Lab Assignment - 6

IP

Open the Wireshark packet sniffer and perform the following:

We will run traceroute with different packet sizes and analyze attributes related to IP header.

For Windows systems, download pingplotter from Internet Connection Troubleshooting | PingPlotter ,the size of the ICMP echo request message can be explicitly set in pingplotter by selecting the menu item Edit-> Options->Packet Options. For Unix systems, give the following command and capture packets:

%traceroute gaia.cs.umass.edu 2500

(2500 is packet size)

Then, start capturing packets in wireshark or (for Windows) in pingplotter, start up pingplotter and enter the name of a target destination in the "Address to Trace Window." Enter 3 in the "# of times to Trace" field. Trace 4 different sizes of packets , i.e. off 56 bytes, 100 bytes, 2500 bytes and 3000 bytes. (set up packet size in pingplotter or give packet size in traceroute command) You can also save different sized packet captures and analyze later about following questions :

You need to analyze ICMP packets, and if not seen, then only you can look for UDP packets.

1. What is the value in the upper layer protocol field in the IP header ?
2. What is the size of the datagram payload ? And what is the size of the header ?
3. Observe and note down if the datagram is fragmented or not?
4. Sort the packets according to IP addresses and observe that ,Which fields in IP datagram are always changing in ICMP packets (or UDP packets) sent and received ?
5. Which fields in the IP header must remain constant ? Why ?
6. Observe and not down values of Identification and TTL fields of packets. (at least 6 packets' data)

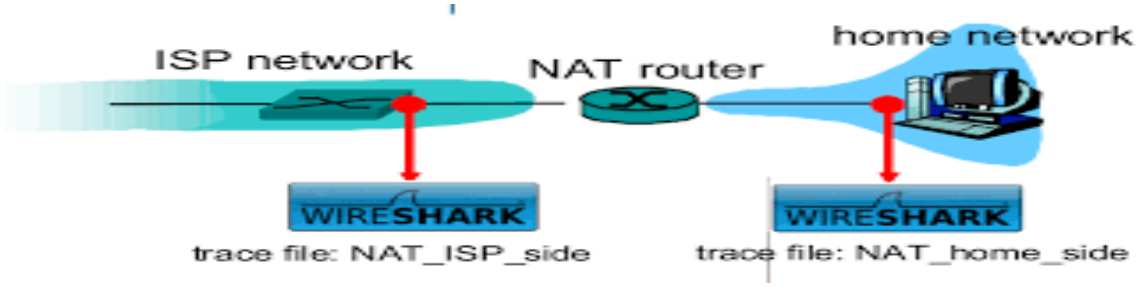
NAT

In this lab, we'll capture packets from a simple web request from a client PC in a home network to a www.google.com server. Within the home network, the home network router provides a NAT service. Figure shows the WireShark trace collection scenario. Analysis the trace files NAT home side.pcap and NAT ISP side.pcap to answer the following questions:

For simplicity concerns, apply filter "tcp.stream eq 2" to both pcap files, this will give you the same tcp stream on both captured files.

1. What is the client ip on the home side? And what is the client ip on the ISP side. What's the relationship between them?
2. For the first packet of this tcp stream, is there anything changed in the tcp header between home side and ISP side? If so, name the header field(s).
3. Focus on the same packet, is there anything changed in the ip header? If so, name the header field(s), and explain why they are changed. Now we focus on a udp stream, apply filter "udp.stream eq 1" to the home side and "udp.stream eq 0" to the ISP side.
4. For the first packet of this udp stream, is there anything changed in the udp header? If so, name the header field(s). Is any other udp header or data changed? If not, explain why udp checksum changed.
5. For the tcp and udp streams we discussed above, for all the packets that the client sent, is the tcp/udp source port ever changed by the router? If not, is it mandatory to keep the same source port before and after NAT translation?

Recall the mechanism of NAT translation, answer the following question:



ICMP

Execute the command "ping gaia.cs.umass.edu" in cmd(Windows) or terminal(Mac Os, Linux), use WireShark to capture the generated ICMP packet (you can use filter "icmp" in WireShark) and answer the following questions:

1. Why is it that an ICMP packet does not have source and destination port numbers?
2. Choose one of the ping request packets sent by your host, what are the ICMP type and code numbers? Find the corresponding ping reply, what are the type and code numbers?\

Open your web browser and try accessing http://gaia.cs.umass.edu:81/, use WireShark to capture the packets and answer the following questions:

3. During the browser trying loading the page, did your host receive any ICMP packets? If yes, what are the type and code of these ICMP packets?
4. Apart from the ICMP headers, what is in the data field of these ICMP packets?
5. Imagine the case when a NAT router receives one of the above ICMP packets on its external interface, how does the router know which internal host to forward this packet to?