ICS 451: Today's plan

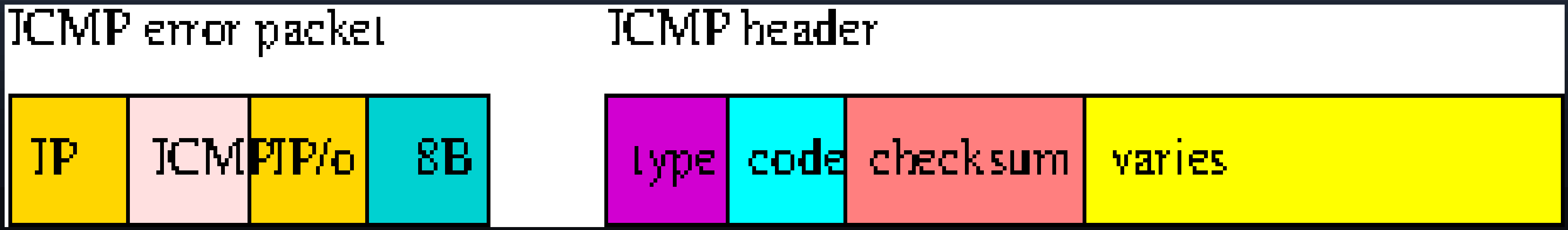
- ICMP
  - ping
  - traceroute
- ARP
- DHCP
- summary of IP processing

ICMP

- Internet Control Message Protocol, 2 functions:
  - error reporting (never sent in response to ICMP error packets)
  - network debugging (ping)
- common errors packets include
  - destination unreachable (network, host, protocol, or port unreachable)
  - fragmentation needed
  - TTL expired (Time Exceeded)
  - redirect
  - parameter problem

ICMP error message format

- IP header, ICMP header, IP header of error packet, 64 bits (8 bytes) of IP payload of error packet
- ICMP header (8 bytes) has type (e.g. destination unreachable), code (e.g. host unreachable) and 4 or more bytes that vary depending on the type



Ping

- ICMP type ECHO, code 0
- ICMP type ECHO REPLY, code 0
- 4 extra bytes hold two bytes of ID (usually process ID) and two bytes of sequence number
- any additional bytes carry sender's time, in binary (usually 8 bytes)
  - when the packet returns, can compare time received (from system) with time sent (in packet)

Traceroute

- When a router drops a packet, it sends back a Time Exceeded in Transit ICMP error
- so if I send a packet with TTL 1, my router should send me an error
- TTL 2, the next router should send me an error
- ...
- the final host should send me port unreachable
  - sometimes blocked by firewalls

Traceroute example, IPv4

```
edo@uhx01 1 % traceroute -n www.ietf.org
traceroute to www.ietf.org (104.20.1.85), 30 hops max, 40 byte packets
 1  128.171.24.193  0.345 ms  0.297 ms  0.186 ms
 2  128.171.1.201  1.478 ms  1.127 ms  1.172 ms
 3  128.171.64.190  1.085 ms  0.945 ms  1.008 ms
 4  205.166.205.48  1.021 ms  0.863 ms  1.083 ms
 5  74.202.119.9  1.541 ms  1.598 ms  14.653 ms
 6  64.129.238.190  70.884 ms  52.823 ms  73.906 ms
 7  * 4.68.71.137  54.391 ms  54.440 ms
 8  4.69.144.138 54.595 ms 4.69.144.202 53.114 ms 4.69.144.74 52.828ms
 9  4.68.70.130  53.027 ms  58.414 ms  55.843 ms
10  62.115.32.214  84.881 ms  81.319 ms  81.322 ms
11  104.20.1.85  62.808 ms  62.954 ms  55.745 ms
```

Traceroute example, IPv6

```
$ traceroute6 -n www.ietf.org

traceroute to www.ietf.org
(2400:cb00:2048:1::6814:55), 30 hops max, 80
byte packets

 1  2001:470:a:446::1  71.683 ms  73.466 ms
75.001 ms

 2  2001:470:0:9b::1  85.305 ms  85.277 ms
85.256 ms

 3  2001:504:16::3417  75.380 ms  75.373 ms
75.324 ms

 4  2400:cb00:28:1024::6ca2:f46a  75.116 ms
2400:cb00:28:1024::6ca2:f415  75.430 ms
2400:cb00:28:1024::6ca2:f46a  75.243 ms
```

Traceroute from a remote place

```
$ traceroute -n www.hawaii.edu

traceroute to www.hawaii.edu (128.171.224.100), 30 hops max,
60 byte packets

 1  192.168.3.1  2.092 ms  1.994 ms  1.974 ms

 2  192.168.2.254  2.851 ms  3.169 ms *

 3  192.168.2.1  4.487 ms  8.058 ms  8.059 ms

 4  133.205.178.201  10.316 ms  10.311 ms  10.297 ms

 5  192.168.4.1  715.377 ms  715.392 ms  716.995 ms

 6  182.33.57.117  747.101 ms  775.364 ms  805.649 ms

 7  182.33.57.213  805.668 ms  814.364 ms  864.240 ms

 8  * * 69.43.227.5  879.163 ms

 9  64.128.3.1  908.117 ms  908.138 ms  926.854 ms

10  66.192.243.238  819.116 ms  819.554 ms  820.491 ms

11  74.202.119.10  792.610 ms  764.176 ms  738.531 ms

12  128.171.213.2  739.577 ms  739.445 ms  709.709 ms
```

Path MTU discovery

- sender sends as big packets as its network MTU can support
  - if TCP MSS option is present in the SYN packets, sender will use the smaller of its own and its peer's MSS
  - TCP also sets the Don't Fragment (DF) bit
    - routers cannot fragment IPv6 packets
- router that cannot forward, sends

  destination unreachable, fragmentation needed
  - payload indicates MTU of interface
- next router may do the same again!

Deprecated ICMP messages

- source quench: send to a sender that is filling up a router's buffers
  - increases traffic during congestion
  - apparently not very effective
- redirect: on this same network, for this destination, use this other router
  - can be used for man-in-the-middle attacks
  - might still work on many systems (there are other ways to do MITM)

IP over Data-Link layer

- when sending or forwarding an IP packet, the routing table gives us interface and next hop
- the next hop is an IP address
- the data link layer requires a MAC address
- we need to *resolve* the IP address to a MAC address
- then the packet can be sent to the MAC address of the next hop

Address Resolution Protocol

- ARP cache saves IP->MAC mappings
  - for a few minutes unless refreshed
- mapping obtained by broadcasting a request
  - Who has 1.2.3.4? tell 5.6.7.8
- response is unicast to the MAC in the request
- responder (and everyone else) also caches the mapping in the request

ARP Variants

- Reverse ARP (RARP): given a MAC address, give me an IP address
  - similar to DHCP, but less commonly used
- Proxy ARP: another device replies “for” the device that has the IP address
  - invisible to the requester
  - useful on non-broadcast networks
    - as long as requet sent to the Proxy ARP server
  - other uses as well

ARP Security

- ARP Spoofing: an ARP reply with incorrect information
  - can make an IP unreachable
  - can make an attacker the Man-In-The-Middle
- only defense is to monitor the LAN for strange ARP replies
  - e.g. several replies for the same request

DHCP

- Dynamic Host Configuration Protocol
- each interface needs an IP address
- configuring each by hand is tedious and error-prone
- instead, configure them all in a DHCP server, let the server tell the machines when they boot
- can also have a “pool” of addresses
  - assigned on demand to anyone on the network
- less adminstration, less chance of people selecting an address at random

Summary: IP processing for incoming packets

- compute IP header checksum
  - discard if incorrect
- reassemble if necessary
- if destination address is one of ours
  - check protocol field
    - deliver to TCP or UDP
    - or do ICMP processing
      - including delivery to ping/traceroute
- otherwise decrement TTL, forward packet
  - if TTL==0 or not forwarding, discard packet

DHCP Information

- Interface IP address
- Netmask
- Default Gateway(s)
- Default Name Server(s)

and quite a bit more!

DHCP Transmission

- DHCP is carried over UDP
- but sender may not have an IP address!
  - nor know IP address of DHCP server
- DHCP packets sent from 0.0.0.0 to 255.255.255.255
  - and over the LAN broadcast address, usually ff:ff:ff:ff:ff:ff

Summary: IP processing for incoming packets

- compute IP header checksum
  - discard if incorrect
- reassemble if necessary
- if destination address is one of ours
  - check protocol field
    - deliver to TCP or UDP
    - or do ICMP processing
      - including delivery to ping/traceroute
- otherwise decrement TTL, forward packet
  - if TTL==0 or not forwarding, discard packet

Summary: IP processing for outgoing packets

- find longest-match route **next-hop-IP,interface**
- compare packet size to MTU of **interface**
  - fragment if needed (or discard if DF==1)
- compute IP header checksum
- check ARP table for **next-hop-IP**
  - if not found, ARP on **interface** and wait for response
- send packet on **interface**, to MAC address corresponding to **next-hop-IP**