**Lab Exercise 1 – 3 ICMP**

**Objective**

To see how ICMP (Internet Control Message Protocol) is used. ICMP is a companion protocol to IP that helps IP to perform its functions by handling various error and test cases. Review that section before doing this lab.

**Requirements**

**Wireshark:** This lab uses the Wireshark software tool to capture and examine a packet trace.

**traceroute / tracert:** This lab uses “traceroute” to find the router level path from your computer to a remote Internet host. traceroute is a standard command-line utility for discovering the Internet paths that your computer uses. It is widely used for network troubleshooting. It comes pre-installed on Window and Mac, and can be installed using your package manager on Linux. On Windows, it is called “tracert”. It has various options, but simply issuing the command “traceroute www.bit.edu.cn” will cause your computer to find and print the path to the remote computer (here [www.bit.edu.cn](http://www.bit.edu.cn)).

**ping:** This lab uses “ping” to send and receive messages. ping is a standard command-line utility for checking that another computer is responsive. It is widely used for network troubleshooting and comes pre-installed on Window, Linux, and Mac. While ping has various options, simply issuing the command “ping www.bit.edu.cn” will cause your computer to send a small number of ICMP ping requests to the remote computer (here www.bing.com), each of which should elicit an ICMP ping response.

**Turn in**

Hand in the trace file you captured and exercise report including your answers to the questions and figures you drew.

**Step 1: Capture a Trace**

1. Pick a remote web server or other publicly reachable Internet host and use ping to send some ping messages and check that it sends replies. For example, “ping www.bit.edu.cn”. You should see several replies indicating that the pings reached the remote host and were returned.

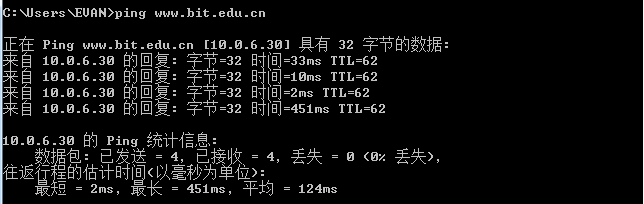


Figure 1: Using ping to bounce messages off a remote host

1. Perform a traceroute to the same remote server to check that you can discover information about the network path. On Windows, type, e.g., “tracert www.bit.edu.cn”. On Linux / Mac, type, e.g., “traceroute www.bit.edu.cn”. If you are on Linux / Mac and behind a NAT (as most home users or virtual machine users) then use the –I option (that was a capital i) to traceroute, e.g., “traceroute –I www.bit.edu.cn”. This will cause traceroute to send ICMP probes like tracert instead of its usual UDP probes; ICMP probes are better able to pass through NAT boxes. A successful example is shown below; save the output as you will need it for later steps. Note that traceroute may take up to a minute to run. Each line shows information about the next IP hop from the computer running traceroute towards the target destination. The lines with “\*”s indicate that there was no response from the network to identity that segment of the Internet path. Some unidentified segments are to be expected. However, if traceroute is not working correctly then nearly all the path will be “\*”s. In this case, try a different remote server, experiment with traceroute.

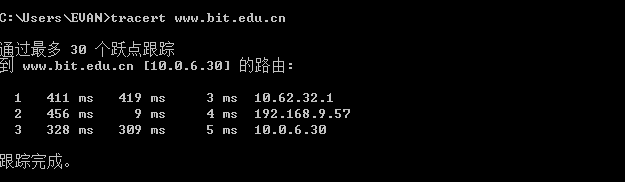


Figure 2: Running traceroute (as tracert on Windows)

1. Launch Wireshark and start a capture with a filter of “**icmp**“. Make sure to check “enable net-work name resolution”. This will translate the IP addresses of the computers and routers sending packets into names, which will help you to recognize the organizations on the network path taken by your packets. Your capture window should be similar to the one pictured below, other than our highlighting. Select the interface from which to capture as the main wired or wireless interface used by your computer to connect to the Internet. If unsure, guess and revisit this step later if your capture is not successful. Uncheck “capture packets in promiscuous mode”. This mode is useful to over-hear packets sent to/from other computers on broadcast networks. We only want to record packets sent to/from your computer. Leave other options at their default values. The capture filter, if present, is used to prevent the capture of other traffic your computer may send or receive. On Wireshark 1.8, the capture filter box is present directly on the options screen, but on Wireshark 1.9, you set a capture filter by double-clicking on the interface.



Figure 3: Setting up the capture options

1. When the capture is started, repeat the ping command you tested, wait a few seconds, and then repeat the traceroute command as well. This time, the ICMP packets sent and received by these two programs will be recorded by Wireshark.
2. After the commands are complete, return to Wireshark and use the menus or buttons to stop the trace. You should have a short trace similar to that shown in the figure below. We have expanded the detail of the ICMP header for a ping request packet in our view. **Be sure to save the output from the** ping **and** traceroute **commands**. You will need it for the later steps.

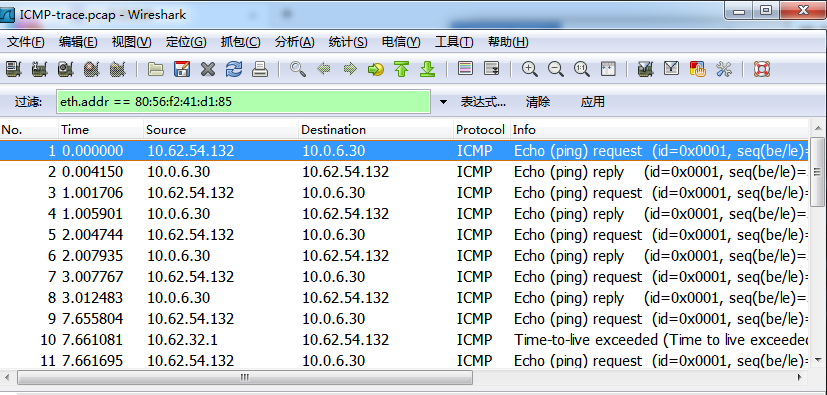


Figure 3: Trace of ping/traceroute traffic showing details of the ICMP header for a ping request.

**Step 2: Echo (ping) Packets**

Start your exploration by selecting an echo (ping) request and reply packet at the start of the trace. Expand the ICMP block (by using the “+” expander or icon) to see the ICMP header and payload details.

**Answer the following questions to demonstrate your understanding of ICMP echo messages:**

1. What are the Type/Code values for an ICMP echo request and echo reply packet, respectively?

Echo request : 8/0

Echo reply : 0/0

1. How do the Identifier and Sequence Number compare for an echo request and the corresponding echo reply?

以第一、二组为例，

1

Echo request Identifier: 0x0001 Sequence number 380(0x017c)

Echo reply Identifier: 0x0001 Sequence number 380(0x017c)

2

Echo request Identifier: 0x0001 Sequence number 381(0x017d)

Echo reply Identifier: 0x0001 Sequence number 381(0x017d)

比较方法是 通过标识符（Identifier）确定是否是所需要的包类型

通过序列号（Sequence number）确定是否是互相对应的一组包

1. How do the Identifier and Sequence Number compare for successive echo request packets?

仍以之前的信息为例

对于连续的请求包，因为包类型不变，其Identifier不变；因为数量不断增加，因此每次Sequence number 递增1

1. Is the data in the echo reply the same as in the echo request or different?

两者发送的数据部分完全相同

**Step 3: TTL Exceeded (traceroute) Packets**

Next, explore traceroute traffic by selecting any Time Exceeded ICMP packet in your trace. Expand the ICMP block (by using the “+” expander or icon) to see the ICMP header and payload details.

**Draw** a picture of one ICMP TTL Exceeded packet to make sure that you understand its nested structure. On your figure, show the position and size in bytes of the IP header, ICMP header with details of the Type/Code and checksum subfields, and the ICMP payload. Within the ICMP payload, draw another rec-tangle that shows the overall structure of the contents of the payload. As usual, your figure can simply show the packet as a long, thin rectangle. To work out sizes, observe that when you click on a protocol block in the middle panel (the block itself, not the “+” expander) then Wireshark will highlight the bytes it corresponds to in the packet in the lower panel and display the length at the bottom of the window.

**Answer the following questions:**

1. What is the Type/Code value for an ICMP TTL Exceeded packet?

Type 11表示超出时间限制

Code 0 表示路由TTL超时 Code 1表示分片重组超时

1. Say how the receiver can safely find and process all the ICMP fields if it does not know ahead of time what kind of ICMP message to expect. The potential issue, as you have probably noticed, is that different ICMP messages can have different formats. For instance, Echo has Sequence and Identifier fields while TTL Exceeded does not.

在每一个ICMP包的包头处有TYPE值声明该包的类型，接收者仅需要 检查这一位就可以知道该包是什么类型进而确定解析方式，包的类型和 对应的编码是事先规定的，因此这种方法是可靠的

1. How long is the ICMP header of a TTL Exceeded packet? Select different parts of the header in Wireshark to see how they correspond to the bytes in the packet.

超时包包头共4字节

类型码1字节，Code 1字节 checksum 2字节

1. The ICMP payload contains an IP header. What is the TTL value in this header? Explain why it has this value. Guess what it will be before you look!

值为1 表示该包已经超出生存周期

**Step 4: Internet Paths**

The source and destination IP addresses in an IP packet denote the endpoints of an Internet path, not the IP routers on the network path the packet travels from the source to the destination. traceroute is a utility for discovering this path. It works by eliciting responses (ICMP TTL Exceeded messages) from the router 1 hop away from the source towards the destination, then 2 hops away from the source, then 3 hops, and so forth until the destination is reached. The responses will identify the IP address of the router. The output from traceroute normally prints the information for one hop per line, including the measured round trip times and IP address and DNS names of the router. The DNS name is handy for working out the organization to which the router belongs. Since traceroute takes advantage of common router implementations, there is no guarantee that it will work for all routers along the path, and it is usual to see “\*” responses when it fails for some portions of the path.

Look at the traceroute portion of the trace, which will have a series of ICMP echo request packets followed by ICMP TTL Exceeded packets. The echo requests are sent from the source (your computer) to the destination whose path is being probed. The TTL Exceeded packets are coming from routers along the path back to your computer, triggered by the TTL field counting down to zero.

**By looking at the details of the packets, answer the following questions:**

1. How does your computer (the source) learn the IP address of a router along the path from a TTL exceeded packet? Say where on this packet the IP address is found. You might proceed by looking at an echo packet to see the source and destination IP addresses. The routers along the path will have a different IP address, and this address will be present on the TTL Exceeded packet. If you are unsure, you can examine the traceroute text output to see the IP addresses of routers and look for these addresses on the TTL Exceeded packets.

本地主机不断向目标IP发送Echo request，在沿途的路由器中产生路由器超时，导致TTL exceeded因此图中的路由器会向本地发送ICMP TTLexceeded的确认包，本地主机由此可获得图中的路由器地址

1. How many times is each router along the path probed by traceroute? Look at the TTL Ex-ceeded responses and see if you can discern a pattern.

除目标路由器之外的中间路由器每个都发送三个TTL exceeded包到本 地主机

1. How does your computer (the source) craft an echo request packet to find (by eliciting a TTL Exceeded response) the router N hops along the path towards the destination? Describe the key attributes of the echo request packet. The echo request packets sent by traceroute are probing successively more distant routers along the path. You can look at these packets and see how they differ when they elicit responses from different routers.

主机向目标不断发送Echo request 包的时候包中序号不断增加，路径中的路由器由近及远所回复的超时包中嵌套的包的序号也依次增加， 由此可判断是第几跳时的路由器，同时本机收到超时包的顺序也可以作为判断第几跳的依据

Using the traceroute output, **sketch a drawing** of the network path. Show your computer (lefthand side) and the remote server (righthand side), both with IP addresses, as well as the routers along the path between them numbered by their distance on hops from the start of the path. You can find the IP address of your computer and the remote server on the packets in the trace that you captured. The output of traceroute will tell you the hop number for each router.

To finish your drawing, label the routers along the path with the name of the real-world organization to which they belong. To do this, you will need to interpret the domain names of the routers given by traceroute. If you are unsure, label the routers with the domain name of what you take to be the organization. Ignore or leave blank any routers for which there is no domain name (or no IP address).

主机 10.62.54.132

Hop1：10.62.32.1

本地局域网内IP

Hop2:192.168.9.57

本地局域网内IP

Dst：10.0.6.30

www.bit.edu.cn

Figure 4: Logical structure of the network