Monitoring and Analysis of Network Traffic using Snoop

Overview

In this laboratory, you will learn how to examine network traffic for analysis and troubleshooting purposes using the `snoop` utility.

There are many different utilities that can be used to monitor network traffic and snoop is just one of them. The process of monitoring traffic is usually called "sniffing" and the corresponding tools are called sniffers.

Sniffers usually work by forcing the network card to enter "promiscuous" mode in which a network card can intercept all packets going through a network cable. Since this operation may cause privacy problems, only root (the privileged administrator account) can access this functionality.

You can prepare for the lab by reading the snoop man page - you can do so by login into one of the Sun workstations and issuing the man snoop command. Some of options that will be most useful to you are -t, -v and -p. You should definitely look at the examples section of the man page too.

Further information on sniffing can be found at the Sniffing FAQ located at the following address: http://www.robertgraham.com/pubs/sniffing-faq.html

Important: A Crash Tutorial on Unix Shell

This is a simple help section for those of you that are not comfortable with the unix shell.

The output of the *snoop* command is a series of text lines. Since it sometimes can be very large, you should use the *more* command to display the information in pages so that you can read a page before you advance to the next one. This is done by "feeding" the output of a command to the input of more as in the following example:

```
# snoop | more
```

The vertical bar in the command line is used to specify that the output of the first command (*snoop* in our case) will be "fed" to the input of the second command (which is *more* in our case), and what you will see on the screen will be the output of the second program. (In this case, the second command will simply print its input one page at a time and wait for the user to press a key when advancing between pages).

Another important command that you will need to use with the "|" construct is the grep command. The grep command is used to search for lines containing a string. For example, if you wanted to display only the lines in the output of snoop that contained the "ICMP Time exceeded" string, you would simply type:

```
# snoop | grep "ICMP Time exceeded"
```

The grep command in the above example will filter the output of snoop and show only the lines that contain the specified string.

You can use the "|" construct in a cascaded manner as shown in the following example:

```
# snoop -v | grep "ICMP Time exceeded" | more
```

This series of commands simply means:

Show me the lines in the output of "snoop -v" that contain the "ICMP Time exceeded" message one page at a time, and wait for me to press a key before advancing to the next pages.

1 Examining ICMP traffic: ping

Objective

To learn about how the ping program works by analyzing network traffic.

Background

The ping command in Unix can be used to see if some machine is connected to the network. When you "ping" a machine, an "ICMP Echo request" packet is sent to the destination. According to the specifications, the destination has to reply with an "ICMP Echo reply" packet.

Procedure

Each workstation in the lab has a file called /snoop.cap. This file contains data that was previously recorded with the snoop utility. There was one ping command issued during the recording of this file. Please examine the contents of this file with the help of the snoop utility and answer the following questions. (Using “snoop –i snoop.cap” combine with other necessary options)

Questions

1. What is the ip address of the computer that issued the ping request?

2. What is the ip address of the computer that replied to the ping request?

3. What was the time at which the ping command was issued?

2 Examining ICMP traffic: *traceroute*

Objective

To learn about how the *traceroute* program works by analyzing network traffic.

Background

The traceroute command in Unix can be used to discover the path packets follow when trying to reach some destinations. *traceroute* does this by sending UDP or ICMP packets to the destination that expire after 1 hop, 2 hops, etc. Again according to the specification, once a packet expires, the router (on which the packet expired) has to notify the sender with an "ICMP Time exceeded" message. The *traceroute* program is able to discover each router along the path by noting theip addresses of the nodes that notify it about expiration of the first, second, ... and so on packets.

Procedure

Each workstation in the lab has a file called /snoop.cap. This file contains data that was previously recorded with the snoop utility. There was one *traceroute* command issued during the recording of this file. Please examine the contents of this file with the help of the snoop utility and answer the following questions.

Questions

1. What is the ip address of the computer that issued the *traceroute* command?

2. What is the destination ip address of the *traceroute* command?

3. List the ip addresses of all the routers that replied to the *traceroute* command with an "ICMP Time exceeded" message.

3 Examining HTTP traffic

Objective

The objective in this section is to learn about the HTTP protocol by analyzing network traffic.

Background

Web browsers retrieve pages by sending HTTP requests to web servers. The HTTP protocol requires a TCP connection to port 80 on a web server (80 is the default port, but can be changed). When a client wants to get the contents of some web page, it connects to the server and asks for the page, to which the server replies with the content of that page. Such a client-server dialogue will be usually fragmented into more than one packet.

Hint: Packets that are part of an HTTP communication can be recognized by the HTTP identifier in the snoop output.

Procedure

Each workstation in the lab has a file called /snoop.cap. This file contains data that was previously recorded with the snoop utility. There were a number of HTTP requests issued during the recording of this file. Please examine the contents of this file with the help of the snoop utility and answer the following questions.

Questions

1. What is the ip address and DNS name of the first web site that has been browsed? At what time was it browsed?

2. What was the URL of the web page that has been requested?

3. What was the User-Agent (that is the web browser program) that issued this request?

4. What was the length of the web page that was retrieved?

4 Examining FTP traffic

Objective

The objective of this section is to learn about the FTP protocol by analyzing network traffic.

Background

FTP is also a TCP based protocol. There are two ports used during a connection. Port 20 (ftp-data) is the port used for transferring data, while port 21 (ftp) is the port to which the commands are sent. When you connect to an FTP site with a simple FTP client, all the commands that you type and most of the visual feedback that you get on the screen is sent to or received from port 21 of the server. The actual data file that you send to or receive from the server is transmitted to/from port 20.

Procedure

Each workstation in the lab has a file called /snoop.cap. This file contains data that were previously recorded with the snoop utility. There were two FTP connections established during the recording of this file. Please examine the contents of this file with the help of the snoop utility and answer the following questions. (Look for the key work “ftp” in the packets”)

Questions

1. What was the first FTP server to which the user connected? Was it an anonymous FTP? If so, please write also the user name and password.

2. What was the second FTP server to which the user connected? Was it an anonymous FTP? If so, please write also the user name and password.

3. What were the names of the files downloaded from the anonymous FTP server?

4. When you list the contents of a directory on the FTP server, which server port is used to send the directory list to the client?