven by long-term traffic trends.

78

Traffic-Aware Routing

- To use most of the existing network capacity, routes can be tailored to traffic patterns that change during the day as network users wake and sleep in different time zones.
- For example, routes may be changed to shift traffic away from heavily used paths by changing the shortest path weights.
- Some local radio stations have helicopters flying around their cities to report on road congestion to make it possible for their mobile listeners to route their packets (cars) around hotspots.
- This is called traffic-aware routing. Splitting traffic across multiple paths is also helpful.

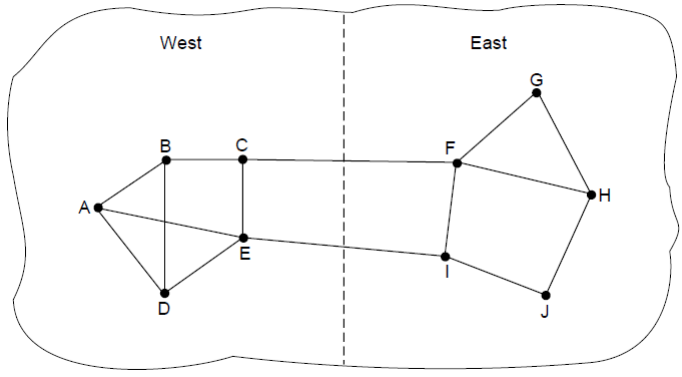
79

Traffic-Aware Routing

- The routing schemes, we have looked at, used fixed link weights. These schemes adapted to changes in topology, but not to changes in load.
- The goal in taking load into account when computing routes is to shift traffic away from hotspots that will be the first places in the network to experience congestion.
- The most direct way to do this is to set the link weight to be a function of the (fixed) link bandwidth and propagation delay plus the (variable) measured load or average queuing delay.
- Least-weight paths will then favor paths that are more lightly loaded, all else being equal.
- Traffic-aware routing was used in the early Internet according to this model However, there is a peril.

80

Traffic-Aware Routing



A network in which the East and West parts are connected by two links.

81

Traffic-Aware Routing

- If load is ignored and only bandwidth and propagation delay are considered, this problem does not occur.
- Attempts to include load but change weights within a narrow range only slow down routing oscillations.
- Two techniques can contribute to a successful solution. The first is multipath routing, in which there can be multiple paths from a source to a destination.
- The second one is for the routing scheme to shift traffic across routes slowly enough that it is able to converge

82

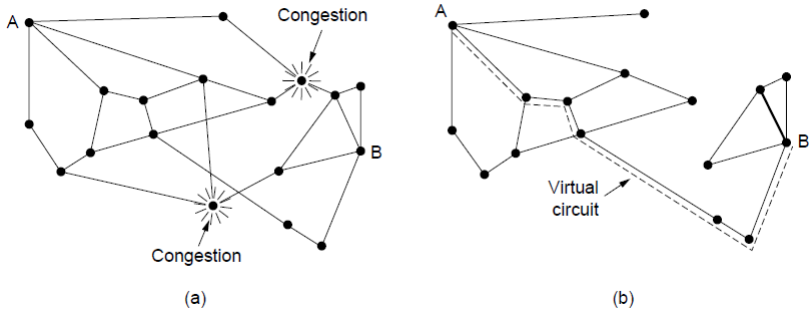
Admission Control

- However, sometimes it is not possible to increase capacity. The only way then to beat back the congestion is to decrease the load.
- In a virtual-circuit network, new connections can be refused if they would cause the network to become congested. This is called admission control.
- However, virtual circuits in computer networks come in all shapes and sizes. Thus, the circuit must come with some characterization of its traffic if we are to apply admission control.
- Traffic is often described in terms of its rate and shape. The problem of how to describe it in a simple yet meaningful way is difficult because traffic is typically bursty—the average rate is only half the story.

83

Admission Control

- Admission control can also be combined with traffic-aware routing by considering routes around traffic hotspots as part of the setup procedure.



(a) A congested network. (b) The portion of the network that is not congested. A virtual circuit from A to B is also shown.

84

Traffic Throttling

- At a finer granularity, when congestion is imminent the network can deliver feedback to the sources whose traffic flows are responsible for the problem.
- The network can request these sources to throttle their traffic, or it can slow down the traffic itself.
- Two difficulties with this approach are:
  - how to identify the onset of congestion, and
  - how to inform the source that needs to slow down.

85

Traffic Throttling

- To tackle the first issue, routers can monitor:
  - The average load (i.e. utilization of the output links),
  - Queueing delay (the buffering of queued packets inside the router), or
  - Packet loss.
- In all cases, rising numbers indicate growing congestion. Of these possibilities, the second one is the most useful.
- Averages of utilization do not directly account for the burstiness of most traffic—a utilization of 50% maybe low for smooth traffic and too high for highly variable traffic.

86

Traffic Throttling

- Counts of packet losses come too late. Congestion has already set in by the time that packets are lost.
- The queueing delay inside routers directly captures any congestion experienced by packets. It should be low most of time, but will jump when there is a burst of traffic that generates a backlog.
- To maintain a good estimate of the queueing delay d, a sample of the instantaneous queue length, s, can be made periodically and d updated according to

$$d_{new} = \alpha d_{old} + (1 - \alpha)s$$

  where the constant α determines how fast the router forgets recent history.
- This is called an **EWMA(Exponentially Weighted Moving Average**).

87

Traffic Throttling

- To tackle the second issue, routers must deliver timely feedback to the senders that are causing the congestion.
- Congestion is experienced in the network, but relieving congestion requires action on behalf of the senders that are using the network.
- To deliver feedback, the router must identify the appropriate senders. It must then warn them carefully, without sending many more packets into the already congested network.

88

Choke Packets

- The most direct way to notify a sender of congestion is to tell it directly.
- In this approach, the router selects a congested packet and sends a choke packet back to the source host, giving it the destination found in the packet.
- The original packet may be tagged (a header bit is turned on) so that it will not generate any more choke packets farther along the path and then forwarded in the usual way.
- To avoid increasing load on the network during a time of congestion, the router may only send choke packets at a low rate.

89

Choke Packets

- When the source host gets the choke packet, it is required to reduce the traffic sent to the specified destination, for example, by 50%.
- It is likely that multiple choke packets will be sent to a given host and destination.
- The host should ignore these additional chokes for the fixed time interval until its reduction in traffic takes effect.
- After that period, further choke packets indicate that the network is still congested.

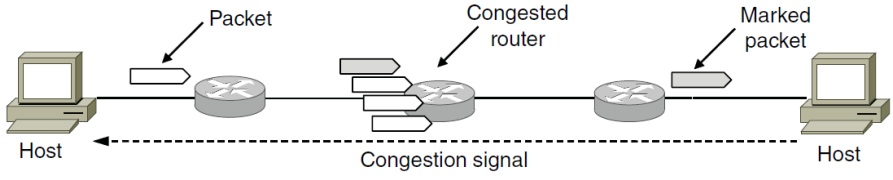
90

Explicit congestion notification

- Instead of generating additional packets to warn of congestion, a router can tag any packet it forwards (by setting a bit in the packet's header) to signal that it is experiencing congestion.
- When the network delivers the packet, the destination can note that there is congestion and inform the sender when it sends a reply packet. The sender can then throttle its transmissions as before.
- This design is called ECN(Explicit Congestion Notification) and is used in the Internet (Ramakrishnan et al., 2001).
- Two bits in the IP packet header are used to record whether the packet has experienced congestion.

91

Explicit congestion notification



Explicit congestion notification

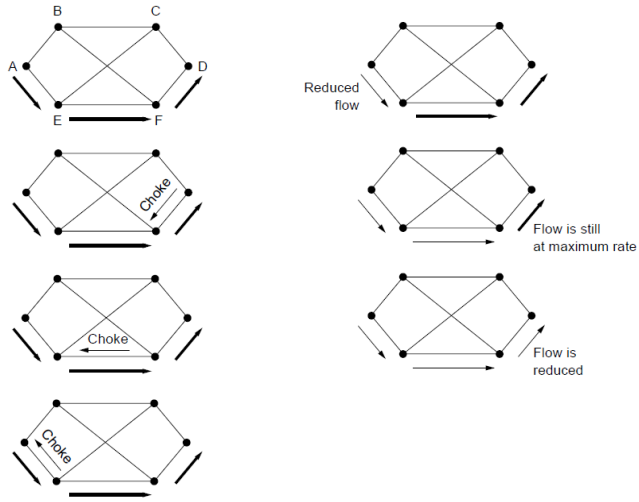
- If any of the routers they pass through is congested, that router will then mark the packet as having experienced congestion as it is forwarded.
- The destination will then echo any marks back to the sender as an explicit congestion signal in its next reply packet.

92

Hop-by-Hop Backpressure

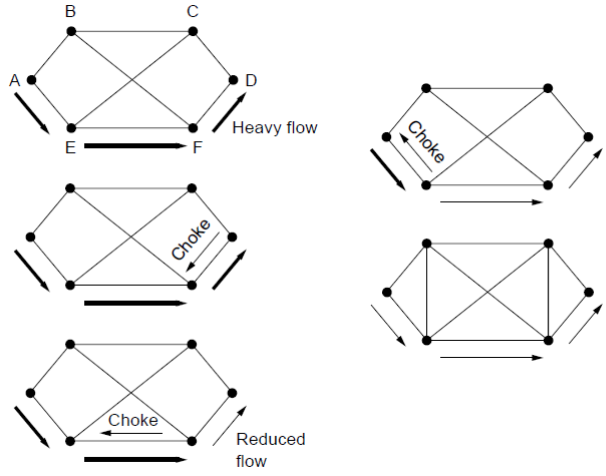
- At high speeds or over long distances, many new packets may be transmitted after congestion has been signaled because of the delay before the signal takes effect.
- Consider, for example, a host in San Francisco (router A in Fig. of next slide) that is sending traffic to a host in New York (router D in Fig.) at the OC-3 speed of 155 Mbps.
- If the New York host begins to run out of buffers, it will take about 40 msec for a choke packet to get back to San Francisco to tell it to slow down.
- An ECN indication will take even longer because it is delivered via the destination.

Hop-by-Hop Backpressure (1)



A choke packet that affects only the source..

Hop-by-Hop Backpressure(2)



A choke packet that affects each hop it passes through.

95

Load Shedding

- When all else fails, the network is forced to discard packets that it cannot deliver. The general name for this is load shedding.
- A good policy for choosing which packets to discard can help to prevent congestion collapse.
- The key question for a router drowning in packets is which packets to drop.
- The preferred choice may depend on the type of applications that use the network.
- For a file transfer, an old packet is worth more than a new one. This is because dropping packet 6 and keeping packets 7 through 10, for example, will only force the receiver to do more work to buffer data that it cannot yet use.

96

Load Shedding

- In contrast, for real-time media, a new packet is worth more than an old one. This is because packets become useless if they are delayed and miss the time at which they must be played out to the user.
- The former policy (old is better than new) is often called wine and the latter (new is better than old) is often called milk because most people would rather drink new milk and old wine than the alternative.
- To implement an intelligent discard policy, applications (senders) must mark their packets to indicate to the network how important they are.
- Then, when packets have to be discarded, routers can first drop packets from the least important class, then the next most important class, and so on.

97

Quality of Service

- The techniques we looked at in the previous discussions are designed to reduce congestion and improve network performance.
- However, there are applications (and customers) that demand stronger performance guarantees from the network than ‘‘the best that could be done under the circumstances.’’
- Multimedia applications in particular, often need a maximum throughput and minimum latency to work.
- An easy solution to provide good quality of service is to build a network with enough capacity for whatever traffic will be thrown at it. The name for this solution is overprovisioning.

98

Quality of Service

- To some extent, the telephone system is over-provisioned because it is rare to pick up a telephone and not get a dial tone instantly.
- There is simply so much capacity available that demand can almost always be met.
- The trouble with this solution is that it is expensive. It is basically solving a problem by throwing money at it.
- Moreover, overprovisioning is based on expected traffic.
- All bets are off, if the traffic pattern changes too much.

99

Quality of Service

Four issues must be addressed to ensure quality of service:

1. What applications need from the network.
2. How to regulate the traffic that enters the network.
3. How to reserve resources at routers to guarantee performance.
4. Whether the network can safely accept more traffic.

- No single technique deals efficiently with all these issues. Instead, a variety of techniques have been developed for use at the network (and transport) layer.
- Practical quality-of-service solutions combine multiple techniques.

100

Flow of data

- A stream of packets from a source to a destination is called a flow (Clark, 1988).
- A flow might be all the packets of a connection in a connection-oriented network, or all the packets sent from one process to another process in a connectionless network.
- The needs of each flow can be characterized by four primary parameters: **bandwidth, delay, jitter, and loss**.
- Together, these determine the QoS (Quality of Service) the flow requires.

102

Application Requirements

Application	Bandwidth	Delay	Jitter	Loss
Email	Low	Low	Low	Medium
File sharing	High	Low	Low	Medium
Web access	Medium	Medium	Low	Medium
Remote login	Low	Medium	Medium	Medium
Audio on demand	Low	Low	High	Low
Video on demand	High	Low	High	Low
Telephony	Low	High	High	Low
Videoconferencing	High	High	High	Low

How stringent the quality-of-service requirements are.

103

Categories of QoS and Examples

- To accommodate a variety of applications, networks may support different categories of QoS.

1. Constant bit rate
   - Telephony
2. Real-time variable bit rate
   - Compressed videoconferencing
3. Non-real-time variable bit rate
   - Watching a movie on demand
4. Available bit rate
   - File transfer

104

Traffic Shaping

- Traffic shaping is about regulating the average rate (and burstiness) of data transmission.
- In contrast, the sliding window protocols we studied earlier limit the amount of data in transit at once, not the rate at which it is sent.
- When a connection is set up, the user and the subnet (i.e., the customer and the carrier) agree on a certain traffic pattern (i.e., shape) for that circuit.
- Sometimes this is called a service level agreement.

105

Traffic Shaping

- As long as the customer fulfils her part of the bargain and only sends packets according to the agreed-on contract, the carrier promises to deliver them all in a timely fashion.
- Traffic shaping reduces congestion and thus helps the carrier live up to its promise.
- Such agreements are not so important for file transfers but are of great importance for real-time data, such as audio and video connections, which have stringent quality-of-service requirements.

106

Traffic Shaping

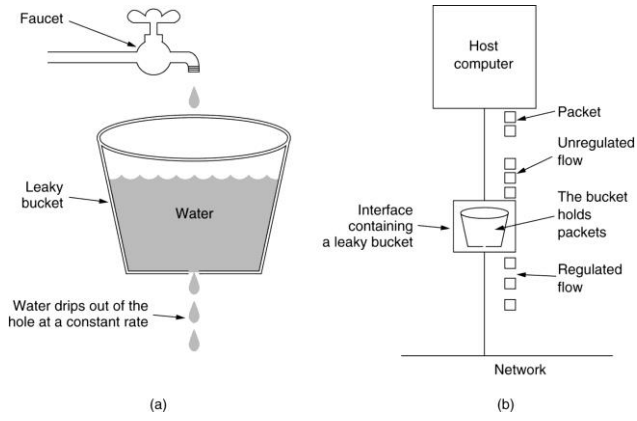
- In effect, with traffic shaping the customer says to the carrier: My transmission pattern will look like this; can you handle it?
- If the carrier agrees, the issue arises of how the carrier can tell if the customer is following the agreement and what to do if the customer is not.
- Monitoring a traffic flow is called **traffic policing**. Agreeing to a traffic shape and policing it afterward are easier with virtual-circuit subnets than with datagram subnets.
- However, even with datagram subnets, the same ideas can be applied to transport layer connections.

107

The Leaky Bucket Algorithm

- Imagine a bucket with a small hole in the bottom, as illustrated in Fig. (a) on next slide. No matter the rate at which water enters the bucket, the outflow is at a constant rate, ρ, when there is any water in the bucket, and zero when the bucket is empty.
- Also, once the bucket is full, any additional water entering it spills over the sides and is lost (i.e., does not appear in the output stream under the hole).
- The same idea can be applied to packets, as shown in Fig. (b) on next slide.

The Leaky Bucket Algorithm



(a) A leaky bucket with water. (b) a leaky bucket with packets.

The Leaky Bucket Algorithm

- Conceptually, each host is connected to the network by an interface containing a leaky bucket, that is, a finite internal queue. If a packet arrives at the queue when it is full, the packet is discarded.
- It was first proposed by Turner (1986) and is called the leaky bucket algorithm.
- In fact, it is nothing other than a single-server queueing system with constant service time.
- The host is allowed to put one packet per clock tick onto the network. This can be enforced by the interface card or by the operating system.

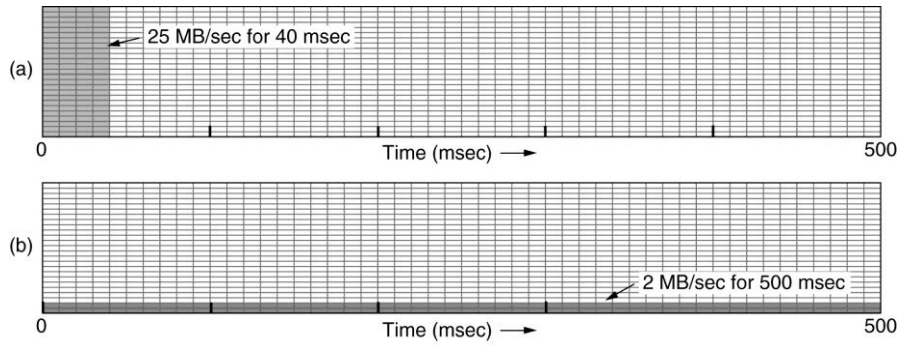
110

The Leaky Bucket Algorithm

- This mechanism turns an uneven flow of packets from the user processes inside the host into an even flow of packets onto the network, smoothing out bursts and greatly reducing the chances of congestion.
- When the packets are all the same size (e.g., ATM cells), this algorithm can be used as described.
- However, when variable-sized packets are being used, it is often better to allow a fixed number of bytes per tick, rather than just one packet.
- Thus, if the rule is 1024 bytes per tick, a single1024-byte packet can be admitted on a tick, two 512-byte packets, four 256-byte packets, and so on. If the residual byte count is too low, the next packet must wait until the next tick.

111

The Leaky Bucket Algorithm

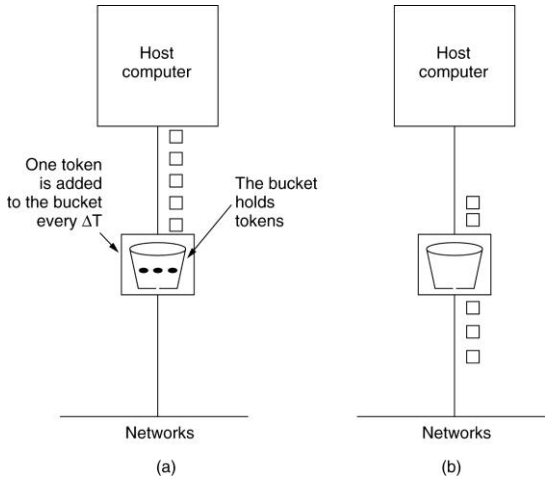


(a) Input to a leaky bucket. (b) Output from a leaky bucket.

The Token Bucket Algorithm

- The leaky bucket algorithm enforces a rigid output pattern at the average rate, no matter how bursty the traffic is.
- For many applications, it is better to allow the output to speed up somewhat when large bursts arrive, so a more flexible algorithm is needed, preferably one that never loses data.
- One such algorithm is the token bucket algorithm.
- In this algorithm, the leaky bucket holds tokens, generated by a clock at the rate of one token every ΔT sec.
- For a packet to be transmitted, it must capture and destroy one token.

The Token Bucket Algorithm



(a) Before. (b) After.

The Token Bucket Algorithm

- The token bucket algorithm provides a different kind of traffic shaping than that of the leaky bucket algorithm.
- The leaky bucket algorithm does not allow idle hosts to save up permission to send large bursts later.
- The token bucket algorithm does allow saving, upto the maximum size of the bucket, *n*.
- This property means that bursts of upto *n* packets can be sent at once, allowing some burstiness in the output stream and giving faster response to sudden bursts of input.

The Token Bucket Algorithm

- Here, too, a minor variant is possible, in which each token represents the right to send not one packet, but *k* bytes.
- A packet can only be transmitted if enough tokens are available to cover its length in bytes. Fractional tokens are kept for future use.
- Essentially what the token bucket does is allow bursts, but up to a regulated maximum length.

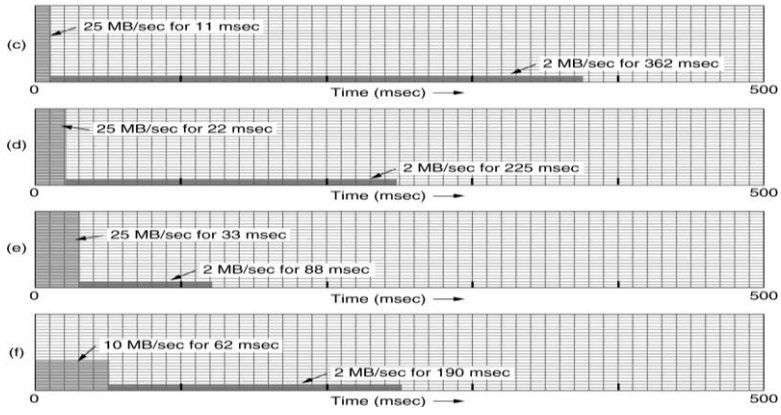
116

The Token Bucket Algorithm

- Look at Fig. (c) (on the next slide) for example. Here we have a token bucket with a capacity of 250 KB.
- Tokens arrive at a rate allowing output at 2MB/sec. Assuming the token bucket is full when the 1-MB burst arrives, the bucket can drain at the full 25 MB/sec for about 11 msec.
- Then it has to cut back to 2MB/sec until the entire input burst has been sent.

117

The Token Bucket Algorithm



Output from a token bucket with capacities of
(c) 250 KB, (d) 500 KB, (e) 750 KB,
(f) Output from a 500KB token bucket feeding a 10-MB/sec leaky bucket.

The Token Bucket Algorithm

- Calculating the length of the maximum rate burst is slightly tricky.
- If we call the burst length *S* sec, the token bucket capacity *C* bytes, the token arrival rate *ρ* bytes/sec, and the maximum output rate *M* bytes/sec.
- We see that an output burst contains a maximum of *C*+ *ρS* bytes.
- We also know that the number of bytes in a maximum-speed burst of length *S* seconds is *MS*.
- Hence we have *MS* = *C*+ *ρS*
- We can solve this equation to get S= C/(M- ρ).

Packet Scheduling

- Being able to regulate the shape of the offered traffic is a good start.
- However, to provide a performance guarantee, we must reserve sufficient resources along the route that the packets take through the network.
- **Algorithms that allocate router resources among the packets of a flow and between competing flows are called packet scheduling algorithms**.
- Kinds of resources can potentially be reserved for different flows:
  1. Bandwidth.
  2. Buffer space.
  3. CPU cycles.

121

Packet Scheduling

- Packet scheduling algorithms allocate bandwidth and other router resources by determining which of the buffered packets to send on the output line next.
- Each router buffers packets in a queue for each output line until they can be sent, and they are sent in the same order that they arrived.
- This algorithm is known as FIFO(First-In First-Out), or equivalently FCFS(First-Come First-Serve).
- FIFO routers usually drop newly arriving packets when the queue is full.
- Since the newly arrived packet would have been placed at the end of the queue, this behavior is called tail drop.

122

Packet Scheduling

- FIFO scheduling is simple to implement, but it is not suited to providing good quality of service because when there are multiple flows, one flow can easily affect the performance of the other flows.
- Processing packets in the order of their arrival means that the aggressive sender can hog most of the capacity of the routers its packets traverse, starving the other flows and reducing their quality of service.
- Many packet scheduling algorithms have been devised that provide stronger isolation between flows and thwart attempts at interference.
- One of the first ones was the Fair Queueing algorithm devised by Nagle (1987).

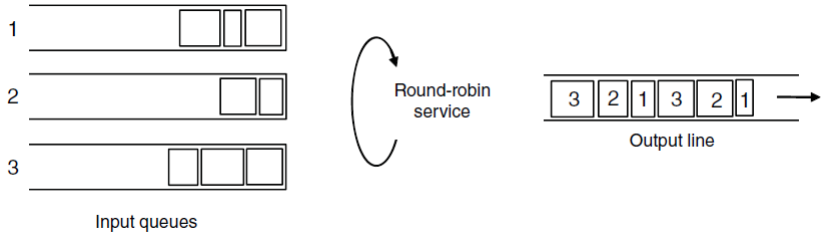
123

Fair Queueing

- The essence of this algorithm is that routers have separate queues, one for each flow for a given output line.
- When the line becomes idle, the router scans the queues round-robin, as shown in Fig. on the next slide.
- It then takes the first packet on the next queue. In this way, with *n* hosts competing for the output line, each host gets to send one out of every *n* packets.
- It is fair in the sense that all flows get to send packets at the same rate.
- Sending more packets will not improve this rate.

124

Fair Queueing



Round-robin Fair Queuing

125

Fair Queueing

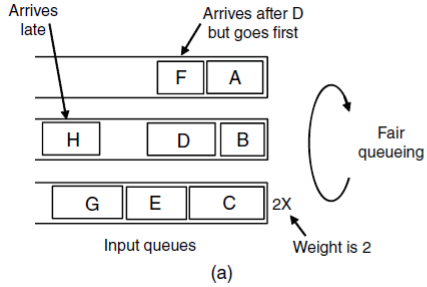
- Although a start, the algorithm has a flaw: it gives more bandwidth to hosts that use large packets than to hosts that use small packets.
- Demers et al. (1990) suggested an improvement in which the round-robin is done in such a way as to simulate a byte-by-byte round-robin, instead of a packet-by-packet round-robin.
- The trick is to compute a virtual time that is the number of the round at which each packet would finish being sent.
- Each round drains a byte from all of the queues that have data to send.

126

Fair Queueing

- The packets are then sorted in order of their finishing times and sent in that order.
- This algorithm and an example of finish times for packets arriving in three flows are illustrated in Fig. on next slide.

Weighted Fair Queueing



(a)

Packet	Arrival time	Length	Finish time	Output order
A	0	8	8	1
B	5	6	11	3
C	5	10	10	2
D	8	9	20	7
E	8	8	14	4
F	10	6	16	5
G	11	10	19	6
H	20	8	28	8

(b)

(a) Weighted Fair Queuing.

(b) Finishing times for the packets.

Weighted Fair Queueing

- One shortcoming of Fair Queueing algorithm in practice is that it gives all hosts the same priority.
- In many situations, it is desirable to give, for example, video servers more bandwidth than, say, file servers.
- This is easily possible by giving the video server two or more bytes per round.
- This modified algorithm is called WFQ(Weighted Fair Queueing).
- Letting the number of bytes per round be the weight of a flow, W, we can now give the formula for computing the finish time:

$$F_i = \max(A_i, F_{i-1}) + L_i/W$$

129

Weighted Fair Queueing

- Where A_i is the arrival time, F_i is the finish time, and L_i is the length of packet i.
- The bottom queue of Fig. (a) on previous slide has a weight of 2, so its packets are sent more quickly as you can see in the finish times given in Fig. (b).

130

Jitter Control

- Packets might carry timestamps and be sent in timestamp order.
- Clark et al. (1992) describe a design in which the timestamp records how far the packet is behind or ahead of schedule as it is sent through a sequence of routers on the path.
- Packets that have been queued behind other packets at a router will tend to be behind schedule, and the packets that have been serviced first will tend to be ahead of schedule.
- Sending packets in order of their timestamps has the beneficial effect of speeding up slow packets while at the same time slowing down fast packets.
- The result is that all packets are delivered by the network with a more consistent delay.

131

Admission Control

- QoS guarantees are established through the process of admission control.
- We first saw admission control used to control congestion, which is a performance guarantee, albeit a weak one.
- The guarantees we are considering now are stronger, but the model is the same.
- The user offers a flow with an accompanying QoS requirement to the network.
- The network then decides, whether to accept or reject the flow based on its capacity and the commitments it has made to other flows.

132

Admission Control

- If it accepts, the network reserves capacity in advance at routers to guarantee QoS when traffic is sent on the new flow.
- The reservations must be made at all of the routers along the route that the packets take through the network.
- Any routers on the path without reservations might become congested, and a single congested router can break the QoS guarantee.
- Given a path, the decision to accept or reject a flow is not a simple matter of comparing the resources (bandwidth, buffers, cycles) requested by the flow with the router's excess capacity in those three dimensions.

133

Admission Control

- It is a little more complicated than that. To start with, although some applications may know about their bandwidth requirements, few know about buffers or CPU cycles.
- So at the minimum, a different way is needed to describe flows and translate this description to router resources.
- Because many parties may be involved in the flow negotiation (the sender, the receiver, and all the routers along the path between them), flows must be described accurately in terms of specific parameters that can be negotiated.
- A set of such parameters is called a flow specification.

134

Admission Control

- Typically, the sender (e.g., the video server) produces a flow specification proposing the parameters it would like to use.
- As the specification propagates along the route, each router examines it and modifies the parameters as need be.
- The modifications can only reduce the flow, not increase it (e.g., a lower data rate, not a higher one).
- When it gets to the other end, the parameters can be established.
- As an example of what can be in a flow specification, consider the example of Fig. on next slide.

135

Admission Control

Parameter	Unit
Token bucket rate	Bytes/sec
Token bucket size	Bytes
Peak data rate	Bytes/sec
Minimum packet size	Bytes
Maximum packet size	Bytes

An example flow specification

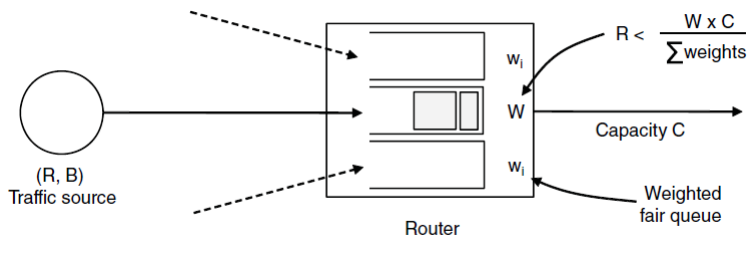
136

Admission Control

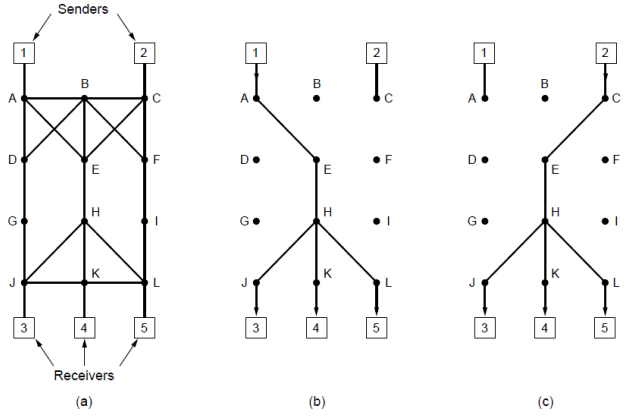
- An interesting question is how a router turns a flow specification into a set of specific resource reservations.
- At first glance, it might appear that if a router has a link that runs at, say, 1 Gbps and the average packet is 1000 bits, it can process 1 million packets/sec.
- This observation is not the case, though, because there will always be idle periods on the link due to statistical fluctuations in the load.
- If the link needs every bit of capacity to get its work done, idling for even a few bits creates a backlog it can never get rid of.

Admission Control

Bandwidth and delay guarantees with token buckets and WFQ.



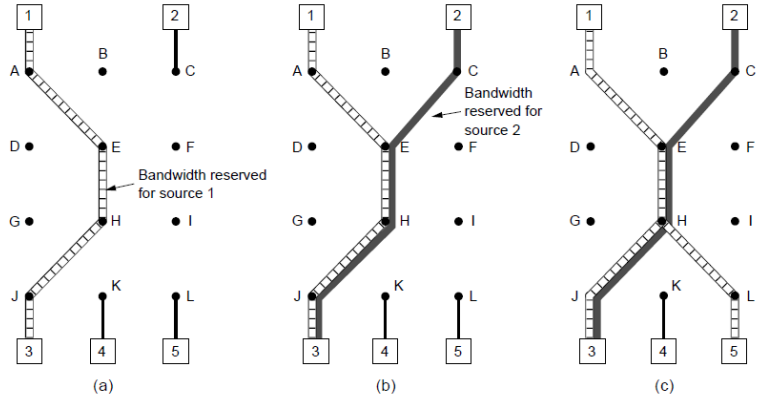
Integrated Services (1)



(a) A network. (b) The multicast spanning tree for host 1. (c) The multicast spanning tree for host 2.

139

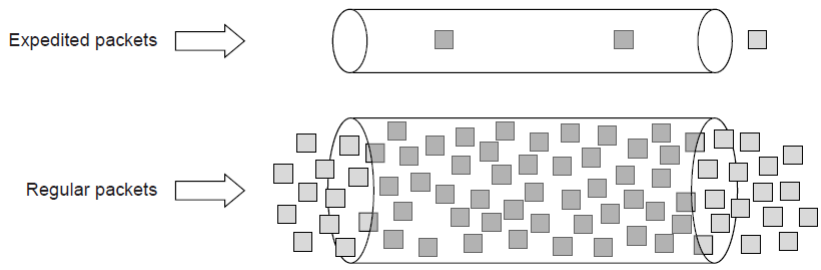
Integrated Services (2)



(a) Host 3 requests a channel to host 1. (b) Host 3 then requests a second channel, to host 2. (c) Host 5 requests a channel to host 1.

140

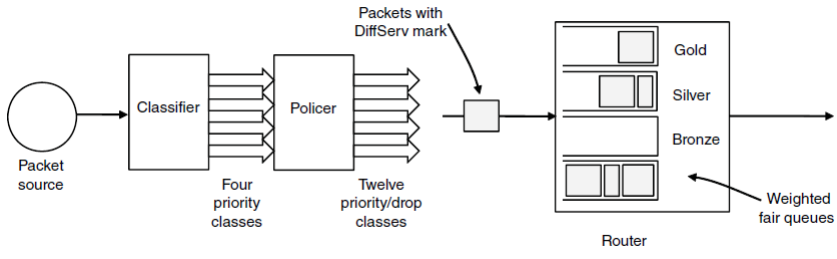
Differentiated Services (1)



Expedited packets experience a traffic-free network

141

Differentiated Services (2)



A possible implementation of assured forwarding

142

Internetworking

- Until now, we have implicitly assumed that there is a single homogeneous network, with each machine using the same protocol in each layer.
- Unfortunately, this assumption is wildly optimistic.
- Many different networks exist, including PANs, LANs, MANs, and WANs.
- Numerous protocols are in widespread use across these networks in every layer.
- Now, we will look at the issues that arise when two or more networks are connected to form an **internetwork**, or more simply an **internet**.

143

Internetworking

- Some pundits speculate that the multiplicity of technologies will go away as soon as everyone realizes how wonderful [fill in your favorite network] is.
- Do not count on it. History shows this to be wishful thinking. Heterogeneity is here to stay.
- If there will always be different networks, it would be simpler if we did not need to interconnect them. This also is unlikely.
- Bob Metcalfe postulated that the value of a network with N nodes is the number of connections that may be made between the nodes, or N^2 (Gilder, 1993).
- This means that large networks are much more valuable than small networks because they allow many more connections, so there always will be an incentive to combine smaller networks.

144

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

Some of the many ways networks can differ

146

How Networks Can Be Connected

- There are two basic choices for connecting different networks:
  - build devices that translate or convert packets from each kind of network into packets for each other network, or,
  - solve the problem by adding a layer of indirection and building a common layer on top of the different networks.
- In either case, the devices are placed at the boundaries between networks.
- Early on, Cerf and Kahn (1974) argued for a common layer to hide the differences of existing networks.
- This approach has been tremendously successful, and the layer they proposed was eventually separated into the TCP and IP protocols.
- Almost four decades later, IP is the foundation of the modern Internet.

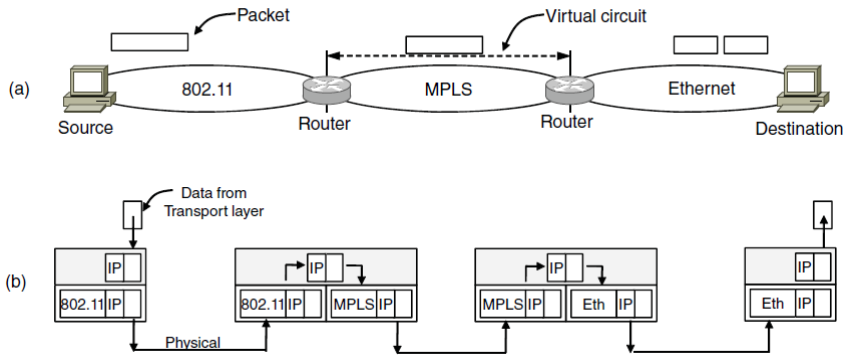
147

How Networks Can Be Connected

- IP provides a universal packet format that all routers recognize and that can be passed through almost every network.
- Let us first explore at a high level how interconnection with a common network layer can be used to interconnect dissimilar networks.
- An internet comprised of 802.11, MPLS, and Ethernet networks is shown in Fig. 5-39(a).
- Suppose that the source machine on the 802.11 network wants to send a packet to the destination machine on the Ethernet network.
- Since these technologies are different, and they are further separated by another kind of network (MPLS), some added processing is needed at the boundaries between the networks.

148

How Networks Can Be Connected



(a) A packet crossing different networks.

(b) Network and link layer protocol processing.

149

How Networks Can Be Connected

- Internetworking has been very successful at building large networks, but it only works when there is a common network layer.
- There have, in fact, been many network protocols over time. Getting everybody to agree on a single format is difficult when companies perceive it to their commercial advantage to have a proprietary format that they control.
- Examples besides IP, which is now the near-universal network protocol, were IPX, SNA, and AppleTalk.
- None of these protocols are still in widespread use, but there will always be other protocols.
- The most relevant example now is probably IPv4 and IPv6. While these are both versions of IP, they are not compatible

150

How Networks Can Be Connected

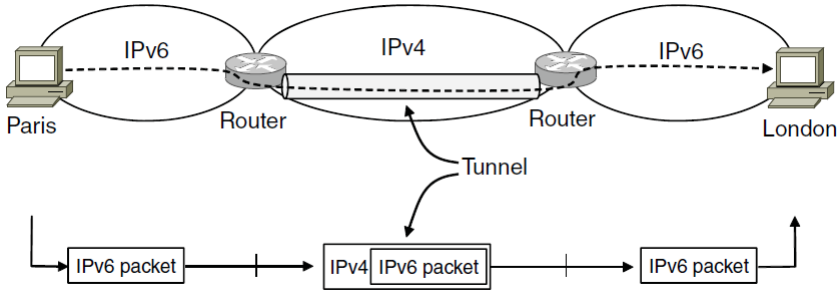
- A router that can handle multiple network protocols is called a **multiprotocol router**.
- It must either translate the protocols, or leave connection for a higher protocol layer. Neither approach is entirely satisfactory.
- Connection at a higher layer, say, by using TCP, requires that all the networks implement TCP (which may not be the case). Then, it limits usage across the networks to applications that use TCP (which does not include many real-time applications).
- The alternative is to translate packets between the networks. However, unless the packet formats are close relatives with the same information fields, such conversions will always be incomplete and often doomed to failure.
- For example, IPv6 addresses are 128 bits long. They will not fit in a 32-bit IPv4 address field, no matter how hard the router tries.
- Getting IPv4 and IPv6 to run in the same network has proven to be a major obstacle to the deployment of IPv6.

151

Tunneling

- Handling the general case of making two different networks interwork is exceedingly difficult.
- However, when the source and destination hosts are on the same type of network, but there is a different network in between.
- As an example, think of an international bank with an IPv6 network in Paris, an IPv6 network in London and connectivity between the offices via the IPv4 Internet.
- This situation is shown in Fig. on the next slide.

Tunneling (1)



Tunneling a packet from Paris to London.

Tunneling

- The solution to this problem is a technique called tunneling.
- To send an IP packet to a host in the London office, a host in the Paris office constructs the packet containing an IPv6 address in London, and sends it to the multiprotocol router that connects the Paris IPv6 network to the IPv4 Internet.
- When this router gets the IPv6 packet, it encapsulates the packet with an IPv4 header addressed to the IPv4 side of the multiprotocol router that connects to the London IPv6 network.
- That is, the router puts a (IPv6) packet inside a (IPv4) packet.

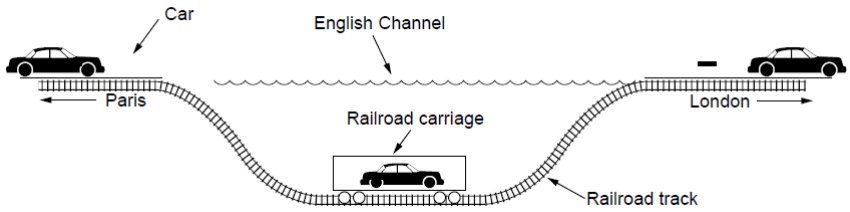
154

Tunneling

- When this wrapped packet arrives, the London router removes the original IPv6 packet and sends it onward to the destination host.
- The path through the IPv4 Internet can be seen as a big tunnel extending from one multiprotocol router to the other.
- The IPv6 packet just travels from one end of the tunnel to the other, snug in its nice box.
- It does not have to worry about dealing with IPv4 at all. Neither do the hosts in Paris or London.
- Only the multiprotocol routers have to understand both IPv4 and IPv6 packets.
- In effect, the entire trip from one multiprotocol router to the other is like a hop over a single link.

155

Tunneling (2)



Tunneling a car from France to England

156

Tunneling

- Tunneling is widely used to connect isolated hosts and networks using other networks.
- The network that results is called an overlay since it has effectively been overlaid on the base network.
- The disadvantage of tunneling is that none of the hosts on the network that is tunneled over can be reached because the packets cannot escape in the middle of the tunnel.
- However, this limitation of tunnels is turned into an advantage with VPNs(Virtual Private Networks).
- A VPN is simply an overlay that is used to provide a measure of security.

157

Packet Fragmentation

Each network or link imposes some maximum size on its packets. These limits have various causes, among them:

1. Hardware (e.g., the size of an Ethernet frame).
2. Operating system (e.g., all buffers are 512 bytes).
3. Protocols (e.g., the number of bits in the packet length field).
4. Compliance with (inter)national standard.
5. Desire to reduce error-induced retransmissions
6. Desire to prevent packet occupying channel too long.

158

Packet Fragmentation

- Because of this, network designers are not free to choose any old maximum packet size they wish.
- Maximum payloads for some common technologies are 1500 bytes for Ethernet and 2272 bytes for 802.11. IP is more generous, allows for packets as big as 65,515 bytes.
- Hosts usually prefer to transmit large packets because this reduces packet overheads such as bandwidth wasted on header bytes.
- An obvious internetworking problem appears when a large packet wants to travel through a network whose maximum packet size is too small.

159

Packet Fragmentation

- One solution is to make sure the problem does not occur in the first place.
- However, this is easier said than done. A source does not usually know the path a packet will take through the network to a destination, so it certainly does not know how small packets must be to get there.
- This packet size is called the Path **MTU (Path Maximum Transmission Unit)**.
- Even if the source did know the path MTU, packets are routed independently in a connectionless network such as the Internet.
- This routing means that paths may suddenly change, which can unexpectedly change the path MTU.

160

Packet Fragmentation

- The alternative solution to the problem is to allow routers to break up packets into fragments, sending each fragment as a separate network layer packet.
- However, converting a large object into small fragments is considerably easier than the reverse process.
- Packet-switching networks, too, have trouble putting the fragments back together again.
- Two opposing strategies exist for recombining the fragments back into the original packet.
- The first strategy is to make fragmentation caused by a ‘‘small packet’’ network transparent to any subsequent networks through which the packet must pass on its way to the ultimate destination.

161

Transparent Packet Fragmentation

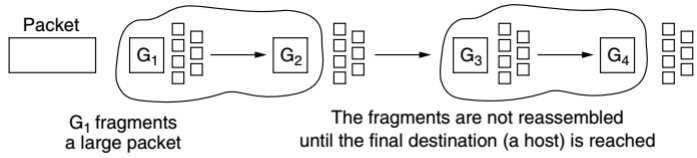
Transparent fragmentation

- In this way, passage through the small-packet network is made transparent. Subsequent networks are not even aware that fragmentation has occurred.
- Transparent fragmentation is straightforward but has some problems.
- For one thing, the exit router must know when it has received all the pieces, so either a count field or an ‘‘end of packet’’ bit must be provided.

Transparent Packet Fragmentation

- Also, because all packets must exit via the same router so that they can be reassembled, the routes are constrained.
- By not allowing some fragments to follow one route to the ultimate destination and other fragments a disjoint route, some performance may be lost.
- More significant is the amount of work that the router may have to do. It may need to buffer the fragments as they arrive, and decide when to throw them away if not all of the fragments arrive.
- Some of this work may be wasteful, too, as the packet may pass through a series of small packet networks and need to be repeatedly fragmented and reassembled.

Nontransparent Packet Fragmentation



(b)

Non transparent fragmentation

- The other fragmentation strategy is to refrain from recombining fragments at any intermediate routers.
- Once a packet has been fragmented, each fragment is treated as though it were an original packet.
- The routers pass the fragments, as shown in Fig. (b), and reassembly is performed only at the destination host.

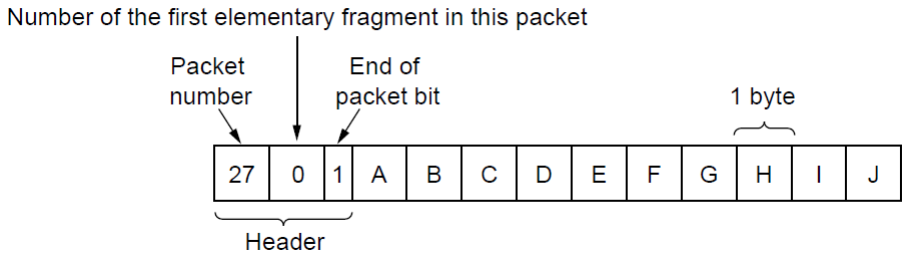
164

Nontransparent Packet Fragmentation

- The main advantage of nontransparent fragmentation is that it requires routers to do less work.
- IP works this way. A complete design requires that the fragments be numbered in such a way that the original data stream can be reconstructed.
- The design used by IP is to give every fragment a packet number (carried on all packets), an absolute byte offset within the packet, and a flag indicating whether it is the end of the packet.
- While simple, this design has some attractive properties. Fragments can be placed in a buffer at the destination in the right place for reassembly, even if they arrive out of order.
- Fragments can also be fragmented if they pass over a network with a yet smaller MTU.

165

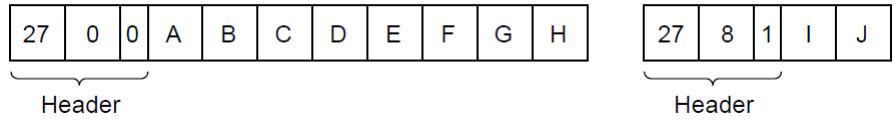
Nontransparent Packet Fragmentation



Fragmentation when the elementary data size is 1 byte. (a) Original packet, containing 10 data bytes.

167

Nontransparent Packet Fragmentation

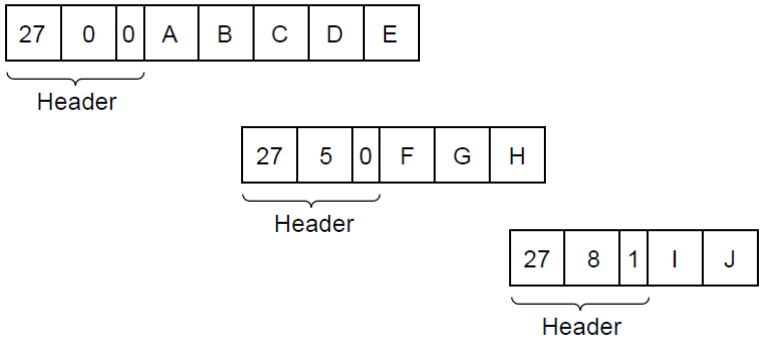


Fragmentation when the elementary data size is 1 byte

(b) Fragments after passing through a network with maximum packet size of 8 payload bytes plus header.

168

Nontransparent Packet Fragmentation



Fragmentation when the elementary data size is 1 byte

(c) Fragments after passing through a size 5 gateway.

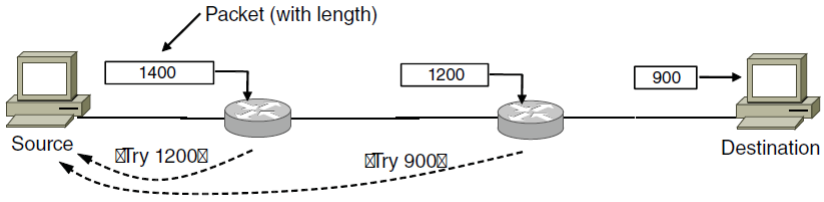
169

Path MTU discovery

- Unfortunately, this design still has problems. The overhead can be higher than with transparent fragmentation because fragment headers are now carried over some links where they may not be needed.
- But the real problem is the existence of fragments in the first place. Kent and Mogul (1987) argued that fragmentation is detrimental to performance because, as well as the header overheads, a whole packet is lost if any of its fragments are lost, and because fragmentation is more of a burden for hosts than was originally realized.
- This leads us back to the original solution of getting rid of fragmentation in the network, the strategy used in the modern Internet.
- The process is called **path MTU discovery** (Mogul and Deering, 1990).

170

Path MTU discovery



Path MTU Discovery

171

Path MTU discovery

- Each IP packet is sent with its header bits set to indicate that no fragmentation is allowed to be performed.
- If a router receives a packet that is too large, it generates an error packet, returns it to the source, and drops the packet.
- This is shown in Fig. 5-44.
- When the source receives the error packet, it uses the information inside to refragment the packet into pieces that are small enough for the router to handle.
- If a router further down the path has an even smaller MTU, the process is repeated.
- The advantage of path MTU discovery is that the source now knows what length packet to send.

172

Path MTU discovery

- If the routes and path MTU change, new error packets will be triggered and the source will adapt to the new path.
- However, fragmentation is still needed between the source and the destination unless the higher layers learn the path MTU and pass the right amount of data to IP.
- The disadvantage of path MTU discovery is that there may be added startup delays simply to send a packet.
- More than one round-trip delay may be needed to probe the path and find the MTU before any data is delivered to the destination.

173

The Network Layer in the Internet

- It is now time to discuss the network layer of the Internet in detail.
- But before getting into specifics, it is worth taking a look at the principles that drove its design in the past and made it the success that it is today.
- These principles are enumerated and discussed in RFC 1958, which is well worth reading (and should be mandatory for all protocol designers).
- This RFC draws heavily on ideas put forth by Clark (1988) and Saltzer et al. (1984).
- The next few slides list the top 10 principles (from most important to least important).

174

The Network Layer Principles

1. Make sure it works
   - Do not finalize the design or standard until multiple prototypes have successfully communicated with each other.
2. Keep it simple
   - When in doubt, use the simplest solution. Put in modern terms: fight features. If a feature is not absolutely essential, leave it out, especially if the same effect can be achieved by combining other features.
3. Make clear choices
   - If there are several ways of doing the same thing, choose one. Having two or more ways to do the same thing is looking for trouble.

175

The Network Layer Principles

4. Exploit modularity
   - This principle leads directly to the idea of having protocol stacks, each of whose layers is independent of all the other ones.
5. Expect heterogeneity
   - Different types of hardware, transmission facilities, and applications will occur on any large network. To handle them, the network design must be simple, general, and flexible.
6. Avoid static options and parameters
   - If parameters are unavoidable (e.g., maximum packet size), it is best to have the sender and receiver negotiate a value rather than defining fixed choices.

176

The Network Layer Principles

7. Look for good design (need not be perfect)
   - Often, the designers have a good design but it cannot handle some weird special case.
   - Rather than messing up the design, the designers should go with the good design and put the burden of working around it on the people with the strange requirements.
8. Strict sending, tolerant receiving
   - In other words, send only packets that rigorously comply with the standards, but expect incoming packets that may not be fully conformant and try to deal with them.

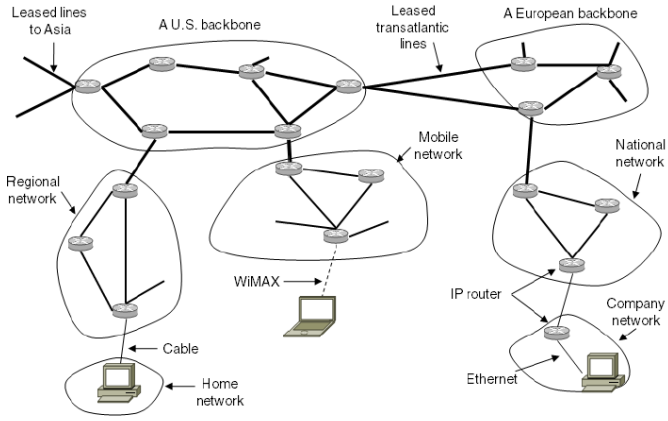
177

The Network Layer Principles

9. Think about scalability
   - If the system is to handle millions of hosts and billions of users effectively, no centralized databases of any kind are tolerable and load must be spread as evenly as possible over the available resources.
10. Consider performance and cost
    - If a network has poor performance or outrageous costs, nobody will use it.

178

The Network Layer in the Internet

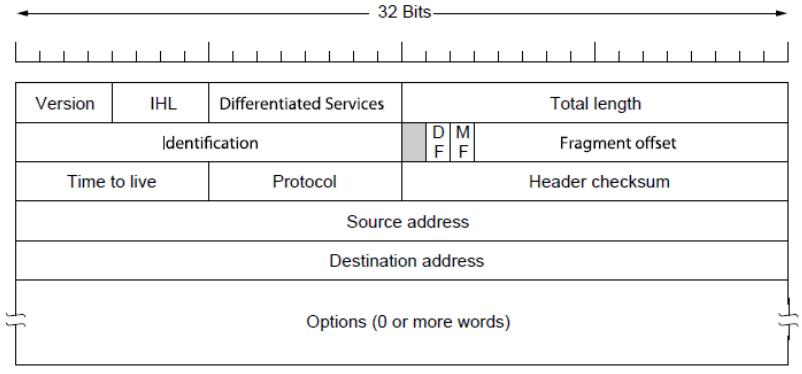


The Internet is an interconnected collection of many networks.

The Network Layer in the Internet

- An appropriate place to start our study of the network layer in the Internet is with the format of the IP datagrams themselves.
- An IPv4 datagram consists of a header part and a body or payload part.
- The header has a 20-byte fixed part and a variable-length optional part. The header format is shown in next slide.
- The bits are transmitted from left to right and top to bottom, with the high-order bit of the Version field going first.
- This is a ‘‘big-endian’’ network byte order. On little-endian machines, such as Intel x86 computers, a software conversion is required on both transmission and reception.)

The IP Version 4 Protocol



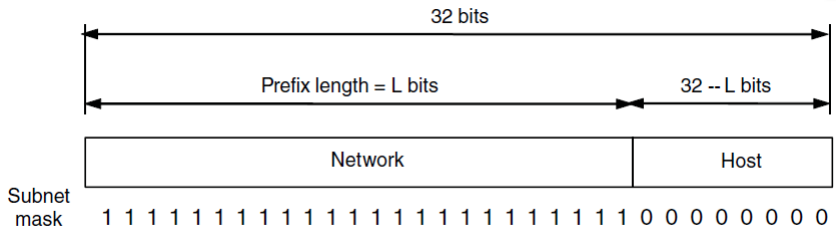
The IPv4 (Internet Protocol) header.

The IP Version 4 Protocol

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

Some of the IP options.

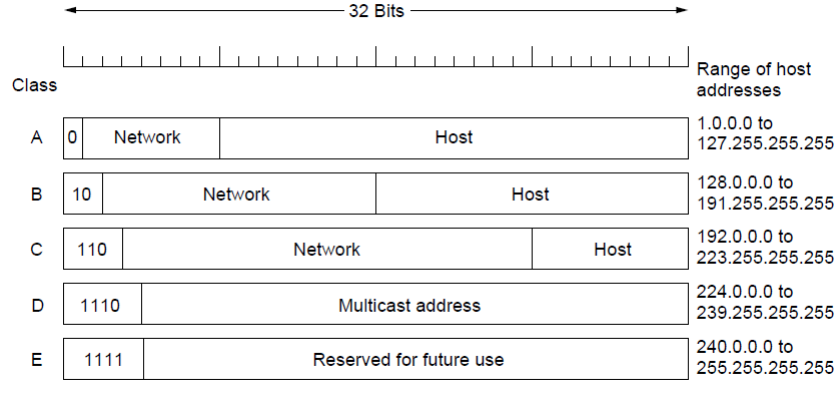
IP Addresses (1)



An IP prefix.

184

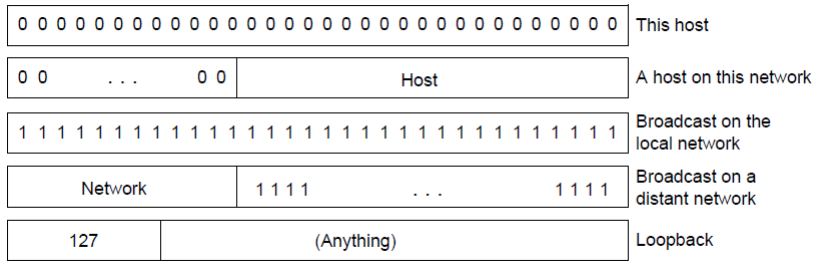
IP Addresses (6)



IP address formats

185

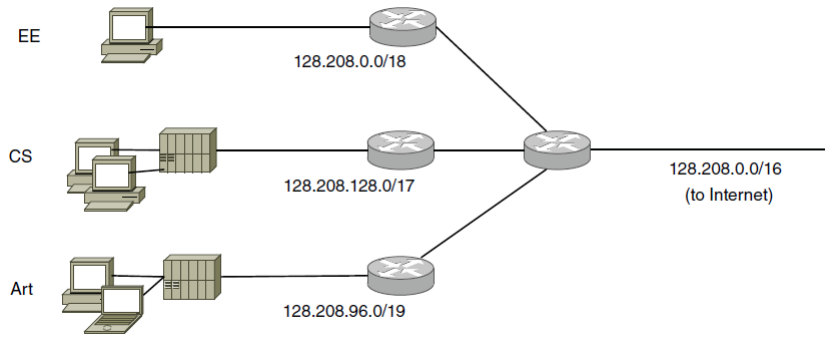
IP Addresses (7)



Special IP addresses

186

IP Addresses (2)



Splitting an IP prefix into separate networks with subnetting.

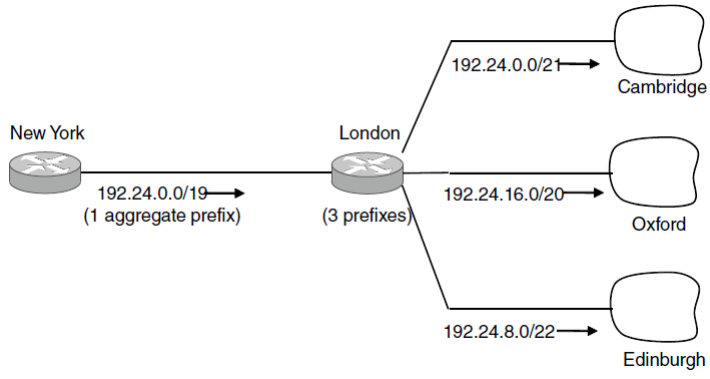
187

IP Addresses (3)

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	194.24.12/22
Oxford	194.24.16.0	194.24.31.255	4096	194.24.16.0/20

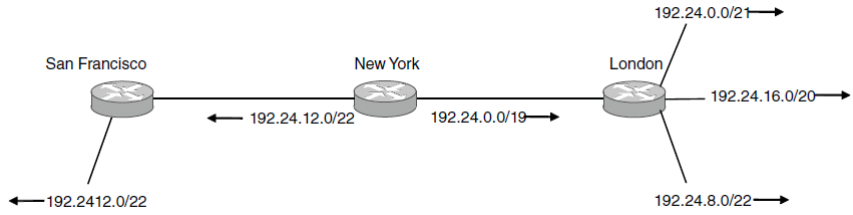
A set of IP address assignments

IP Addresses (4)



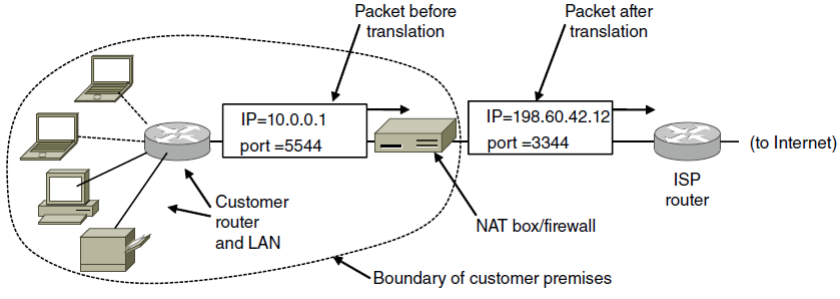
Aggregation of IP prefixes

IP Addresses (5)



Longest matching prefix routing at the New York router.

IP Addresses (8)



Placement and operation of a NAT box.

IP Version 6 (IPv6)

- IPv4 has been in heavy use for decades. It has worked extremely well, as demonstrated by the exponential growth of the Internet.
- Unfortunately, IP has become a victim of its own popularity: it is close to running out of addresses.
- Even with CIDR and NAT using addresses more sparingly, the last IPv4 addresses are expected to be assigned by ICANN very soon.
- This looming disaster was recognized almost two decades ago, and it sparked a great deal of discussion and controversy within the Internet community about what to do about it.

192

IP Version 6 (IPv6)

- The only long-term solution is to move to larger addresses. IPv6 is a replacement design that does just that.
- It uses 128-bit addresses; a shortage of these addresses is not likely any time in the foreseeable future.
- However, IPv6 has proved very difficult to deploy. It is a different network layer protocol that does not really interwork with IPv4, despite many similarities.
- Also, companies and users are not really sure why they should want IPv6 in any case.
- The result is that IPv6 is deployed and used on only a tiny fraction of the Internet (estimates are 1%) despite having been an Internet Standard since 1998.

193

IP Version 6 Goals

- Seeing these problems on the horizon, in 1990 IETF started work on a new version of IP, one that would never run out of addresses, would solve a variety of other problems, and be more flexible and efficient as well. Its major goals were:
  - Support billions of hosts, even with inefficient address allocation.
  - Reduce the size of the routing tables.
  - Simplify the protocol, to allow routers to process packets faster.
  - Provide better security (authentication and privacy).
  - Pay more attention to the type of service, particularly for real-time data.
  - Aid multicasting by allowing scopes to be specified.
  - Make it possible for a host to roam without changing its address.
  - Allow the protocol to evolve in the future.
  - Permit the old and new protocols to coexist for years.

194

IP Version 6

- To develop a protocol that met all these requirements, IETF issued a call for proposals and discussion in RFC 1550.
- Twenty-one responses were initially received. By December 1992, seven serious proposals were on the table.
- They ranged from making minor patches to IP, to throwing it out altogether and replacing it with a completely different protocol.
- Three of the better proposals were published in IEEE Network (Deering, 1993; Francis, 1993; and Katz and Ford, 1993).
- After much discussion, revision, and jockeying for position, a modified combined version of the Deering and Francis proposals, by now called SIPP (Simple Internet Protocol Plus) was selected and given the designation IPv6.
- IPv6 meets IETF's goals fairly well. It maintains the good features of IP, discards or deemphasizes the bad ones, and adds new ones where needed.

195

IP Version 6

- In general, IPv6 is not compatible with IPv4, but it is compatible with the other auxiliary Internet protocols, including TCP, UDP, ICMP, IGMP, OSPF, BGP, and DNS, with small modifications being required to deal with longer addresses.
- The main features of IPv6 are discussed below. More information about it can be found in RFCs 2460 through 2466.
- First and foremost, IPv6 has longer addresses than IPv4. They are 128 bits long, which solves the problem that IPv6 set out to solve: providing an effectively unlimited supply of Internet addresses.
- The second major improvement of IPv6 is the simplification of the header. It contains only seven fields (versus 13 in IPv4). This change allows routers to process packets faster and thus improves throughput and delay.

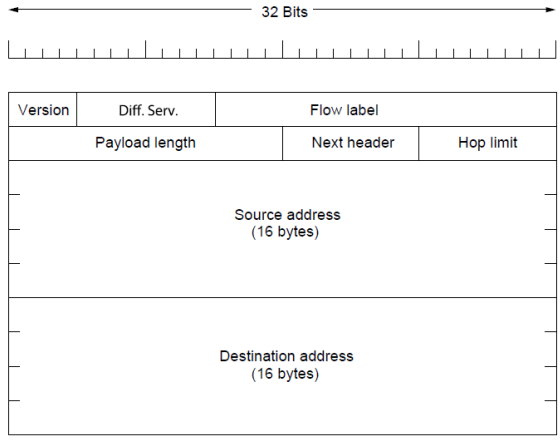
196

IP Version 6

- The third major improvement is better support for options. This change was essential with the new header because fields that previously were required are now optional (because they are not used so often).
- In addition, the way options are represented is different, making it simple for routers to skip over options not intended for them.
- This feature speeds up packet processing time.
- A fourth area in which IPv6 represents a big advance is in security.
- Authentication and privacy are key features of the new IP. These were later retrofitted to IPv4.
- Finally, more attention has been paid to quality of service.

197

IP Version 6 :The Main Header



The IPv6 fixed header (Mandatory).

198

IP Version 6 :The Main Header

- The **Version** field is always 6 for IPv6 (and 4 for IPv4).
- The **Differentiated services** field (originally called Traffic class) is used to distinguish the class of service for packets with different real-time delivery requirements.
- It is used with the differentiated service architecture for quality of service in the same manner as the field of the same name in the IPv4 packet.
- Also, the low-order 2 bits are used to signal explicit congestion indications, again in the same way as with IPv4.
- The **Flow label** field provides a way for a source and destination to mark groups of packets that have the same requirements and should be treated in the same way by the network, forming a pseudo connection.

199

IP Version 6 :The Main Header

- When a packet with a **nonzero Flow label** shows up, all the routers can look it up in internal tables to see what kind of special treatment it requires.
- In effect, flows are an attempt to have it both ways: the flexibility of a datagram network and the guarantees of a virtual-circuit network.
- Each flow for quality of service purposes is designated by the source address, destination address, and flow number.
- This design means that up to 2^{20} flows may be active at the same time between a given pair of IP addresses.
- It is expected that flow labels will be chosen randomly, rather than assigned sequentially starting at 1, so routers are expected to hash them.

200

IP Version 6 :The Main Header

- The **Payload length** field tells how many bytes follow the 40-byte header.
- The name was changed from the IPv4 **Total length** field because the meaning was changed slightly: the 40 header bytes are no longer counted as part of the length (as they used to be).
- This change means the payload can now be 65,535 bytes instead of a mere 65,515 bytes.
- The **Next header** field lets the cat out of the bag. The reason the header could be simplified is that there can be additional (optional) extension headers.
- This field tells which of the (currently) six extension headers, if any, follow this one.
- If this header is the last IP header, the Next header field tells which transport protocol handler (e.g., TCP, UDP) to pass the packet to.

201

IP Version 6 :The Main Header

- The **Hop limit** field is used to keep packets from living forever. It is, in practice, the same as the **Time to live** field in IPv4, namely, a field that is decremented on each hop.
- In theory, in IPv4 it was a time in seconds, but no router used it that way, so the name was changed to reflect the way it is actually used.
- Next come the 16 bytes **Source address** and **Destination address** fields.
- A new notation has been devised for writing 16-byte addresses.
- They are written as eight groups of four hexadecimal digits with colons between the groups, like this:

  8000:0000:0000:0000:0123:4567:89AB:CDEF

  8000::123:4567:89AB:CDEF
- Finally, IPv4 addresses can be written as a pair of colons and an old dotted decimal number, for example:
- ::192.31.20.46

202

IP Version 6 :The Main Header

- It is worthy to compare the IPv4 header with the IPv6 header to see what has been left out in IPv6.
- The IHL field is gone because the IPv6 header has a fixed length.
- The Protocol field was taken out because the Next header field tells what follows the last IP header (e.g., a UDP or TCP segment).
- All the fields relating to fragmentation were removed because IPv6 takes a different approach to fragmentation.
- To start with, all IPv6-conformant hosts are expected to dynamically determine the packet size to use.
- They do this using the path MTU discovery procedure.

203

IP Version 6 :The Main Header

- Also, the minimum-size packet that routers must be able to forward has been raised from 576 to 1280 bytes to allow 1024 bytes of data and many headers.
- Finally, the Checksum field is gone because calculating it greatly reduces performance.
- With the reliable networks now used, combined with the fact that the data link layer and transport layers normally have their own checksums, the value of yet another checksum was deemed not worth the performance price it extracted.
- Removing all these features has resulted in a lean and mean network layer protocol.
- Thus, the goal of IPv6—a fast, yet flexible, protocol with plenty of address space—is met by this design.

204

IP Version 6: Extension Headers

- Some of the missing IPv4 fields are occasionally still needed, so IPv6 introduces the concept of (optional) extension headers.
- These headers can be supplied to provide extra information, but encoded in an efficient way.
- Six kinds of extension headers are defined at present as shown below:

Extension header	Description
Hop-by-hop options	Miscellaneous information for routers
Destination options	Additional information for the destination
Routing	Loose list of routers to visit
Fragmentation	Management of datagram fragments
Authentication	Verification of the sender's identity
Encrypted security payload	Information about the encrypted contents

205

IP Version 6: Extension Headers

- Each one is optional, but if more than one is present they must appear directly after the fixed header, and preferably in the order listed.
- Some of the headers have a fixed format; others contain a variable number of variable-length options.
- For these, each item is encoded as a (Type, Length, Value) tuple. The Type is a 1-byte field telling which option this is.
- The Type values have been chosen so that the first 2 bits tell routers that do not know how to process the option what to do.
- The choices are: skip the option; discard the packet; discard the packet and send back an ICMP packet; and discard the packet but do not send ICMP packets for multicast addresses (to prevent one bad multicast packet from generating millions of ICMP reports).
- The Length is also a 1-byte field. It tells how long the value is (0 to 255 bytes).
- The Value is any information required, up to 255 bytes.

206

IP Version 6: Hop-by-hop Extension Header

- The hop-by-hop header is used for information that all routers along the path must examine.
- So far, one option has been defined: support of datagrams exceeding 64 KB. The format of this header is shown below.
- When it is used, the Payload length field in the fixed header is set to 0.

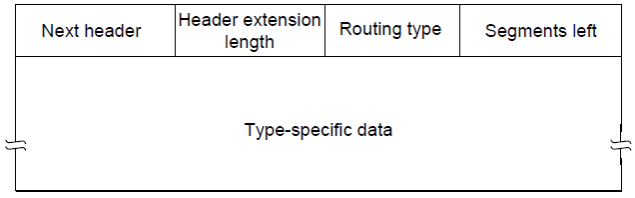
Next header	0	194	4
Jumbo payload length			

The hop-by-hop extension header for large datagrams (jumbograms).

207

IP Version 6: Routing Extension Header

- The **destination options** header is intended for fields that need only be interpreted at the destination host.
- The **routing** header lists one or more routers that must be visited on the way to the destination.
- It is very similar to the IPv4 loose source routing in that all addresses listed must be visited in order, but other routers not listed may be visited in between.
- The format of the routing header is shown below:



The extension header for routing.

208

IP Version 6: Extension Header

- The **fragment header** deals with fragmentation similarly to the way IPv4 does.
- The header holds the datagram identifier, fragment number, and a bit telling whether more fragments will follow.
- In IPv6, unlike in IPv4, only the source host can fragment a packet. Routers along the way may not do this.
- The **authentication header** provides a mechanism by which the receiver of a packet can be sure of who sent it.
- The encrypted security payload makes it possible to encrypt the contents of a packet so that only the intended recipient can read it.

209

Internet Control Protocols

- In addition to IP, which is used for data transfer, the Internet has several companion control protocols that are used in the network layer.
- They include ICMP, ARP, and DHCP. In this section, we will look at each of these in turn, describing the versions that correspond to IPv4 because they are the protocols that are in common use.
- ICMP and DHCP have similar versions for IPv6; the equivalent of ARP is called NDP (Neighbor Discovery Protocol) for IPv6.

210

Internet Control Message Protocol (ICMP)

- The operation of the Internet is monitored closely by the routers. When something unexpected occurs during packet processing at a router, the event is reported to the sender by the **ICMP (Internet Control Message Protocol)**.
- ICMP is also used to test the Internet. About a dozen types of ICMP messages are defined.
- Each ICMP message type is carried encapsulated in an IP packet. The most important ones are listed on next slide

211

Internet Control Message Protocol (ICMP)

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

The principal ICMP message types.

212

Internet Control Message Protocol (ICMP)

- The **DESTINATION UNREACHABLE** message is used when the router cannot locate the destination or when a packet with the DF bit cannot be delivered because a "small-packet" network stands in the way.
- The **TIME EXCEEDED** message is sent when a packet is dropped because its TtL (Time to live) counter has reached zero. This event is a symptom that packets are looping, or that the counter values are being set too low.
- One clever use of this error message is the **traceroute** utility that was developed by Van Jacobson in 1987.
- **Traceroute** finds the routers along the path from the host to a destination IP address.

213

Internet Control Message Protocol (ICMP)

- The **PARAMETER PROBLEM** message indicates that an illegal value has been detected in a header field.
- This problem indicates a bug in the sending host's IP software or possibly in the software of a router transited.
- The **SOURCE QUENCH** message was long ago used to throttle hosts that were sending too many packets. When a host received this message, it was expected to slow down.
- The **REDIRECT** message is used when a router notices that a packet seems to be routed incorrectly. It is used by the router to tell the sending host to update to a better route.

214

Internet Control Message Protocol (ICMP)

- The **ECHO** and **ECHO REPLY** messages are sent by hosts to see if a given destination is reachable and currently alive. Upon receiving the ECHO message, the destination is expected to send back an ECHO REPLY message.
- These messages are used in the ping utility that checks if a host is up and on the Internet.
- The **TIMESTAMP REQUEST** and **TIMESTAMP REPLY** messages are similar, except that the arrival time of the message and the departure time of the reply are recorded in the reply.
- This facility can be used to measure network performance.

215

Internet Control Message Protocol (ICMP)

- The **ROUTER ADVERTISEMENT** and **ROUTER SOLICITATION** messages are used to let hosts find nearby routers.
- A host needs to learn the IP address of at least one router to be able to send packets off the local network.
- In addition to these messages, others have been defined. The online list is now kept at www.iana.org/assignments/icmp-parameters.

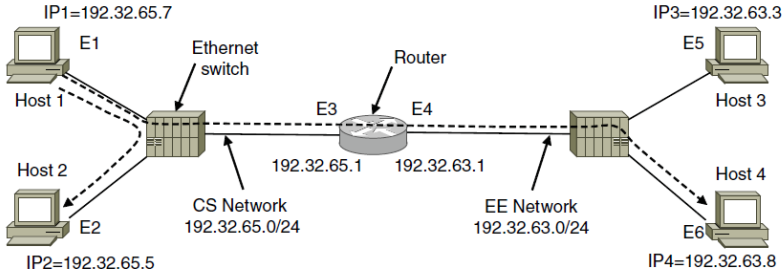
216

ARP—The Address Resolution Protocol

- Although every machine on the Internet has one or more IP addresses, these addresses are not sufficient for sending packets.
- Data link layer NICs (Network Interface Cards) such as Ethernet cards do not understand Internet addresses.
- The NICs send and receive frames based on 48-bit Ethernet addresses.
- They know nothing at all about 32-bit IP addresses.
- The question now arises, how do IP addresses get mapped onto data link layer addresses, such as Ethernet?

217

ARP—The Address Resolution Protocol



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

Two switched Ethernet LANs joined by a router

218

ARP—The Address Resolution Protocol

- In the example on previous slide, a small university with two /24 networks is illustrated.
- One network (CS) is a switched Ethernet in the Computer Science Dept.
- It has the prefix 192.32.65.0/24. The other LAN (EE), also switched Ethernet, is in Electrical Engineering and has the prefix 192.32.63.0/24.
- The two LANs are connected by an IP router.
- Each machine on an Ethernet and each interface on the router has a unique Ethernet address, labeled E1 through E6, and a unique IP address on the CS or EE network.

219

ARP—The Address Resolution Protocol

- How do IP addresses get mapped onto data link layer addresses, such as Ethernet?
- One solution is to have a configuration file somewhere in the system that maps IP addresses onto Ethernet addresses.
- While this solution is certainly possible, for organizations with thousands of machines keeping all these files up to date is an error-prone, time-consuming job.
- A better solution for Hosts to output a broadcast packet onto the Ethernet asking who owns the destination IP address.
- The broadcast will arrive at every machine on the Ethernet, and each one will check its IP address.

220

ARP—The Address Resolution Protocol

- The destination host alone will respond with its Ethernet address.
- In this way the source host learns that destination IP address is on the host with a particular Ethernet address.
- The protocol used for asking this question and getting the reply is called **ARP (Address Resolution Protocol).**
- Almost every machine on the Internet runs it. ARP is defined in RFC 826.
- The advantage of using ARP over configuration files is the simplicity.
- The system manager does not have to do much except assign each machine an IP address and decide about subnet masks. ARP does the rest.

221

ARP—The Address Resolution Protocol

- Various optimizations are possible to make ARP work more efficiently.
- To start with, once a machine has run ARP, it caches the result in case it needs to contact the same machine shortly.
- Next time it will find the mapping in its own cache, thus eliminating the need for a second broadcast.
- In many cases, the destination host will need to send back a reply, forcing it, too, to run ARP to determine the sender's Ethernet address.
- This ARP broadcast can be avoided by having sender host include its IP-to-Ethernet mapping in the ARP packet.
- When the ARP broadcast arrives at destination host, the pair (IP Address, Ethernet Address) is entered into destination host's ARP cache.
- In fact, all machines on the Ethernet can enter this mapping into their ARP caches.

222

ARP—The Address Resolution Protocol

- To allow mappings to change, for example, when a host is configured to use a new IP address (but keeps its old Ethernet address), entries in the ARP cache should time out after a few minutes.
- A clever way to help keep the cached information current and to optimize performance is to have every machine broadcast its mapping when it is configured.
- This broadcast is generally done in the form of an ARP looking for its own IP address.
- There should not be a response, but a side effect of the broadcast is to make or update an entry in everyone's ARP cache.
- This is known as a gratuitous ARP.
- If a response does (unexpectedly) arrive, two machines have been assigned the same IP address.
- The error must be resolved by the network manager before both machines can use the network.

223

OSPF—An Interior Gateway Routing Protocol

- The Internet is made up of a large number of independent networks or ASes (Autonomous Systems) that are operated by different organizations, usually a company, university, or ISP.
- Inside of its own network, an organization can use its own algorithm for internal routing, or intradomain routing, as it is more commonly known.
- An intradomain routing protocol is also called an interior gateway protocol.
- Then next, we will study the problem of routing between independently operated networks, or interdomain routing.
- For that case, all networks must use the same interdomain routing protocol or exterior gateway protocol.
- The protocol that is used in the Internet is BGP (Border Gateway Protocol).

226

OSPF—An Interior Gateway Routing Protocol

- Early intradomain routing protocols used a distance vector design, based on the distributed Bellman-Ford algorithm inherited from the ARPANET.
- RIP (Routing Information Protocol) is the main example that is used to this day. It works well in small systems, but less well as networks get larger.
- It also suffers from the count-to-infinity problem and generally slow convergence.
- The ARPANET switched over to a link state protocol in May 1979 because of these problems, and in 1988 IETF began work on a link state protocol for intradomain routing.
- That protocol, called OSPF (Open Shortest Path First), became a standard in 1990.

227

OSPF—An Interior Gateway Routing Protocol

- It drew on a protocol called IS-IS (Intermediate-System to Intermediate-System), which became an ISO standard.
- Because of their shared heritage, the two protocols are much more alike than different. For the complete story, see RFC 2328.
- They are the dominant intradomain routing protocols, and most router vendors now support both of them.
- OSPF is more widely used in company networks, and IS-IS is more widely used in ISP networks.

228

Design Goals of OSPF

- Given the long experience with other routing protocols, the group designing OSPF had a long list of requirements that had to be met.
- First, the algorithm had to be published in the open literature, hence the "O" in OSPF. A proprietary solution owned by one company would not do.
- Second, the new protocol had to support a variety of distance metrics, including physical distance, delay, and so on.
- Third, it had to be a dynamic algorithm, one that adapted to changes in the topology automatically and quickly.
- Fourth, and new for OSPF, it had to support routing based on type of service.
- The new protocol had to be able to route real-time traffic one way and other traffic a different way.

229

Design Goals of OSPF

- At the time, IP had a Type of service field, but no existing routing protocol used it. This field was included in OSPF but still nobody used it, and it was eventually removed.
- Perhaps this requirement was ahead of its time, as it preceded IETF's work on differentiated services, which has rejuvenated classes of service.
- Fifth, and related to the above, OSPF had to do load balancing, splitting the load over multiple lines.
- Most previous protocols sent all packets over a single best route, even if there were two routes that were equally good.
- The other route was not used at all. In many cases, splitting the load over multiple routes gives better performance.

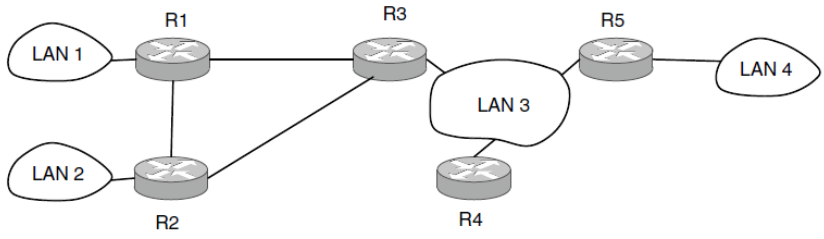
230

Design Goals of OSPF

- Sixth, support for hierarchical systems was needed.
- By 1988, some networks had grown so large that no router could be expected to know the entire topology.
- Seventh, some modicum of security was required to prevent fun-loving students from spoofing routers by sending them false routing information.
- Finally, provision was needed for dealing with routers that were connected to the Internet via a tunnel. Previous protocols did not handle this well.

231

OSPF—An Interior Gateway Routing Protocol



An autonomous system

232

OSPF—An Interior Gateway Routing Protocol

- An example of an autonomous system network is given in figure of previous slide.
- Hosts are omitted because they do not generally play a role in OSPF, while routers and networks (which may contain hosts) do.
- Most of the routers in figure are connected to other routers by point-to-point links, and to networks to reach the hosts on those networks.
- However, routers R3, R4, and R5 are connected by a broadcast LAN such as switched Ethernet.
- OSPF operates by abstracting the collection of actual networks, routers, and links into a directed graph in which each arc is assigned a weight (distance, delay, etc.).

233

OSPF—An Interior Gateway Routing Protocol

- A point-to-point connection between two routers is represented by a pair of arcs, one in each direction. Their weights may be different.
- A broadcast network is represented by a node for the network itself, plus a node for each router.
- The arcs from that network node to the routers have weight 0.
- They are important nonetheless, as without them there is no path through the network.
- Other networks, which have only hosts, have only an arc reaching them and not one returning.
- This structure gives routes to hosts, but not through them.

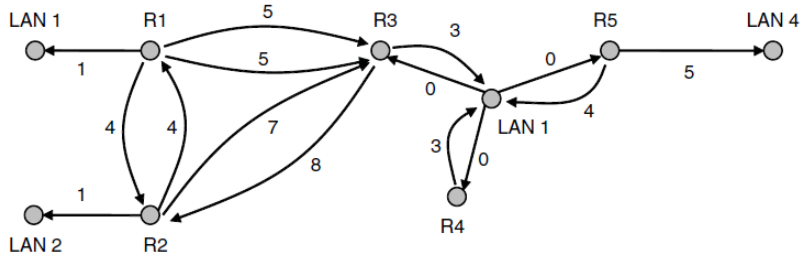
234

OSPF—An Interior Gateway Routing Protocol

- Figure on the next slide shows the graph representation of the network shown earlier in the figure.
- What OSPF fundamentally does is represent the actual network as a graph like this and then use the link state method to have every router compute the shortest path from itself to all other nodes.
- Multiple paths may be found that are equally short. In this case, OSPF remembers the set of shortest paths and during packet forwarding, traffic is split across them.
- This helps to balance load. It is called ECMP (Equal Cost MultiPath).

235

OSPF—An Interior Gateway Routing Protocol



A graph representation of the previous slide.

OSPF—An Interior Gateway Routing Protocol

- Many of the ASes in the Internet are themselves large and nontrivial to manage.
- To work at this scale, OSPF allows an AS to be divided into numbered **areas**, where an **area** is a network or a set of contiguous networks.
- Areas do not overlap but need not be exhaustive, that is, some routers may belong to no area.
- Routers that lie wholly within an area are called **internal routers**.
- An area is a generalization of an individual network. Outside an area, its destinations are visible but not its topology.
- This characteristic helps routing to scale.

OSPF—An Interior Gateway Routing Protocol

- Every AS has a **backbone area**, called area 0. The routers in this area are called **backbone routers**.
- All areas are connected to the backbone, possibly by tunnels, so it is possible to go from any area in the AS to any other area in the AS via the backbone.
- A tunnel is represented in the graph as just another arc with a cost.
- As with other areas, the topology of the backbone is not visible outside the backbone.
- Each router that is connected to two or more areas is called an area border router. It must also be part of the backbone.

238

OSPF—An Interior Gateway Routing Protocol

- The job of an **area border router** is to summarize the destinations in one area and to inject this summary into the other areas to which it is connected.
- This summary includes cost information but not all the details of the topology within an area. Passing cost information allows hosts in other areas to find the best area border router to use to enter an area.
- Not passing topology information reduces traffic and simplifies the shortest-path computations of routers in other areas.
- However, if there is only one border router out of an area, even the summary does not need to be passed.

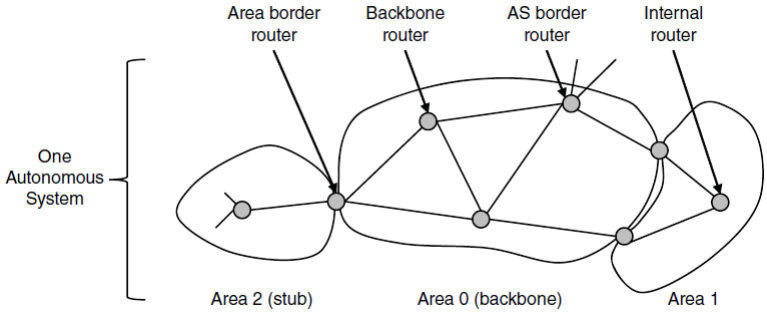
239

OSPF—An Interior Gateway Routing Protocol

- Routes to destinations out of the area always start with the instruction "Go to the border router." This kind of area is called a stub area.
- The last kind of router is the AS boundary router. It injects routes to external destinations on other ASes into the area.
- The external routes then appear as destinations that can be reached via the AS boundary router with some cost.
- An external route can be injected at one or more AS boundary routers.
- The relationship between ASes, areas, and the various kinds of routers is shown in figure of next slide.
- One router may play multiple roles, for example, a border router is also a backbone router.

240

OSPF—An Interior Gateway Routing Protocol



The relation between ASes, backbones, and areas in OSPF.

241

OSPF—An Interior Gateway Routing Protocol

- During normal operation, each router within an area has the same link state database and runs the same shortest path algorithm.
- Its main job is to calculate the shortest path from itself to every other router and network in the entire AS.
- An area border router needs the databases for all the areas to which it is connected and must run the shortest path algorithm for each area separately.
- For a source and destination in the same area, the best intra-area route (that lies wholly within the area) is chosen.
- For a source and destination in different areas, the inter-area route must go from the source to the backbone, across the backbone to the destination area, and then to the destination.

242

OSPF—An Interior Gateway Routing Protocol

- This algorithm forces a star configuration on OSPF, with the backbone being the hub and the other areas being spokes.
- Because the route with the lowest cost is chosen, routers in different parts of the network may use different area border routers to enter the backbone and destination area.
- Packets are routed from source to destination "as is." They are not encapsulated or tunneled (unless going to an area whose only connection to the backbone is a tunnel).
- Also, routes to external destinations may include the external cost from the AS boundary router over the external path, if desired, or just the cost internal to the AS.

243

OSPF—Message Types

Message type	Description
Hello	Used to discover who the neighbors are
Link state update	Provides the sender's costs to its neighbors
Link state ack	Acknowledges link state update
Database description	Announces which updates the sender has
Link state request	Requests information from the partner

The five types of OSPF messages

244

OSPF—An Interior Gateway Routing Protocol

- Finally, we can put all the pieces together. Using flooding, each router informs all the other routers in its area of its links to other routers and networks and the cost of these links.
- This information allows each router to construct the graph for its area(s) and compute the shortest paths. The backbone area does this work, too.
- In addition, the backbone routers accept information from the area border routers in order to compute the best route from each backbone router to every other router.
- This information is propagated back to the area border routers, which advertise it within their areas. Using this information, internal routers can select the best route to a destination outside their area, including the best exit router to the backbone.

245

BGP—The Exterior Gateway Routing Protocol

- Within a single AS, OSPF and IS-IS are the protocols that are commonly used.
- Between ASes, a different protocol, called BGP (Border Gateway Protocol), is used.
- A different protocol is needed because the goals of an intradomain protocol and an interdomain protocol are not the same.
- All an intradomain protocol has to do is move packets as efficiently as possible from the source to the destination. It does not have to worry about politics.
- In contrast, interdomain routing protocols have to worry about politics a great deal (Metz, 2001).

246

BGP—The Exterior Gateway Routing Protocol

- For example, a corporate AS might want the ability to send packets to any Internet site and receive packets from any Internet site.
- However, it might be unwilling to carry transit packets originating in a foreign AS and ending in a different foreign AS, even if its own AS is on the shortest path between the two foreign ASes (“That’s their problem, not ours”).
- On the other hand, it might be willing to carry transit traffic for its neighbors, or even for specific other ASes that paid it for this service.
- Telephone companies, for example, might be happy to act as carriers for their customers, but not for others.

247

BGP—The Exterior Gateway Routing Protocol

- Exterior gateway protocols in general, and BGP in particular, have been designed to allow many kinds of routing policies to be enforced in the interAS traffic.
- Typical policies involve political, security, or economic considerations.
- A few examples of routing constraints are:
  1. No commercial traffic for educational network
  2. Never put Iraq on route starting at Pentagon
  3. Choose cheaper network
  4. Choose better performing network
  5. Don't go from Apple to Google to Apple

248

BGP—The Exterior Gateway Routing Protocol

- A routing policy is implemented by deciding what traffic can flow over which of the links between ASes.
- One common policy is that a customer ISP pays another provider ISP to deliver packets to any other destination on the Internet and receive packets sent from any other destination.
- The customer ISP is said to buy transit service from the provider ISP.
- This is just like a customer at home buying Internet access service from an ISP.
- To make it work, the provider should advertise routes to all destinations on the Internet to the customer over the link that connects them.

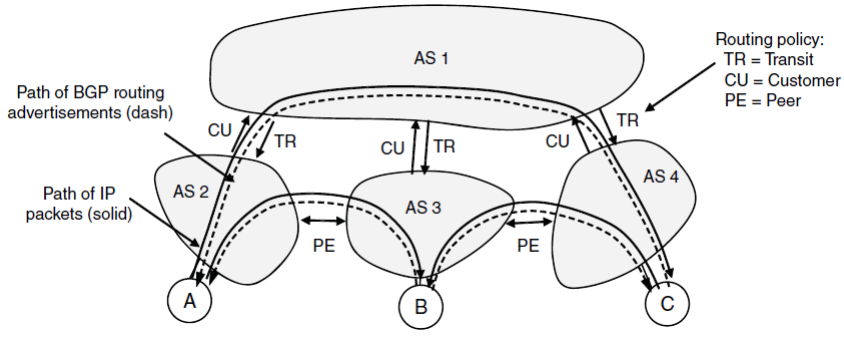
249

BGP—The Exterior Gateway Routing Protocol

- In this way, the customer will have a route to use to send packets anywhere.
- Conversely, the customer should advertise routes only to the destinations on its network to the provider.
- This will let the provider send traffic to the customer only for those addresses; the customer does not want to handle traffic intended for other destinations.
- We can see an example of transit service in Figure of next slide.
- There are four Ases that are connected. The connection is often made with a link at IXPs (Internet eXchange Points), facilities to which many ISPs have a link for the purpose of connecting with other ISPs. AS2, AS3, and AS4 are customers of AS1. They buy transit service from it.

250

BGP—The Exterior Gateway Routing Protocol



Routing policies between four Autonomous Systems

251

BGP—The Exterior Gateway Routing Protocol

- Thus, when source A sends to destination C, the packets travel from AS2 to AS1 and finally to AS4.
- The routing advertisements travel in the opposite direction to the packets. AS4 advertises C as a destination to its transit provider, AS1, to let sources reach C via AS1.
- Later, AS1 advertises a route to C to its other customers, including AS2, to let the customers know that they can send traffic to C via AS1.
- In Figure, all of the other ASes buy transit service from AS1.
- This provides them with connectivity so they can interact with any host on the Internet. However, they have to pay for this privilege.

252

BGP—The Exterior Gateway Routing Protocol

- Suppose that AS2 and AS3 exchange a lot of traffic. Given that their networks are connected already, if they want to, they can use a different policy—they can send traffic directly to each other for free.
- This will reduce the amount of traffic they must have AS1 deliver on their behalf, and hopefully it will reduce their bills.
- This policy is called **peering**. To implement peering, two ASes send routing advertisements to each other for the addresses that reside in their networks.
- Doing so makes it possible for AS2 to send AS3 packets from A destined to B and vice versa.
- However, note that peering is not transitive. In Figure, AS3 and AS4 also peer with each other.

253

BGP—The Exterior Gateway Routing Protocol

- This peering allows traffic from C destined for B to be sent directly to AS4.
- What happens if C sends a packet to A? AS3 is only advertising a route to B to AS4. It is not advertising a route to A.
- The consequence is that traffic will not pass from AS4 to AS3 to AS2, even though a physical path exists. This restriction is exactly what AS3 wants.
- It peers with AS4 to exchange traffic, but does not want to carry traffic from AS4 to other parts of the Internet since it is not being paid to so do.
- Instead, AS4 gets transit service from AS1. Thus, it is AS1 who will carry the packet from C to A.

254

BGP—The Exterior Gateway Routing Protocol

- Now that we know about transit and peering, we can also see that A, B, and C have transit arrangements. For example, A must buy Internet access from AS2.
- A might be a single home computer or a company network with many LANs.
- However, it does not need to run BGP because it is a stub network that is connected to the rest of the Internet by only one link.
- So the only place for it to send packets destined outside of the network is over the link to AS2. There is nowhere else to go.
- This path can be arranged simply by setting up a default route. For this reason, we have not shown A, B, and C as ASes that participate in interdomain routing.

255

BGP—The Exterior Gateway Routing Protocol

- On the other hand, some company networks are connected to multiple ISPs.
- This technique is used to improve reliability, since if the path through one ISP fails, the company can use the path via the other ISP.
- This technique is called multihoming. In this case, the company network is likely to run an interdomain routing protocol (e.g., BGP) to tell other ASes which addresses should be reached via which ISP links.
- BGP is a form of distance vector protocol, but it is quite unlike intradomain distance vector protocols such as RIP.
- We have already seen that policy, instead of minimum distance, is used to pick which routes to use.

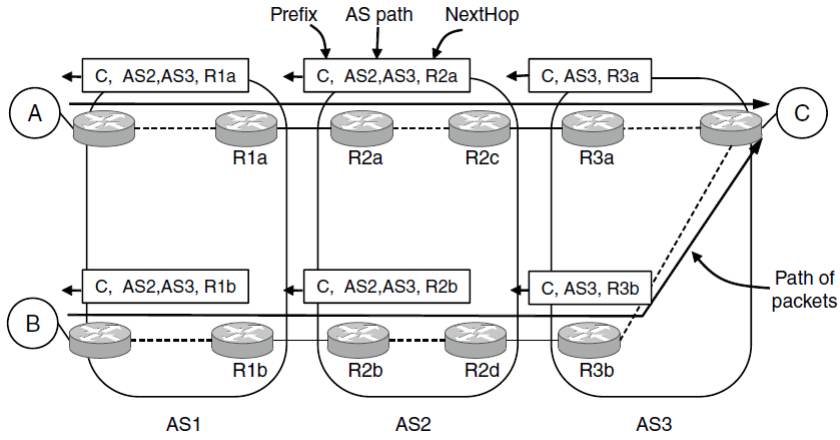
256

BGP—The Exterior Gateway Routing Protocol

- Another large difference is that instead of maintaining just the cost of the route to each destination, each BGP router keeps track of the path used.
- This approach is called a **path vector protocol**. The path consists of the next hop router (which may be on the other side of the ISP, not adjacent) and the sequence of ASes, or AS path, that the route has followed (given in reverse order).
- Finally, pairs of BGP routers communicate with each other by establishing TCP connections. Operating this way provides reliable communication and also hides all the details of the network being passed through.
- An example of how BGP routes are advertised is shown in Figure on next slide.

257

BGP—The Exterior Gateway Routing Protocol



Propagation of BGP route advertisements

258

BGP—The Exterior Gateway Routing Protocol

- There are three ASes and the middle one is providing transit to the left and right ISPs.
- A route advertisement to prefix C starts in AS3. When it is propagated across the link to R2c at the top of the figure, it has the AS path of simply AS3 and the next hop router of R3a.
- At the bottom, it has the same AS path but a different next hop because it came across a different link. This advertisement continues to propagate and crosses the boundary into AS1.
- At router R1a, at the top of the figure, the AS path is AS2, AS3 and the next hop is R2a.
- Carrying the complete path with the route makes it easy for the receiving router to detect and break routing loops.

259

BGP—The Exterior Gateway Routing Protocol

- The rule is that each router that sends a route outside of the AS prepends its own AS number to the route. (This is why the list is in reverse order.)
- When a router receives a route, it checks to see if its own AS number is already in the AS path. If it is, a loop has been detected and the advertisement is discarded.
- However, and somewhat ironically, it was realized in the late 1990s that despite this precaution BGP suffers from a version of the count-to-infinity problem (Labovitz et al., 2001).
- There are no long-lived loops, but routes can sometimes be slow to converge and have transient loops.

260

BGP—The Exterior Gateway Routing Protocol

- So far we have seen how a route advertisement is sent across the link between two ISPs.
- We still need some way to propagate BGP routes from one side of the ISP to the other, so they can be sent on to the next ISP.
- This task could be handled by the intradomain protocol, but because BGP is very good at scaling to large networks, a variant of BGP is often used. It is called iBGP (internal BGP) to distinguish it from the regular use of BGP as eBGP (external BGP).
- The rule for propagating routes inside an ISP is that every router at the boundary of the ISP learns of all the routes seen by all the other boundary routers, for consistency.

261

BGP—The Exterior Gateway Routing Protocol

- If one boundary router on the ISP learns of a prefix to IP 128.208.0.0/16, all the other routers will learn of this prefix.
- The prefix will then be reachable from all parts of the ISP, no matter how packets enter the ISP from other ASes.
- We can now describe the key missing piece, which is how BGP routers choose which route to use for each destination.
- Each BGP router may learn a route for a given destination from the router it is connected to in the next ISP and from all of the other boundary routers (which have heard different routes from the routers they are connected to in other ISPs).
- Each router must decide which route in this set of routes is the best one to use.

262

BGP—The Exterior Gateway Routing Protocol

- Ultimately the answer is that it is up to the ISP to write some policy to pick the preferred route.
- However, this explanation is very general and not at all satisfying, so we can at least describe some common strategies.
- The first strategy is that routes via peered networks are chosen in preference to routes via transit providers.
- The former are free; the latter cost money. A similar strategy is that customer routes are given the highest preference.
- It is only good business to send traffic directly to the paying customers.
- A different kind of strategy is the default rule that shorter AS paths are better.

263

BGP—The Exterior Gateway Routing Protocol

- The above discussion should make clear that each BGP router chooses its own best route from the known possibilities.
- It is not the case, as might naively be expected, that BGP chooses a path to follow at the AS level and OSPF chooses paths within each of the ASes.
- BGP and the interior gateway protocol are integrated much more deeply.
- This means that, for example, BGP can find the best exit point from one ISP to the next and this point will vary across the ISP.
- It also means that BGP routers in different parts of one AS may choose different AS paths to reach the same destination.

264

BGP—The Exterior Gateway Routing Protocol

- Care must be exercised by the ISP to configure all of the BGP routers to make compatible choices given all of this freedom, but this can be done in practice.
- Amazingly, we have only scratched the surface of BGP. For more information, see the BGP version 4 specification in RFC 4271 and related RFCs.
- However, realize that much of its complexity lies with policies, which are not described in the specification of the BGP protocol.

265

Mobile IP

Goals

1. Mobile host use home IP address anywhere.
2. No software changes to fixed hosts
3. No changes to router software, tables
4. Packets for mobile hosts – restrict detours
5. No overhead for mobile host at home.

References

Text Book:
1. Computer Networks, Andrew S. Tanenbaum, David J. Wetherall , 5th Edition, Pearson/Prentice Hall Publictaion.
Reference Book:
1. Data and Computer Communication, William Stallings, 8th Edition, Pearson/Prentice Hall Publictaion.
2. Data Communications and Networking, Behrouz A Forouzan, 3/e, McGrawHill Publication.
3. Computer Networks: A Systems Approach, Bruce S. Davie and Larry L. Peterson, 4e, Morgan Kaufmann Publication.
4. The TCP/IP Guide, by Charles M. Kozierok, Free online Resource http://www.tcpipguide.com/free/.

The End

268