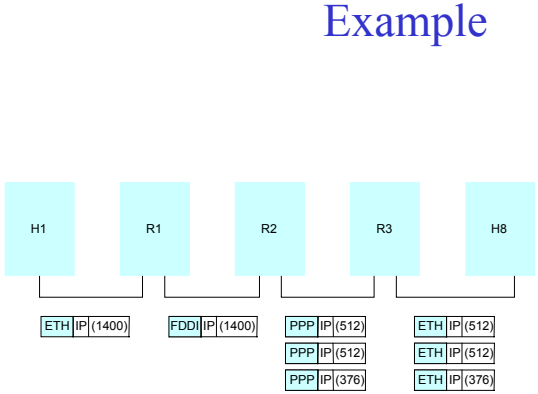
Project

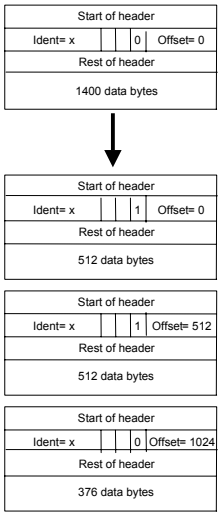
- Don't connect socket before calling em_connect()
- The emulator drops packets that it causes errors in: you don't have to check the packets you receive for errors
- Handling the high error rate in experiment 3

Fragmentation and Reassembly

- Each network has some MTU
- Strategy
  - fragment when necessary (MTU < Datagram)
  - try to avoid fragmentation at source host
  - re-fragmentation is possible
  - fragments are self-contained datagrams
  - use CS-PDU (not cells) for ATM
  - delay reassembly until destination host
  - do not recover from lost fragments

Example





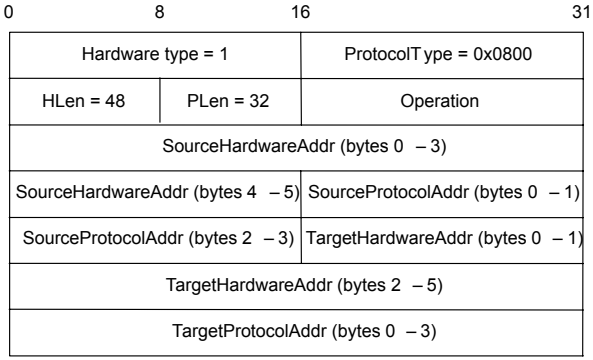
Interaction of errors and fragmentation

- 8KByte response to remote file system reads
  - Why choose a response size bigger than the network MTU?
- Response must be fragmented for ethernet – 6 fragments per datagram
- If there is a high error rate – lots of retransmissions
  - Bad ethernet interface story

Address Translation

- Map IP addresses into physical addresses
  - destination host
  - next hop router
- Techniques
  - encode physical address in host part of IP address
  - table-based
- ARP
  - table of IP to physical address bindings
  - broadcast request if IP address not in table
  - target machine responds with its physical address
  - table entries are discarded if not refreshed

ARP Packet Format



ARP Details

- Request Format
  - HardwareType: type of physical network (e.g., Ethernet)
  - ProtocolType: type of higher layer protocol (e.g., IP)
  - HLEN & PLEN: length of physical and protocol addresses
  - Operation: request or response
  - Source/Target-Physical/Protocol addresses
- Notes
  - table entries timeout in about 10 minutes
  - update table with source when you are the target
  - update table if already have an entry
  - do not refresh table entries upon reference

DHCP

- Configure the networking on a host when it boots
  - IP address, netmask, hostname, nameservers, etc.
    - Preconfigured or Pooled
    - www.isc.org/products/DHCP
  - Relies on broadcast -- special broadcast address 255.255.255.255
    - Can't use "normal" broadcast address because netmask isn't known yet
      - (ipaddr & netmask) | ~netmask
  - "Inverse" of the problem solved by ARP

Internet Control Message Protocol (ICMP)

- Echo (ping)
- Redirect (from router to source host)
- Destination unreachable (protocol, port, or host)
- TTL exceeded (so datagrams don't cycle forever)
- Checksum failed
- Reassembly failed
- Cannot fragment

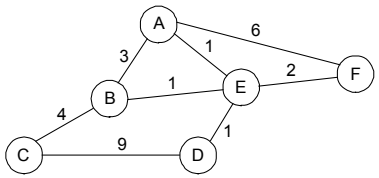
Routing

Outline

- Algorithms
- Scalability

Overview

- Forwarding vs Routing
  - forwarding: to select an output port based on destination address and routing table
  - routing: process by which routing table is built
- Network as a Graph

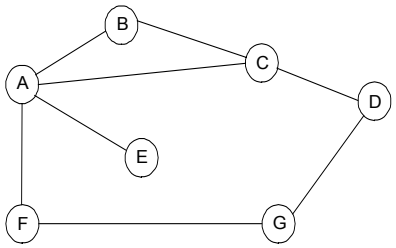


- Problem: Find lowest cost path between two nodes
- Factors
  - static: topology
  - dynamic: load

Distance Vector

- Each node maintains a set of triples
  - `(Destination, Cost, NextHop)`
- Exchange updates directly connected neighbors
  - periodically (on the order of several seconds)
  - whenever table changes (called *triggered* update)
- Each update is a list of pairs:
  - `(Destination, Cost)`
- Update local table if receive a "better" route
  - smaller cost
  - came from next-hop
  - Refresh existing routes; delete if they time out

Example



Destination	Cost	NextHop
A	1	A
C	1	C
D	2	C
E	2	A
F	2	A
G	3	A

Routing Loops

- Example 1
  - F detects that link to G has failed
  - F sets distance to G to infinity and sends update t o A
  - A sets distance to G to infinity since it uses F to reach G
  - A receives periodic update from C with 2-hop path to G
  - A sets distance to G to 3 and sends update to F
  - F decides it can reach G in 4 hops via A
- Example 2
  - link from A to E fails
  - A advertises distance of infinity to E
  - B and C advertise a distance of 2 to E
  - B decides it can reach E in 3 hops; advertises this to A
  - A decides it can read E in 4 hops; advertises this to C
  - C decides that it can reach E in 5 hops…