

Lab 3 – tShark and ftp

Introduction

This lab focuses on the practical application of command line packet tracing on a Windows machine. This lab can be completed in pairs, however collaboration in class and working in groups outside of class is acceptable for working through the hurdles and roadblocks that you may run into during the lab work.

The purpose of this lab is to continue with packet captures by learning more about what expected traffic looks like and translating that into a step-by step analysis to communicate the signature or behavior of the packets being focused on.

Specs and Expected and New Knowledge

- Wireshark
- tShark
- tcpdump
- ftp.....
- 2 Ubuntu VMs, 1 windows workstation VM

Part 1 - Environment

All tools that are installed or demonstrated in class are for education purposes and must only be used on controlled systems where you have expressed access and/or permission, and never to attack a real-world system without legal authority.

You *have* permission to conduct active and passive reconnaissance within the course virtual environment.

TASK 1: Open Wireshark and Capture DNS and ICMP Packets

Goal: Use Wireshark to capture ICMP packets from a Internet based connection and document findings

1. Open Wireshark and begin capture on interface that is used for Internet traffic
2. Open terminal window.
3. On the CLI, run the ping command, set it to run forever, and ping a well known domain or your choosing. (www.amazon.com, www.comcast.net, etc)
4. After about 10 seconds, stop the ping and the packet capture.
5. Screen shot interface information for the machine you are using for this lab.
6. Locate the 1st packet in the packet capture that is related to your ping command. HINT: it's not ICMP.

7. You must prove that you have located the 1st packet. You prove it by showing that the source/destination IP address matches your computer and that the packet contains to correct flags and information to be the correct-first packet.
8. Locate and document the DNS reply packet
9. Locate 1 ICMP packet from packet capture, prove it, and provide a walk through of what you see. (similar to lab 2)

TASK 2: Configuring and using tShark

Goal: Configure a Windows Desktop Environment to locate and execute tshark.exe.

- A. Follow the steps on Hands-on Project 3-1: Setting the PATH Variable to Work with TShark on page 110 of your textbook.

Goal: Use TShark to Capture ICMP Packets

- B. Follow the steps on Hands-on Project 3-2: Using TShark to Capture ICMP Packets on page 111 of your textbook.

----END OF WORK COMPLETED IN-CLASS LAB TIME----

Part 2 – Virtualized Environment

Ubuntu: You will be using 2 network Ubuntu workstations for the part 2 of the lab. You will be installing an Insecure ftp service on 1 ubuntu machine and you will be a) capturing packets and b) attempting to connect with the ftp server from the other Ubuntu machine.

TASK 3: Install insecure FTP server on Ubuntu

1. From the Ubuntu command line: type **vsftpd**. If this service is installed, it will respond. If it is not, the terminal will tell you to install it. Type the command it suggests.
2. Once the install is complete, you can test the ftp server locally.
3. To test that the daemon/service is running type: **ps -ef | grep "ftp"**
 - a. The result should be 2 entries (1 is your grep command and the other is the process line for the service running).
4. Now, we need to make sure that authorized users can login to the ftp server. Edit the /etc/vsftpd.conf file to change 1 line: remove the hashtag '#' from the line:

```
write_enable=YES
```

5. Save the file and restart the service by typing:

```
sudo systemctl restart vsftpd.service
```

6. Now that the ftp server is up, running and ready to receive connections, we need to install Wireshark on this same linux machine. Type wireshark in the terminal and see if it needs to be installed. If so, follow the suggested apt get/install command.

7. During the install you will see a popup regarding who can use wireshark, select NO and click enter.
8. Once the install is complete test that wireshark works by typing

```
sudo wireshark&
```

TASK 4: Capture FTP Traffic

1. To setup this capture, be sure you start on the machine that is your ftp server. Turn on Wireshark and begin capturing traffic.
2. On your client machine, type

```
ftp x.x.x.x (where the x's are the numbers of the IP address of  
your ftp server.)
```

3. Your username and password is the same as what you use to login to the linux ftp server machine.
4. Once you successfully login, type help. Look at the available commands, look around, and when you are finished, type exit.
5. Return to Wireshark on your ftpserver machine and stop the capture.
6. Locate the initial connection of the ftp packets, identify where the user name and password were captured.

Analysis

Using the guidance from this lab activity and your knowledge of Wireshark, discuss in detail how you were able to generate network traffic and how you were able to identify the traffic that was uniquely yours. Screen shots, appropriately cropped and labelled, will assist in communicating your challenges and results. Provide screen shots for evidence of the work completed; and to assist in your description of the output from the various commands you used. Clearly label all screen shots.

1. In part 1, using tshark, what were the purposes of each option that was used to run the tshark command?
2. In part 2, what challenges, if any, did you have in locating the ftp packets?
3. In the capture of the ftp packets, what did you notice about the password capture? Explain.
4. What was the biggest “ah-hah!” moment during this lab, and why was it an “ah-hah!” moment. If you didn’t have an “ah-hah!” moment, what was your biggest challenge with this lab? Why?
5. FTP is an inherently insecure protocol. What are some solutions/steps that could be taken to make FTP more secure. Remember, as part of the lab you specifically changed the setting to have authenticated users. Why would that be risky and make the server/host more vulnerable to attack?

