**Chapter 09: Internet Control Message Protocol Version 4**

Table of Contents

[9.2 Messages 3](#_Toc86833894)

[Redirection 5](#_Toc86833895)

[Message Format 6](#_Toc86833896)

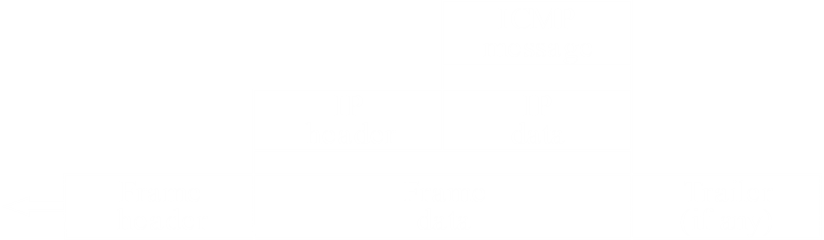
[9.3 Debugging Tools 8](#_Toc86833897)

[Ping 8](#_Toc86833898)

[Traceroute 9](#_Toc86833899)

There are situations where things go wrong with the IP protocol. Perhaps a router must discard a packet in the middle of the path because it cannot find the destination or because the time-to-live has expired. The IP protocol has no built-in mechanisms to notify the original host about the issue.

To deal with this situation, we have the **Internet Control Message Protocol** (ICMP). When a router drops a packet, it creates an **ICMP message**. This could be either an **error reporting** message or a **query** message. This message is put in the **data section** of a new **IP packet**.



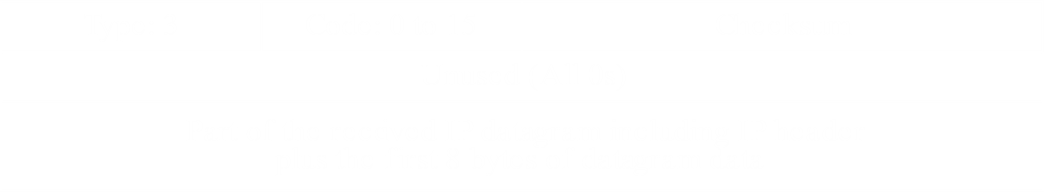
This also requires an **IP header**, which must include a destination address. In this case, the destination is the original source, and its address can be found in the original IP packet. The source now becomes the router that dropped the packet.

## 9.2 Messages

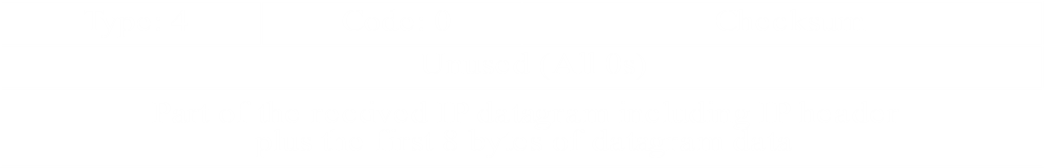
ICMP supports two types of messages, **error reporting messages** and **query messages**, each of which can be divided into subcategories.

* Error Reporting Messages:

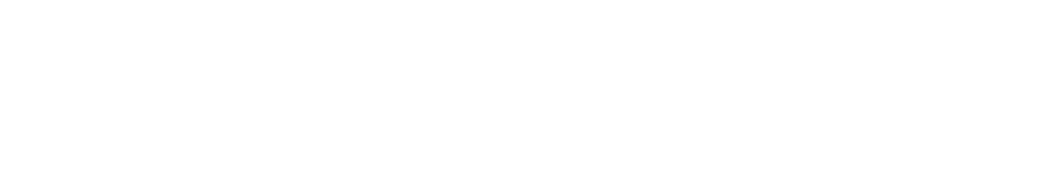
1. **Destination Unreachable**

****

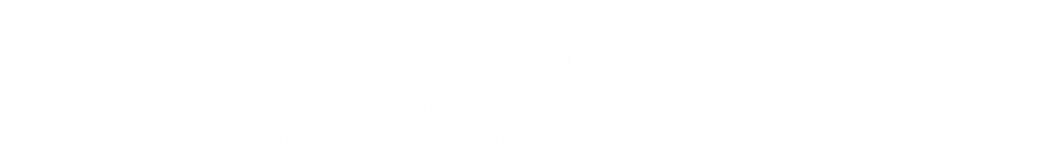
1. **Source Quench** – This means the data is coming at a higher rate than the destination can process. Thus, the source not only knows that the packet was dropped, but also that it needs to decrease the outgoing rate. We are achieving some **flow control** by using ICMP.

****

1. **Time Exceeded** – The time to live has expired or the reassembly time for fragments has expired.

****

1. **Parameter Problem** – One of the parameters is wrong, for example if the version field says IPv10, or one of the options that is required cannot be performed.

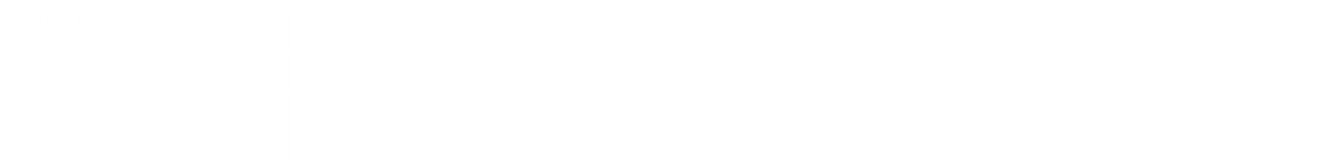
****

1. **Redirection** – This is explained below.

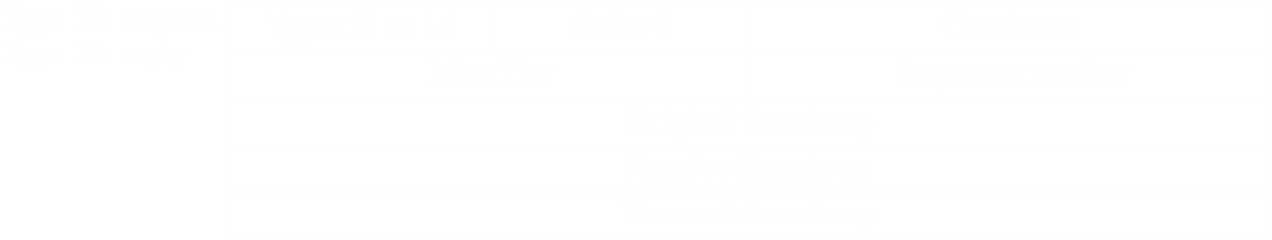
* Query Messages:

Query messages work as **request-reply pairs**.

1. **Echo Request or Reply** – This is a ping. It is used to check whether a device is reachable.

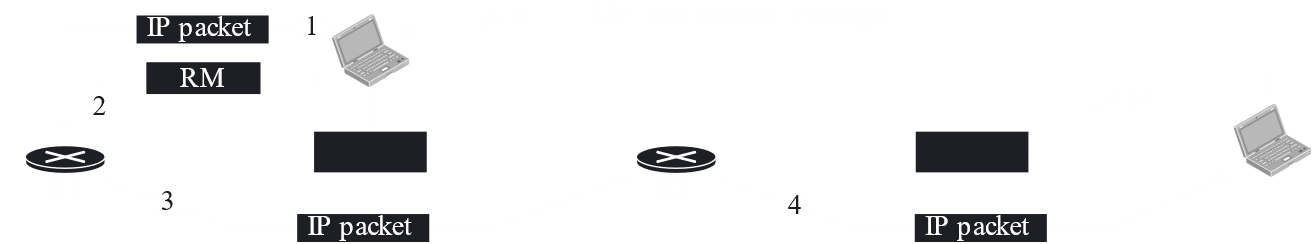
****

1. **Timestamp Request or Reply** – This is used to calculate the round-trip time, even if the devices are not synchronized. If the exact one-way time duration is known, this can also be used to synchronize the devices.

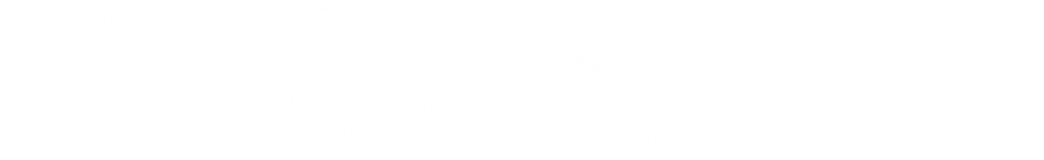


Notice that the receive timestamp as well as the transmit timestamp are sent back.

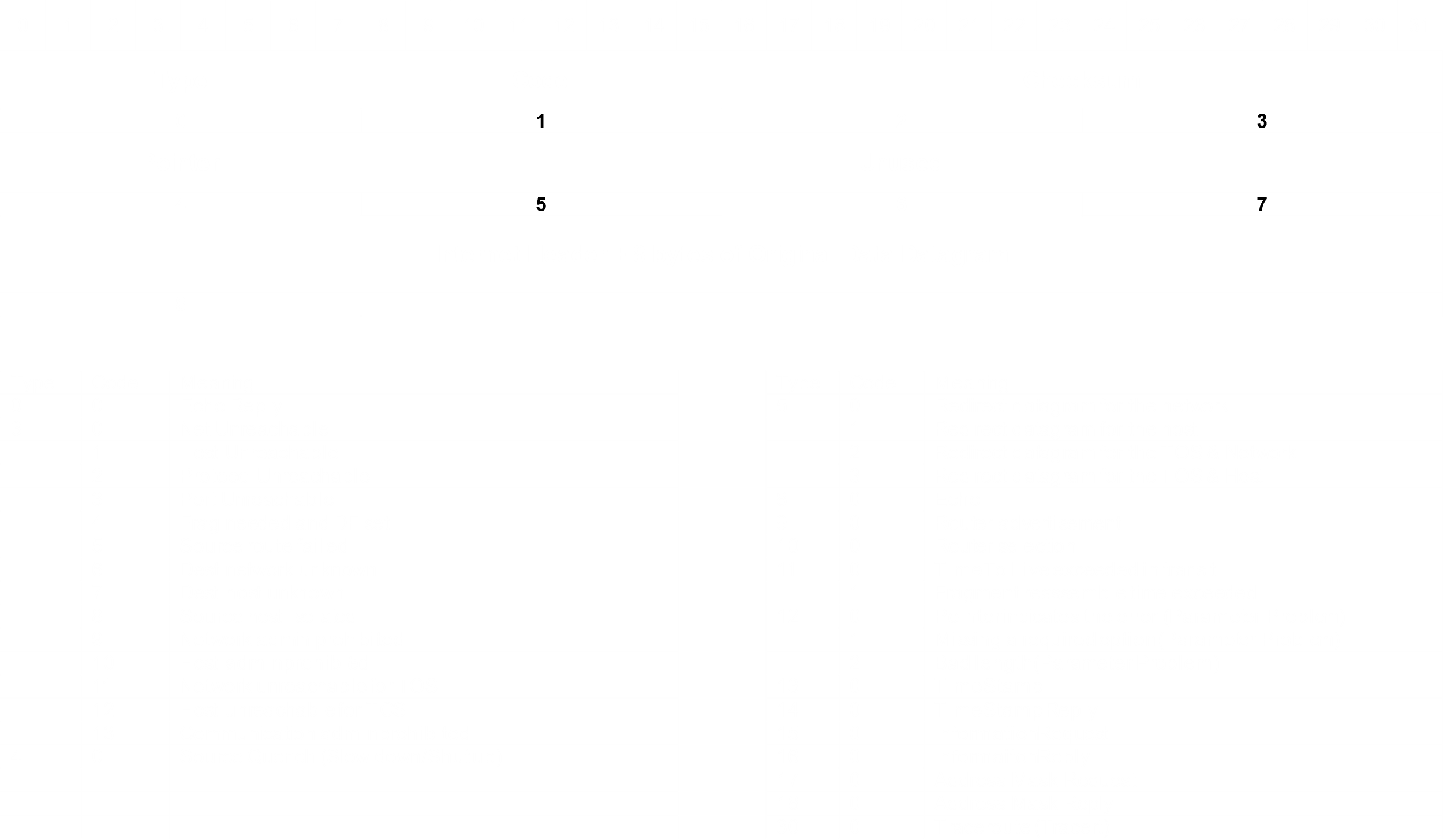
### Redirection



Consider that device A wishes to send data to device B. It should be doing this through router R2. However, due to an error, A sends the data to R1 instead. R1 realizes that the data should have gone to R2, so it sends a **redirection message** to device A to inform it about this mistake. Additionally, it **does not drop** the packet. Instead, it sends it to R2.



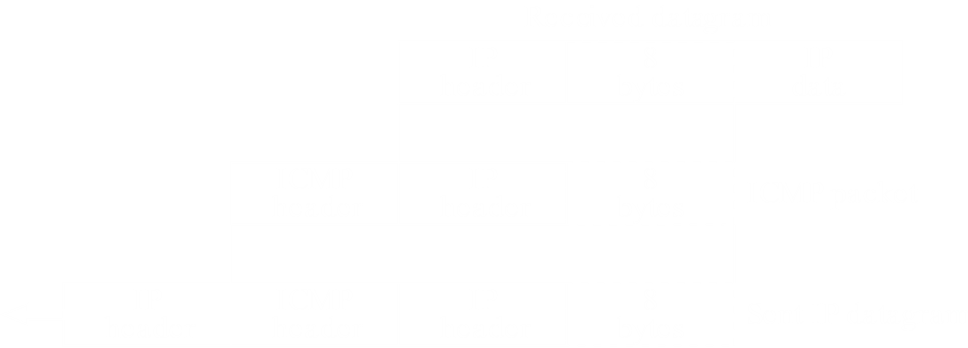
### Message Format



The first field is **type**, which is an 8-bit field. This field defines the error reporting type, one of the types we saw above. Some of these, such as types 15 through 18, are deprecated.

The second field is **code**, which is also an 8-bit field. This specifies the specific error that has occurred. For example, a Type 3 error, destination unreachable, could be because the network is unreachable, the host is unreachable, the port is unreachable, or due to a variety of other reasons.

The third and fourth fields are occupied by the **checksum**, totalling 16 bits.



When creating the IP packet to send back, the ICMP protocol takes the **IP header** from the original IP packet and the **first 8 bytes** of the actual data. To the front of this, it adds the **ICMP header**, which we saw above. All of this acts as the **data section** of the new IP packet, so a **new IP header** is added to all of it before sending the IP datagram.

Notice that the **first 8 bytes** of the original data was included. This is because those 8 bytes contain the **port numbers** and **segmentation information**, originally added by the transport layer of the sender.

## 9.3 Debugging Tools

There are two debugging tools that use ICMP, **Ping** and **Traceroute**.

### Ping

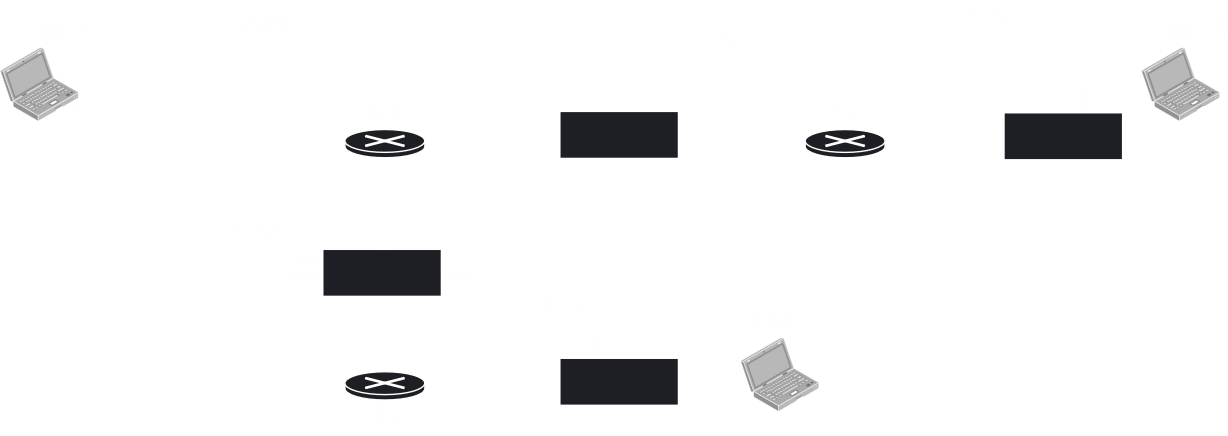
The **Ping** program can be used to check if a host is reachable. Several **ICMP echo requests** are sent to the host, and the host responds with **ICMP echo replies**. The sending-time is inserted into the data section of the message, and this time is used along with the time at which the reply arrives to determine the **round-trip time** (RTT). A **sequence** number is also used to track the order of the packets.

$ ping fhda.edu  
PING fhda.edu (153.18.8.1) 56 (84) bytes of data.  
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp\_seq=0 ttl=62 time=1.91 ms  
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp\_seq=1 ttl=62 time=2.04 ms  
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp\_seq=2 ttl=62 time=1.90 ms  
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp\_seq=3 ttl=62 time=1.97 ms  
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp\_seq=4 ttl=62 time=1.93 ms  
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp\_seq=5 ttl=62 time=2.00 ms  
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp\_seq=6 ttl=62 time=1.94 ms  
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp\_seq=7 ttl=62 time=1.94 ms  
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp\_seq=8 ttl=62 time=1.97 ms  
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp\_seq=9 ttl=62 time=1.89 ms  
64 bytes from tiptoe.fhda.edu (153.18.8.1): icmp\_seq=10 ttl=62 time=1.98 ms  
--- fhda.edu ping statistics ---  
11 packets transmitted, 11 received, 0% packet loss, time 10103 ms  
rtt min/avg/max = 1.899/1.955/2.041 ms

CMD

### Traceroute

The **traceroute** program can be used to trace the route of a packet from source to destination. It makes use of two ICMP packets, the time exceeded packet and the destination unreachable packet, to do this.



First, Host A sends a packet to R1 with a TTL of 1. R1 receives the packet and decrements the TTL. It sees that the TTL is 0 now, so it discards the packet and returns a **time-exceeded** message.

Next, Host A sends a packet to R1 with a TTL of 2. R1 decrements the TTL value and sends the packet to R2. R2 decrements the TTL value, realizes that the value is 0, and sends a time-exceeded message to Host A.

Similarly, the TTL value is incremented one by one and the same process repeats for every hop until the destination is reached. However, when the destination is reached, this will not work. This is because even if the TTL value becomes 0 when the packet reaches the destination, it will not be dropped and no time-exceeded message will be sent. To deal with this, Host A will set the **destination port** for this last hop to an unacceptable value. The destination will see this and will send back a **destination unreachable** message instead.

The traceroute program also calculates the **round-trip time**.

$ traceroute fhda.edu  
traceroute to fhda.edu (153.18.8.1), 30 hops max, 38 byte packets  
1 Dcore.fhda.edu (153.18.31.25) 0.995 ms 0.899 ms 0.878 ms  
2 Dbackup.fhda.edu (153.18.251.4) 1.039 ms 1.064 ms 1.083 ms  
3 tiptoe.fhda.edu (153.18.8.1) 1.797 ms 1.642 ms 1.757 ms

CMD

The traceroute program may seem similar to **strict source routing**, but the key difference is that this is used for debugging. If one of the routers is down for example, strict source routing will not tell us which one. The traceroute program will.