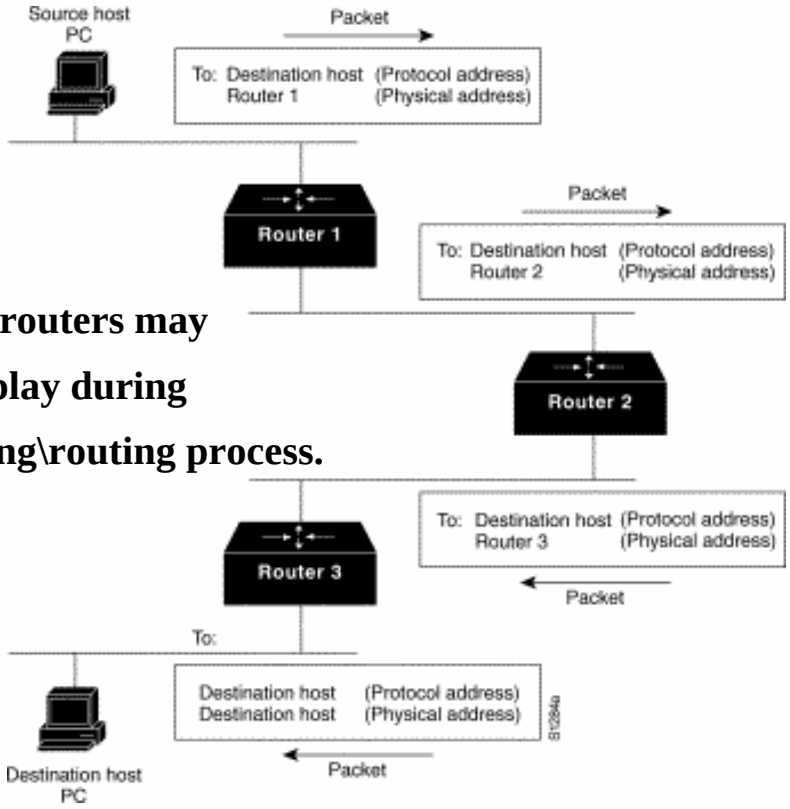
# Network Layer

Routing Algorithms

Routing Protocols

# Routing Algorithms

- Algorithms may be adaptive or non-adaptive (static)
- Static algorithms
  - Shortest path
  - Flooding
  - Flow based routing
- Dynamic algorithms
  - Distance vector routing
  - Link state routing




**Numerous routers may come into play during the switching\routing process.**



# Shortest Path - Dijkstra

- The Diagram on the next slide demonstrates Dijkstra's algorithm
- Work your way through the graph and ensure that you know how to find the shortest path
- For a demonstration visit ...
  - http://students.ceid.upatras.gr/~papagel/project/kef5_7_1.htm

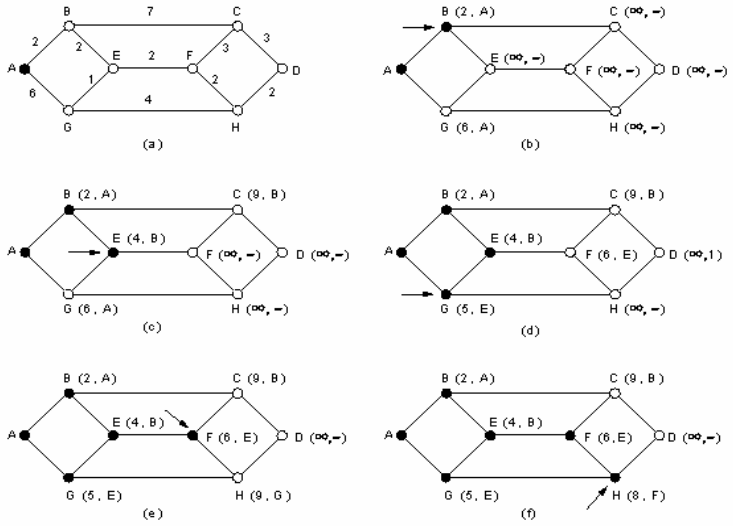



Fig. 5-6. The first five steps used in computing the shortest path from $A$ to $D$. The arrows indicate the working node.

# Flooding

- Every incoming packet is sent out on every outgoing line except the one it came in on.
- Huge numbers of duplicates
- Use hop-counter (time to live) to create a time to live field for packet. When hop-counter decrements to 0, do not retransmit.
- Not practical protocol for most applications although it chooses shortest path by choosing all paths. Very robust, good for military application.

# Flow based routing

- From queuing theory calculate load based on flow of traffic.
- “Longer” routes may have a greater throughput and make for a better pathway
- Must know
  - topology
  - traffic matrix
  - line capacity matrix



# Routing Protocols

RIP

OSPF

BGP

# Dynamic - Distance Vector Routing

- A dynamic routing algorithm!
- Each router maintains a table (vector) with best known distance to destinations and which line to use to get there.
- Also known as RIP
- *Distance* may be no. of hops, time delay, total number of packets queued on path, etc.



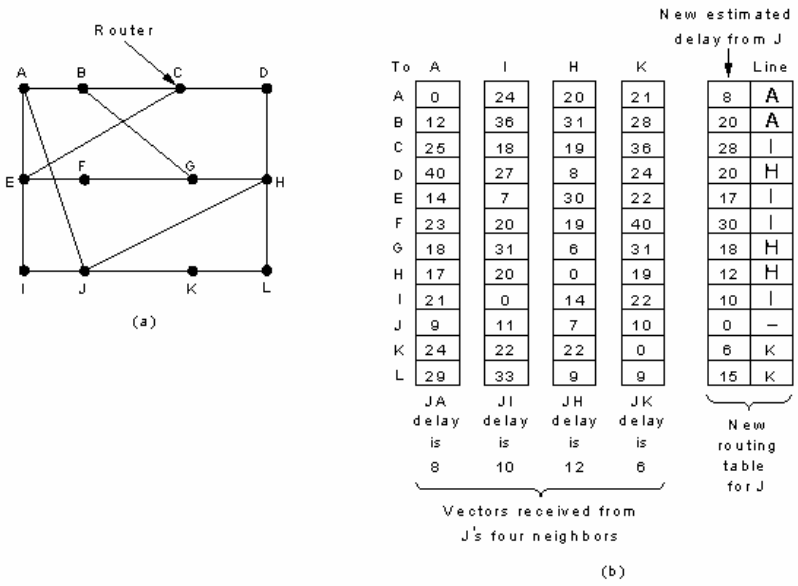
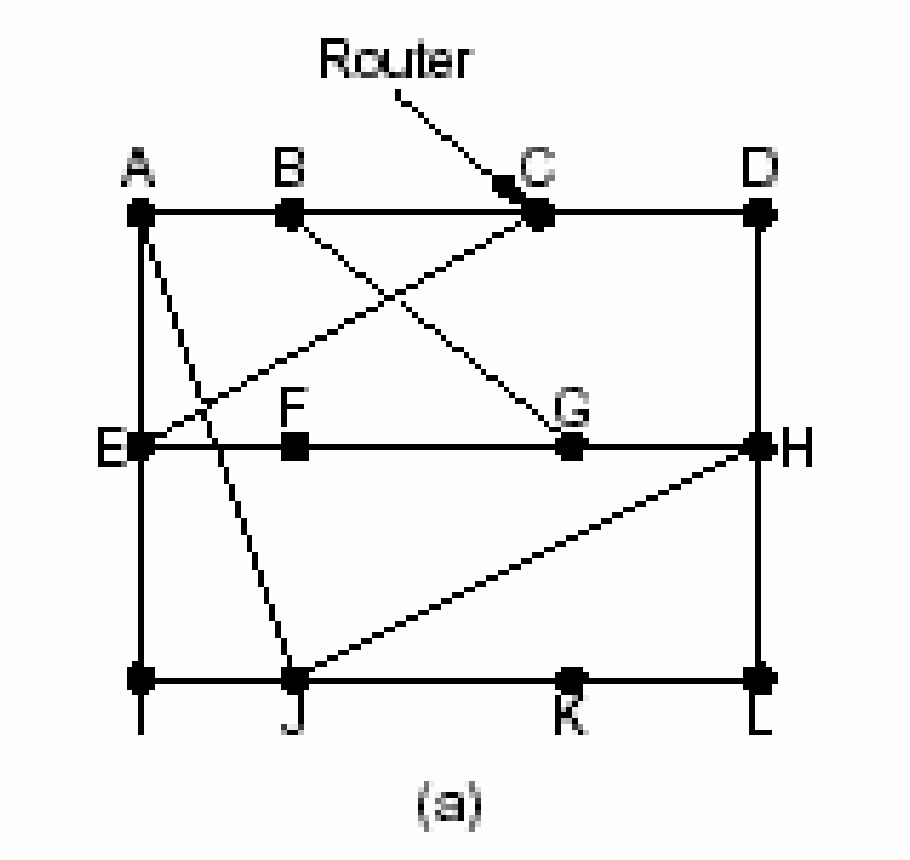



Fig. 5-10. (a) A subnet. (b) Input from *A*, *I*, *H*, *K*, and the new routing table for *J*.






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

Vectors received from J's four neighbors

# Updating Process

- First four columns show delay vectors received from neighbours of router J
- A has delay of 12msec to B, 25msec to C, 40msec to D...
- J has measured its delays to A, I, H and K as 8, 10, 12, 6msec
- J computes new route to router G… A takes 8msec, A gets to G in 18msec, so J can get to G in 26msec via A
- Compute delay to G via I, H and K as 41(31+10), 18(6+12) and 37(31+6)
- Best is 18 so entry to table is 18 and route is via H



# Count-to-infinity Problem

- Convergence is very slow.
- Good news propagates quickly, bad news slowly (if at all)
- In fact when a router goes down, all others start counting new route to crashed router ad infinitum.

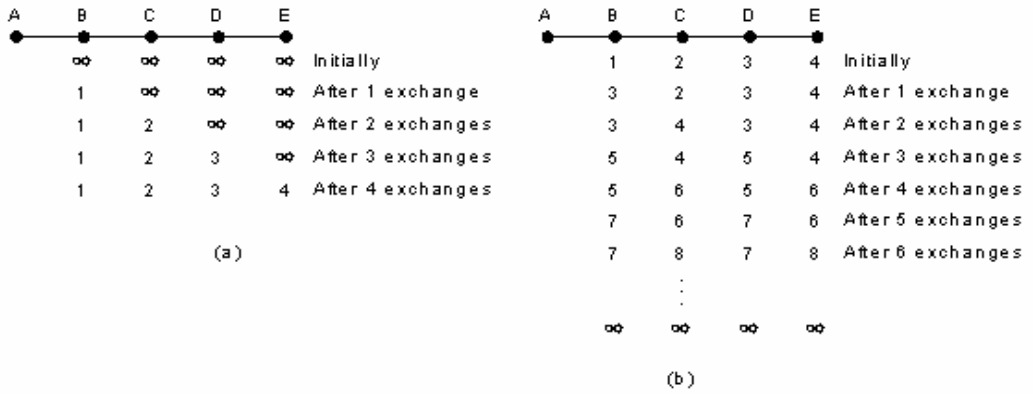



**Fig. 5-11. The count-to-infinity problem.**

# Split Horizon Hack

- There are no good solutions to count to infinity, only ad hoc solutions.
- Same as distance vector, except that distance to X is not reported on the line that packets for X are sent on, in fact it reports it as ∞
- Initial state: C tells D truthful distance to A, but tells B it is ∞
- Similarly D tells truth to E but lies to C (∞ )
- When A crashes, B sets distance to A as ∞ as both neighbours report distance as ∞
- On next exchange C hears that A is unreachable from both its neighbours so sets distance to ∞
- Bad news thus propagates quickly
- This hack does not always work, Ref Tanenbaum P. 358-9

# Dynamic - Link State Routing

- Link State Routing began to replace Distance Vector routing in 1979.
- Metric now takes into account bandwidth, not just queue length.
- Distance Vector took too long to converge, needed something better.



# Link State Features

- Discover neighbours and learn their network addresses.
- Measure delay or cost to each neighbour
- Construct a packet containing all of its information.
- Send this packet to all other routers.
- Compute the shortest path to each other router (Dijkstra)

# Neighbour Discovery

- On booting, a router sends out a "*hello*" packet on each line.
- Routers on other end are expected to respond saying who they are.
- In diagram, network itself is modelled as an artificial node.



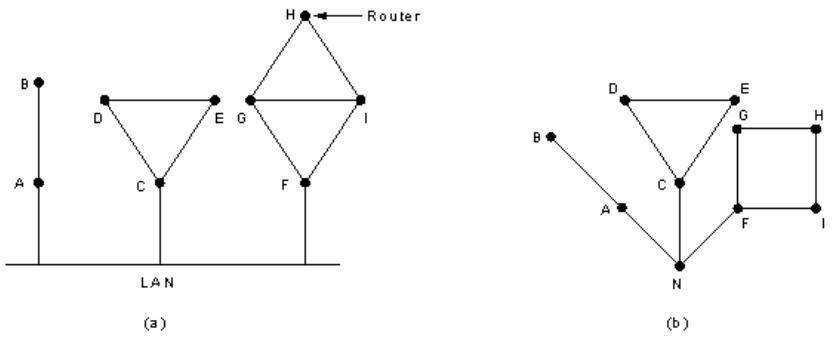



Fig. 5-13. (a) Nine routers and a LAN. (b) A graph model of (a).

# Line Cost

- Line costs may be computed on the basis of allowing for queue lengths or not.
- Use special *echo* packet to bounce off other router and time the transits.
- To take load into account start timer when echo packet is queued, to ignore it, start timer when echo packet reaches front of queue.



# Building Link State Packets

- Packet starts with ID of sender.
- SEQ no. and Age fields follow.
- When to build them?
  - Periodically
  - On an event
  - On appreciable change

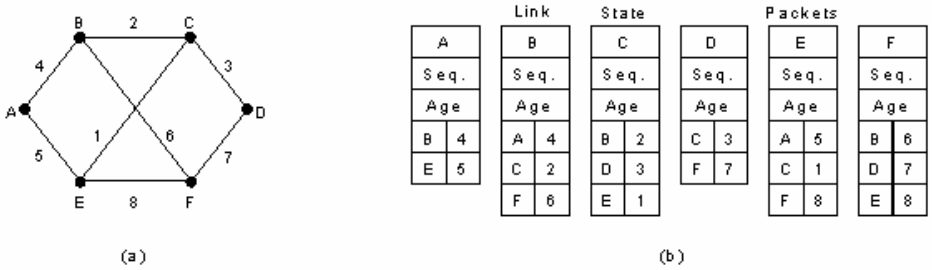



Fig. 5-15. (a) A subnet. (b) The link state packets for this subnet.

# Distributing Link State Packets

- Use flooding to distribute.
- SEQ no. constitutes a version field, only newer versions are considered. Records of current SEQ nos for routers are maintained.
- Age field constitutes a TTL field, decremented by each receiving router to zero then discarded.
- On receiving new packet, router checks validity, if valid flood it.

# Computing the New Routes

- Construct a graph describing subnet.
- Apply Dijkstra's algorithm to compute shortest path.
- Some problems exist
  - large subnets can have memory (RAM) problems
  - router failures give rise to incorrect information
- OSPF uses a link state algorithm (later)



# Link State Vs Distance Vector

- Link- state algorithms flood routing information to all nodes in the internetwork.
- Each router, however, sends only the portion of the routing table that describes the state of its own links
- Distance Vector algorithms call for each router to send all or some portion of its routing table, but only to its neighbours.
- Link- state algorithms send small updates everywhere, while distance vector algorithms send larger updates only to neighbouring routers.

- Because they converge more quickly, link- state algorithms are somewhat less prone to routing loops than distance-vector algorithms.
- On the other hand, link- state algorithms require more CPU power and memory than distance vector algorithms.
- Link-state algorithms, therefore, can be more expensive to implement and support.
- Distance vector suffers from count-to-infinity problem.



# Hierarchical Routing

- As network grows, tables grow also.
- Divide routers into regions, a router in a region does not know the topology of another region, only its entry point.
- Path lengths may increase with hierarchy.
- How many levels to use?

# Routing in the Internet

Fragmentation & Reassembly

IGP (RIP & OSPF)

BGP (Distance Vector)

Mobile IP

# Fragmentation & Reassembly

- A network imposes a maximum packet size
- Issues
  - Hardware
  - OS
  - Underlying Protocols
  - Standards compliance
  - Error control (reduce retransmissions)
  - Channel sharing

- Transparent fragmentation
  - Entry gateway fragments and sends all fragments to single exit gateway (on that network) where reassembly occurs.
  - Exit gateway must know when all fragments are available
  - Some routing performance hit may be encountered.
  - Large overhead in processing.



- Nontransparent fragmentation
  - Do reassembly at destination host
  - Every host must be capable of reassembly
  - Header processing lasts for full lifetime
- Must number fragments, what about one corrupt fragment?
- How do we retransmit a fragment?

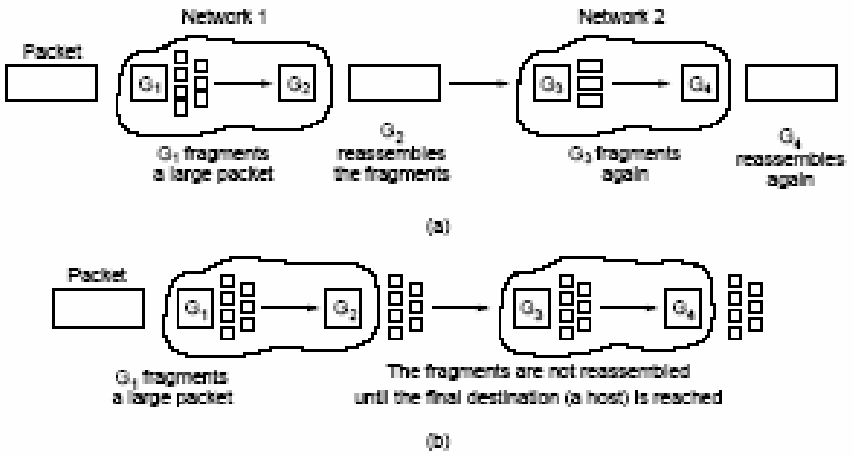



Fig. 5-41. (a) Transparent fragmentation. (b) Nontransparent fragmentation.



# IP Addresses - Recap

- IP V4 addresses specify a TCP/IP address for (usually) a specific machine.
- A machine can have many IP addresses, and IP address can apply to one machine only.
- 4 Byte or decimal dot notation.
- Class A, B and C common. We have class B, 136.206.X.X in DCU (prove it!)

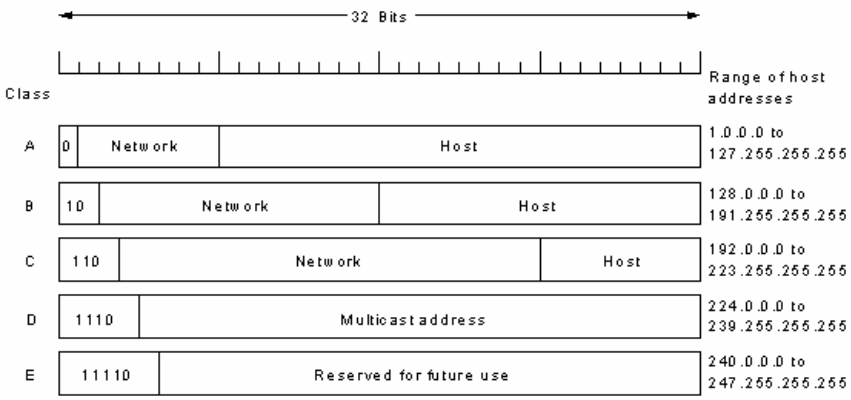



Fig. 5-47. IP address formats.

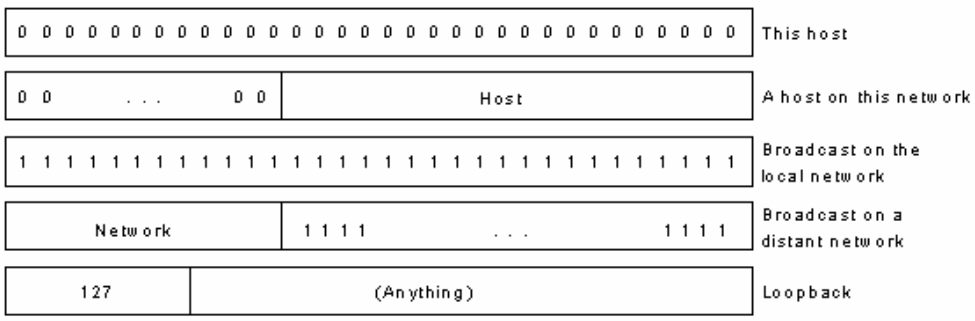
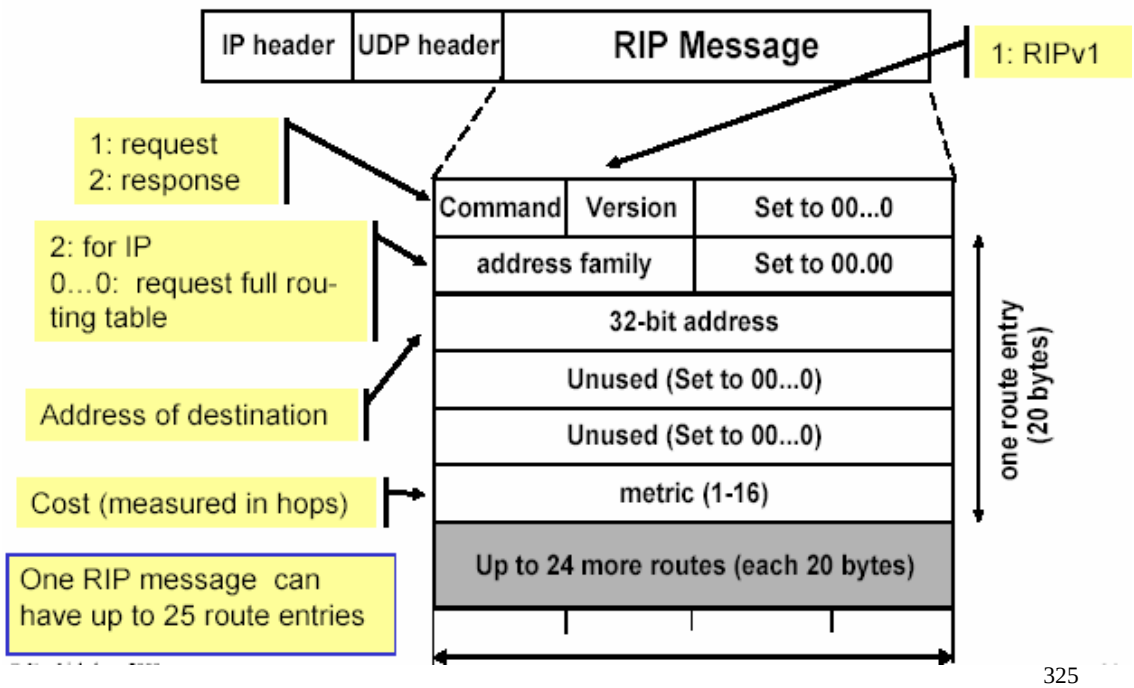



Fig. 5-48. Special IP addresses.

# RIPv1 Packet Format

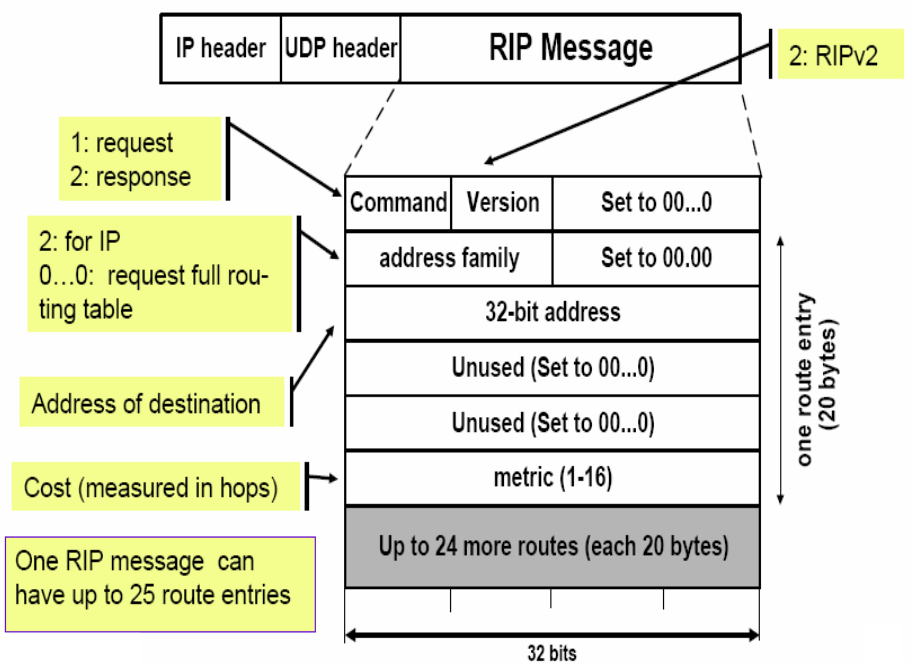





# RIP V2

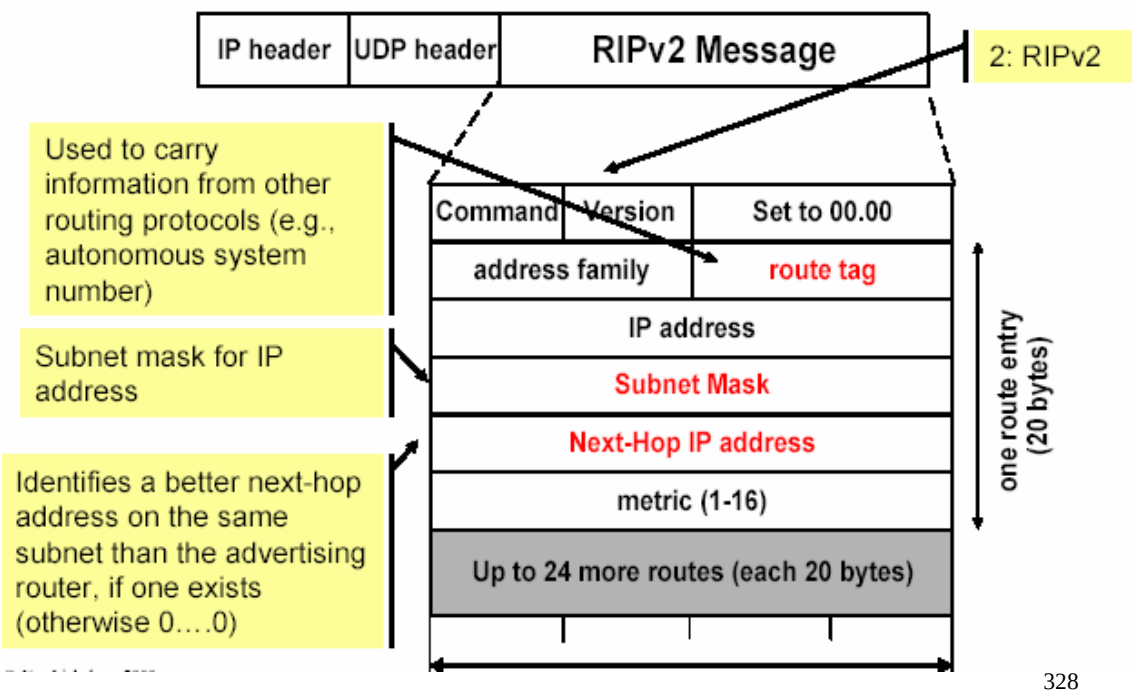
- RIPv2 is an extension of RIPv1:
  - Subnet masks are carried in the route information
  - Authentication of routing messages
  - Route information carries next-hop address
  - Exploites IP multicasting
  - Extensions of RIPV2 are carried in unused fields of RIPv1 messages

# RIP V2 Packet Format






# RIP Packet Format V2



# RIP messages

- RIP in Unix OS is *routed* daemon.
- Dedicated port for RIP is UDP port 520.
- Two types of messages:
  - **Request messages** used to ask neighbouring nodes for an update
  - **Response messages** contains an update



# RIP Routing

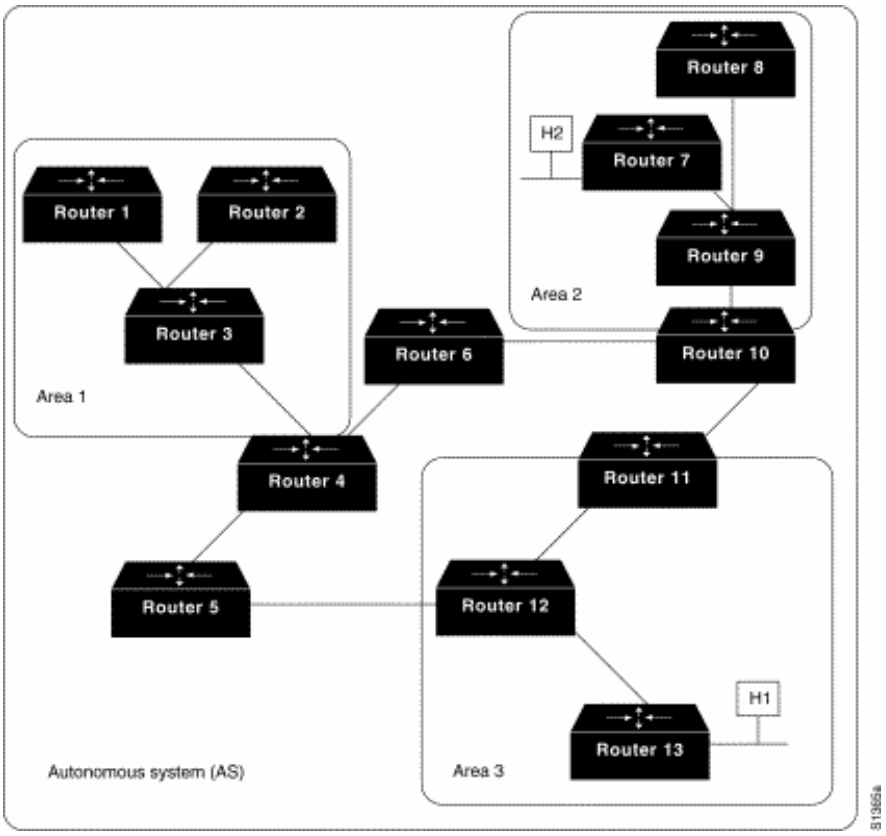
- **Initialization:** Send a **request packet** (command = 1, address family=0..0) on all interfaces:
- RIPv1 uses broadcast if possible,
- RIPv2 uses multicast address 224.0.0.9, if possible requesting routing tables from neighbouring routers
- **Request received**: Routers that receive above request send their entire routing table
- **Response received**: Update the routing table
- **Regular routing updates**: Every 30 seconds, send all or part of the routing tables to every neighbour in a response message
- **Triggered Updates:** Whenever the metric for a route change, send entire routing table.

# IGP & OSPF

- Interior Gateway Routing Protocol
- Original protocol was RIP (distance vector)
- RFC 1247 describes the 1988 OSPF
- OSPF is **Open**. Spec. is in public domain.
- Based on Dijkstra's algorithm.
- *Link-state-advertisements*, LSA, are sent to all routers within same area, with info on interfaces, metrics used, and other variables.



**An OSPF AS consists of multiple areas linked by routers**

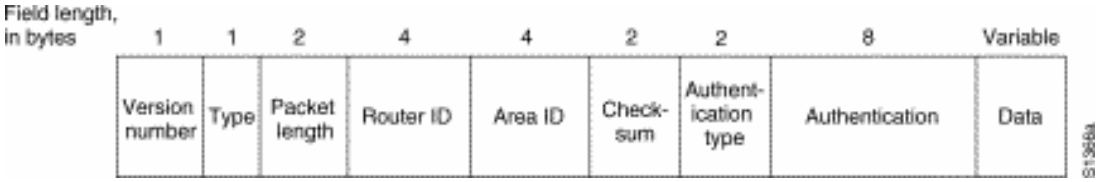


# OSPF

- Can operate within hierarchy (RIP cannot)
- Routers may operate in multiple areas.
  - *Area Border Routers*
- Term *Domain* means Autonomous System.
- *Backbone* is responsible for sharing info between areas. This is area 0.
- *Designated Router* is said to be adjacent to all other routers, thus is central repository. Reduces complexity.



# OSPF Packet Format




- *Type*---Identifies the OSPF packet type as one of the following:
  - Hello: Establishes and maintains neighbour relationships.
  - Database Description: Describes the contents of the topological database. These messages are exchanged when an adjacency is initialised.
  - Link-state Request: Requests pieces of the topological database from neighbour routers. These messages are exchanged after a router discovers (by examining database-description packets) that parts of its topological database are out of date.
  - Link-state Update: Responds to a link-state request packet. These messages also are used for the regular dispersal of LSAs. Several LSAs can be included within a single link-state update packet.
  - Link-state Acknowledgement: Acknowledges link-state update packets.

- *Packet Length*---Specifies the packet length, including the OSPF header, in bytes.
- *Router ID*---Identifies the source of the packet.
- *Area ID*---Identifies the area to which the packet belongs. All OSPF packets are associated with a single area.
- *Checksum*---Checks the entire packet contents for any damage suffered in transit.
- *Authentication Type*---Contains the authentication type. All OSPF protocol exchanges are authenticated. The Authentication Type is configurable on a per-area basis.
- *Authentication*---Contains authentication information.
- Data---Contains encapsulated upper-layer information.



# Other OSPF Features

- *Multipath routing*
  - Multiple paths to the same destination. Unlike single-path algorithms, multipath algorithms permit traffic multiplexing over multiple lines. They can provide substantially better throughput and reliability.
- Upper-layer *type-of-service* (TOS) requests
  - Might specify that certain data is urgent. If OSPF has high-priority links at its disposal, these can be used to transport the urgent datagram

# Routing in the Internet

BGP

# BGP (A Distance Vector Protocol)

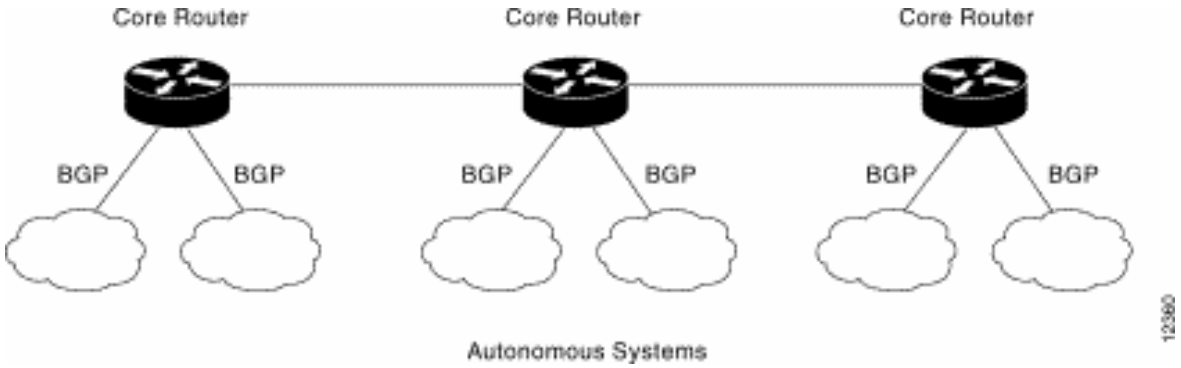
- BGP performs interdomain routing in TCP/IP networks.
- BGP is an exterior gateway protocol (EGP), which means that it performs routing between multiple autonomous systems or domains and exchanges routing and reachability information with other BGP systems.
- BGP was developed to replace its predecessor, the now obsolete *Exterior Gateway Protocol* (EGP).
- BGP solves serious problems with EGP and scales to Internet growth more efficiently.
- RFC 1771 Describes BGP4, the current version of BGP
- BGP links routers with open TCP connections for routing information exchanges.

# BGP and Politics

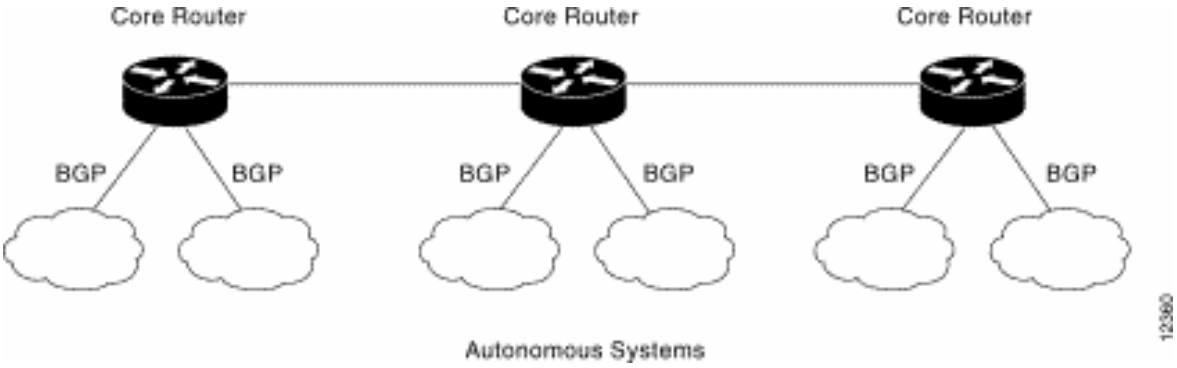
- Routing “policy” may be implemented.
- Policy is configured at the router, not part of the standard.
- BGP notes not just “distance” to foreign network, but also path used. More than just a distance vector protocol.
- Any route given to a router which breaks policy is given a distance of $\infty$ prior to applying Dijkstra’s algorithm.
- BGP does not suffer from count-to-infinity as it can discount routes which involve itself and consider only routes which are independent.



Core routers can use BGP to route traffic between autonomous systems.



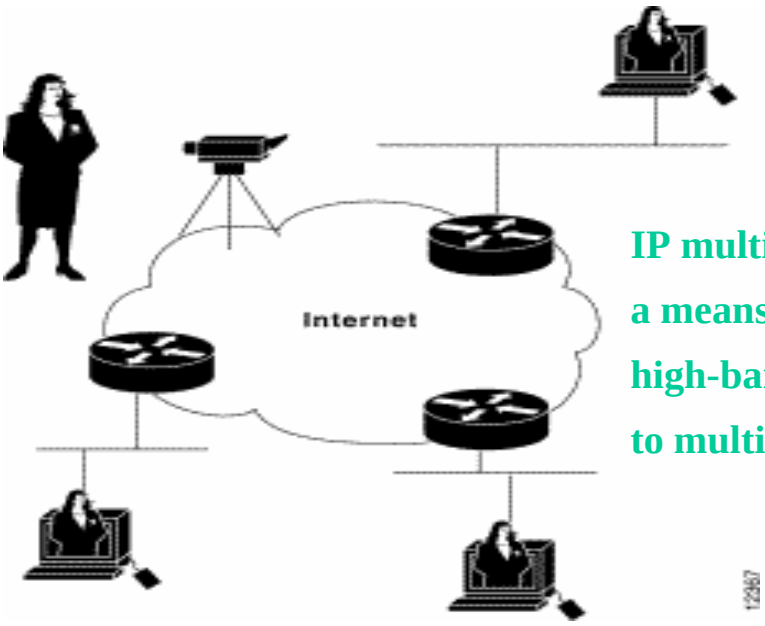
# BGP Diagram




**Core routers can use BGP to route traffic between autonomous systems.**



# IP Multicast




**IP multicast provides a means to deliver high-bandwidth traffic to multiple destinations.**

- IP multicast routing arose because unicast and broadcast techniques do not handle the requirements of new applications efficiently
- Multicast addressing supports the transmission of a single IP datagram to multiple hosts.
- A single packet is sent to a multicast group, which is identified by a single IP destination group address.
- A principle component of IP multicast is the Internet Group-Membership Protocol (IGMP).



# IGMP

- Internet Group-Membership Protocol (IGMP) relies on Class D IP addresses for the creation of multicast groups and is defined in RFC 1112.
- IGMP is used to dynamically register individual hosts in a multicast group with a Class D address.
- Hosts identify group memberships by sending IGMP messages, and traffic is sent to all members of that multicast group.
- Under IGMP, routers listen to IGMP messages and periodically send out queries to discover which groups are active or inactive on particular LANs.
- Routers communicate with each other by using one or more protocols to build multicast routes for each group.