

ELEC / COMP 177 – Fall 2011

Computer Networking

➔ Internet Control Message Protocol (ICMP)

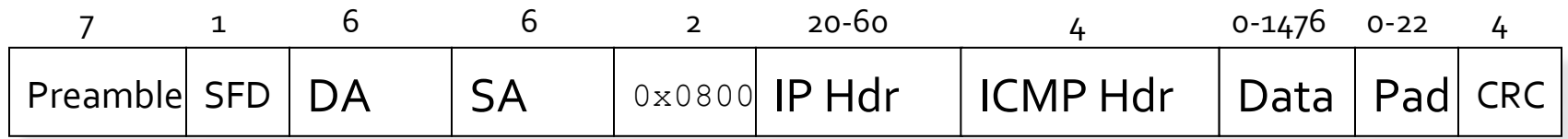
Some slides from Kurose and Ross, *Computer Networking*, 5th Edition

Internet Control Message Protocol

- One of the core protocols in the Internet
- Primarily used to communicate errors among routers and hosts
 - IP datagram errors
 - Communicate routing information/errors
 - Communicate diagnostics
- Not (typically) used by applications
 - Applications communicate application-level errors using higher level protocols
 - Ping and traceroute are the exceptions

ICMP Packets

Bytes:



ICMP Packet

- ICMP packets are encapsulated in IP datagrams
 - IP protocol field: ICMP (0x01)
- Header fields
 - Type (1 byte)
 - Code (1 byte)
 - Checksum (2 bytes)

ICMP in IP in Ethernet

Destination MAC Address				
Destination MAC Address		Source MAC Address		
Source MAC Address				
Type (0x0800)		Version	HdrLen	Type of Service
Total Length		Identification		
Flags	Fragment Offset	Time-To-Live		Protocol (0x01)
Header Checksum		Source IP Address		
Source IP Address		Destination IP Address		
Destination IP Address		Options and Padding		
Options and Padding		Type		Code
Checksum		Payload		
Ethernet CRC				

ICMP Types

- Echo request (0x8) /reply (0x0)
- Destination unreachable (0x3)
- Router solicitation (0xA) /advertisement (0x9)
- Time exceeded (0xB)
- Timestamp request (0xD) / reply (0xE)
- Traceroute (0x1E)
- ...

Ping

- Common tool used to test basic connectivity
 - Is target host alive?
 - Is there a route to the target host?
 - How long does it take to reach the target host?
- Allows collection of basic diagnostic information
 - I.e., is your network set up correctly?

Ping

```
dhcp-10-10-207-20:~ shafer$ ping -c 3 www.pacific.edu
PING www.pacific.edu (192.168.200.100): 56 data bytes
64 bytes from 192.168.200.100: icmp_seq=0 ttl=252 time=0.738 ms
64 bytes from 192.168.200.100: icmp_seq=1 ttl=252 time=1.025 ms
64 bytes from 192.168.200.100: icmp_seq=2 ttl=252 time=0.776 ms
```

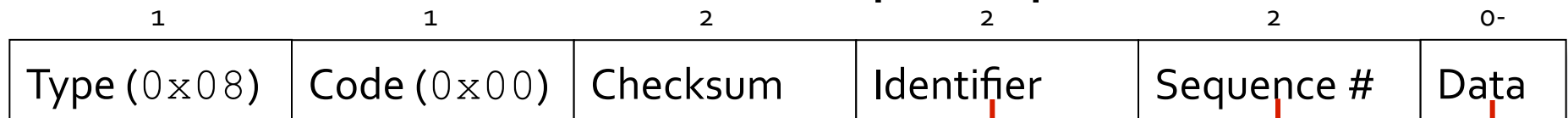
```
--- www.pacific.edu ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 0.738/0.846/1.025/0.127 ms
```

```
dhcp-10-10-207-20:~ shafer$ ping -c 3 www.google.com
PING www.l.google.com (74.125.19.103): 56 data bytes
64 bytes from 74.125.19.103: icmp_seq=0 ttl=56 time=7.534 ms
64 bytes from 74.125.19.103: icmp_seq=1 ttl=56 time=7.295 ms
64 bytes from 74.125.19.103: icmp_seq=2 ttl=56 time=7.661 ms
```

```
--- www.l.google.com ping statistics ---
3 packets transmitted, 3 packets received, 0.0% packet loss
round-trip min/avg/max/stddev = 7.295/7.497/7.661/0.152 ms
```

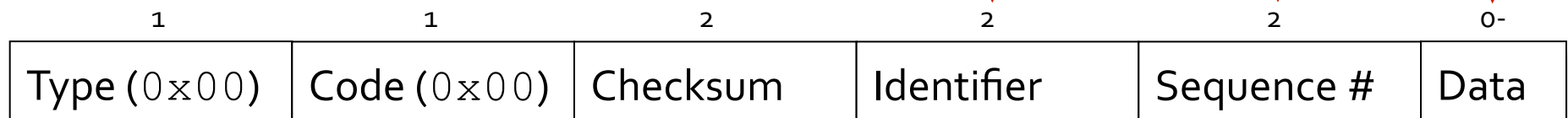
ICMP Echo

- Ping uses ICMP
 - ICMP Echo Request (type 8)
 - ICMP Echo Reply (type 0)
- Sender creates Echo Request packets



Copy

- Receiver replies with Echo Reply packets



Ping

```
shafer@comp519:~$ ping www.amazon.com
PING www.amazon.com (72.21.206.5) 56(84) bytes of data.
From xe-2-4.r04.asbnva01.us.ce.gin.ntt.net (168.143.105.30) icmp_seq=3 Packet
    filtered
From xe-2-4.r04.asbnva01.us.ce.gin.ntt.net (168.143.105.30) icmp_seq=9 Packet
    filtered
From xe-2-4.r04.asbnva01.us.ce.gin.ntt.net (168.143.105.30) icmp_seq=10 Packet
    filtered

--- www.amazon.com ping statistics ---
11 packets transmitted, 0 received, +3 errors, 100% packet loss, time 10002ms
shafer@comp519:~$
```

Filtered Ping Response

The image shows a Wireshark capture window titled "(Untitled) - Wireshark (on comp519)". The main display area shows a list of network packets. Packet 244 is highlighted in orange and contains an ICMP message: "Destination unreachable (Communication administratively filtered)". This packet is the focus of the red circle. Below the packet list, the packet details pane shows the structure of the ICMP message, including the type (3) and code (13). The packet bytes pane at the bottom shows the raw data in hexadecimal and ASCII.

No.	Time	Source	Destination	Protocol	Info
243	33.185608	168.7.19.135	168.7.23.142	TCP	2695 > ssh [ACK] Seq=4480 Ack=3648 Win=64095 Len=0
244	33.194047	168.143.105.30	168.7.23.142	ICMP	Destination unreachable (Communication administratively filtered)
245	33.194228	168.7.23.142	10.128.92.32	DNS	Standard query PTR 30.105.143.168.in-addr.arpa

Frame 244 (70 bytes on wire, 70 bytes captured)
Ethernet II, Src: Cisco_21:9a:40 (00:15:c7:21:9a:40), Dst: TyanComp_5a:71:aa (00:e0:81:5a:71:aa)
Internet Protocol, Src: 168.143.105.30 (168.143.105.30), Dst: 168.7.23.142 (168.7.23.142)
Version: 4
Header length: 20 bytes
Differentiated Services Field: 0x00 (DSCP 0x00: Default; ECN: 0x00)
Total Length: 56
Identification: 0x04de (1246)
Flags: 0x00
Fragment offset: 0
Time to live: 248
Protocol: ICMP (0x01)
Header checksum: 0xec3 [correct]
Source: 168.143.105.30 (168.143.105.30)
Destination: 168.7.23.142 (168.7.23.142)
Internet Control Message Protocol
Type: 3 (Destination unreachable)
Code: 13 (Communication administratively filtered)
Checksum: 0xe3d7 [correct]
Internet Protocol, Src: 168.7.23.142 (168.7.23.142), Dst: 72.21.210.11 (72.21.210.11)
Internet Control Message Protocol

```
0000 00 e0 81 5a 71 aa 00 15 c7 21 9a 40 08 00 45 00  ...Zq... .!@.E.
0010 00 38 04 de 00 00 f8 01 ec a3 a8 8f 69 1e a8 07  .8..... .i...
0020 17 8e 03 0d e3 d7 00 00 00 00 45 00 00 54 00 00  .|. .... .E..T..
0030 40 00 36 01 6a f3 a8 07 17 8e 48 15 d2 0b 08 00  @.6.j... ..H....
0040 74 a0 9c 62 00 18                                t..b..
```

Type (icmp.type), 1 byte | P: 266 D: 266 M: 0 Drops: 0

ICMP Destination Unreachable

1	1	2	2	2	28-68
Type (0x03)	Code	Checksum	Unused	Next Hop MTU	Orig. IP Hdr + 8B

- Example codes
 - 0: Network unreachable
 - 1: Host unreachable
 - 4: Datagram too large (DF flag set)
 - "Next hop MTU" set in this case
 - 9: Destination network administratively prohibited
 - 13: Communication administratively prohibited
- Original IP header and start of payload helps sender to match ICMP error to original datagram

Traceroute Revisited

- Tool to find the route IP packets take through the Internet
- Exploits the TTL field
 - Send packets with successively increasing TTL values
 - See what routers respond with ICMP “Time Exceeded” errors

ICMP Time Exceeded

1	1	2	4	28-68
Type (0x0B)	Code	Checksum	Unused	Orig. IP Hdr + 8B

- Code 0: TTL equals 0 during transit
- Original IP header and start of payload helps sender to match ICMP error to original datagram
 - Notice “unused” field is larger, so header/payload starts at same offset

Using Traceroute

```
jshafer@ecs-network:~$ traceroute www.google.com
traceroute to www.google.com (74.125.224.48), 30 hops max, 60 byte packets
 1  10.10.5.252 (10.10.5.252)  0.540 ms  0.717 ms  0.700 ms
 2  10.0.0.93 (10.0.0.93)  0.663 ms  0.645 ms  0.718 ms
 3  10.0.0.105 (10.0.0.105)  0.362 ms  0.570 ms  0.626 ms
 4  138.9.253.252 (138.9.253.252)  0.930 ms  0.915 ms  0.895 ms
 5  74.202.6.5 (74.202.6.5)  4.209 ms  4.192 ms  4.443 ms
 6  paol-pr1-ge-2-0-0-0.us.twtelecom.net (66.192.242.66)  5.727 ms  5.393 ms
    5.373 ms
 7  216.239.49.250 (216.239.49.250)  6.629 ms  6.608 ms  6.588 ms
 8  64.233.174.15 (64.233.174.15)  6.530 ms  6.514 ms  6.514 ms
 9  nuq04s06-in-f16.1e100.net (74.125.224.48)  6.039 ms  6.025 ms  6.461 ms
```

Traceroute

eth0 - Wireshark

File Edit View Go Capture Analyze Statistics Telephony Tools Help

Filter: + Expression... Clear Apply

No.	Time	Source	Destination	Protocol	Info
85	1.029246	10.10.4.101	10.6.163.131	SSH	Encrypted response packet len=48
86	1.030436	10.6.163.131	10.10.4.101	TCP	63110 → ssh [ACK] Seq=1277774414 Win=65535 Len=0 TSV=257440615 TCEP=870440608
87	1.039146	10.10.4.101	10.10.4.226	DNS	Standard query AAAA www.google.com
88	1.040190	10.10.4.101	10.10.4.226	DNS	Standard query response AAAA www.google.com
89	1.040903	10.10.4.101	10.10.4.226	DNS	Standard query response AAAA www.google.com
90	1.042590	10.10.4.101	10.10.4.226	DNS	Standard query response AAAA www.google.com A 74.125.224.48 A 74.125.224.48
91	1.043089	10.10.4.101	74.125.224.48	UDP	Source port: 55543 Destination port: traceroute
92	1.043199	10.10.4.101	74.125.224.48	UDP	Source port: 57292 Destination port: 33438
93	1.043223	10.10.4.101	74.125.224.48	UDP	Source port: 45669 Destination port: 33439
94	1.043246	10.10.4.101	74.125.224.48	UDP	Source port: 45949 Destination port: 33440
95	1.043277	10.10.4.101	74.125.224.48	UDP	Source port: 56903 Destination port: 33441
96	1.043319	10.10.4.101	74.125.224.48	UDP	Source port: 54125 Destination port: 33442
97	1.043346	10.10.4.101	74.125.224.48	UDP	Source port: 59435 Destination port: 33443
98	1.043369	10.10.4.101	74.125.224.48	UDP	Source port: 43748 Destination port: 33444
99	1.043391	10.10.4.101	74.125.224.48	UDP	Source port: 50196 Destination port: 33445
100	1.043414	10.10.4.101	74.125.224.48	UDP	Source port: 33446 Destination port: 33446
101	1.043436	10.10.4.101	74.125.224.48	UDP	Source port: 33447 Destination port: 33447
102	1.043462	10.10.4.101	74.125.224.48	UDP	Source port: 33448 Destination port: 33448
103	1.043606	10.0.0.105	10.10.4.101	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
104	1.043623	10.10.5.252	10.10.4.101	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
105	1.043775	10.10.4.101	10.10.4.226	DNS	Standard query PTR 252.5.10.10.in-addr.arpa
106	1.043832	10.10.5.252	10.10.4.101	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
107	1.043836	10.0.0.93	10.10.4.101	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
108	1.043840	10.10.5.252	10.10.4.101	ICMP	Time-to-live exceeded (Time to live exceeded in transit)
109	1.043843	10.0.0.93	10.10.4.101	ICMP	Time-to-live exceeded (Time to live exceeded in transit)

Send "probe" packets with varying TTLs

DNS Request and Response: lookup www.google.com, receive 74.125.224.48

Receive ICMP Responses

Frame 107 (70 bytes on wire, 70 bytes captured)

Ethernet II, Src: Cisco_Et1_7d:00 (00:22:90:2f:7d:00), Dst: Vmware_bf:7c:88 (00:50:56:bb:7c:88)

Internet Protocol, Src: 10.0.0.105 (10.0.0.105), Dst: 10.10.4.101 (10.10.4.101)

Internet Control Message Protocol

Type: 11 (Time-to-live exceeded)

Code: 0 (Time to live exceeded in transit)

Checksum: 0x235b [correct]

Internet Protocol, Src: 10.10.4.101 (10.10.4.101), Dst: 74.125.224.48 (74.125.224.48)

Frame (frame), 70 bytes

Packets: 801 Displayed: 801 Marked: 0 Dropped: 0

Profile: Default

Traceroute ICMP

- Traceroute requires sender to send a bunch of probe packets
 - Varying TTLs
 - May get sent along different routes
 - May send extra packets before sender finds destination
- Better method?
 - Send one packet with “traceroute” IP Option set
 - Routers along path respond with traceroute ICMP packet
 - Receiving host responses also include IP Option

Traceroute IP Option

- Used on packets between “client” and “server”
- Format:
 - Type (1 byte) = 82
 - Length (1 byte) = 12
 - ID Number (2 bytes)
 - An arbitrary number set by the transmitter to identify this particular traceroute request
 - Outbound Hop Count (2 bytes)
 - Number of routers this outbound packet passed through (incremented at each hop for an outgoing packet from client)
 - Return Hop Count (2 bytes)
 - Number of routers this return packet passed through (incremented at each hop for an incoming packet from server)
 - Originator IP Address (4 bytes)

Traceroute ICMP Packet

- Used for all return packets (“the reply”) from router to client
- Since this is an ICMP packet, it includes IP headers (e.g. router IP address)
- Format
 - Type (1 byte) - Traceroute: 0x1E
 - Code (1 byte)
 - 0: outbound packet forwarded
 - 1: no route for outbound packet, discarded
 - ICMP header checksum (2 bytes)
 - Identifier (2 bytes)
 - ID number copied from IP traceroute option of packet
 - NOT identifier from IP header
 - Unused (2 bytes)
 - Outbound hop count (2 bytes)
 - Copied from IP traceroute option of original (request) packet
 - Return hop count (2 bytes)
 - Output link speed (4 bytes, measured in bytes per second of the link over which this packet will be sent)
 - Output link MTU (4 bytes, measured in bytes)

ICMP Traceroute Process

- Send IP packet with IP traceroute option
- Actions taken by router
 - Send ICMP Traceroute packet to sender
 - Increment hop count in IP option field
 - Forward IP packet normally (if possible)
- Now, sender need only send one packet
 - Packet will be routed normally
 - Each router will notify sender
- Problems with this design?
 - Not commonly used/implemented

Resources

- <http://www.networksorcery.com>
 - RFC sourcebook