LAB EXERCISE

4

COMP3134

Introduction to Cyber Security

Week: 5

Objective(s):
Tracing network communication and analyze network activity

Learning Outcome(s):
Identify and filter various protocols and network ports
Implement various tactics to attack a network or application

Table of Contents

Contents

Summary

Goal: Tracing network communication and analyze network activity

In Effort To: Identify and filter various protocols and network ports before and after implement various tactics to attack a network or application

A. Clone GitHub Repo

Clone course GitHub repo to any location on your local machine
Navigate to the location above and create a folder named **wk5**
Use this local folder created above to create all the files necessary for this Lab Exercise

B. Packet Sender

What is Packet Sender?

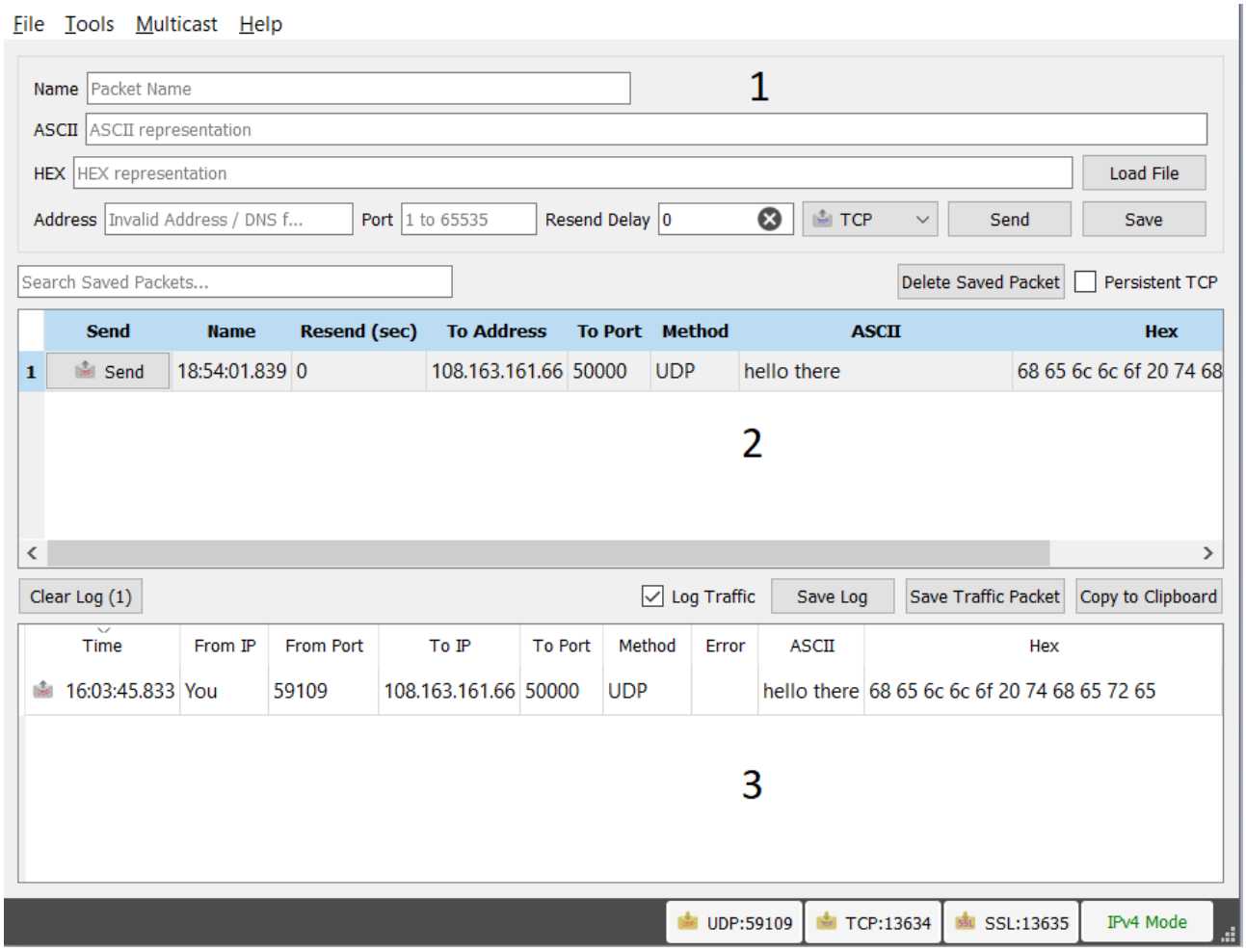
Packet Sender is an open source utility to allow sending and receiving TCP, UDP, and SSL (encrypted TCP) packets. The mainline branch officially supports Windows, Mac, and Desktop Linux (with Qt). Other places may recompile and redistribute Packet Sender. Packet Sender is free and licensed GPL v2 or later. It can be used for both commercial and personal use.

Packet Sender Uses

- Controlling network-based devices in ways beyond their original apps
- Test automation (using its command line tool and/or hotkeys)
- Testing network APIs (using the built-in TCP, UDP, SSL clients)
- Malware analysis (using the built-in UDP, TCP, SSL servers)
- Troubleshooting secure connections (using SSL).
- Testing network connectivity/firewalls (by having 2 Packet Senders talk to each other)
- Stress-testing a device (using intense network generator tool)
- Tech support (by sending customers a portable Packet Sender with pre-defined settings and packets)
- Sharing/Saving/Collaboration using the Packet Sender Cloud service.

C. Getting Started with Packet Sender

When you launch Packet Sender, you will be presented with the following screen.



Panels

1. This area is where you create a packet
    a. You create the packet name and data in hex and ASCII
    b. You input the destination, delay and protocol of the packet
    c. Click "Send" to immediately send. Click "Save" to send later.
2. Is the area of your saved packets
    a. Clicking on the Send button will send or resend the packet
        i. During packet resending, there will be a button to cancel.
    b. You can double-click to directly edit fields in this table.
    c. Select multiple by holding down the shift button and selecting multiple packets. The click "Multi-Send" in Panel 1 area.
3. Is the log of packet traffic

Other Notable Areas/Notes

- In the bottom right, there are UDP, TCP, and SSL server status and port(s). You can click to activate or deactivate these. Packet Sender supports binding to any number of ports.
- During packet resending, there will be a button to cancel all resends.
- Fields can be rearranged by going to File => Settings then going to the Display tab.
- Please check your firewall. Windows aggressively blocks TCP-based servers. Packet Sender will still work if the firewall blocks it, but it can't receive unsolicited TCP-based packets.

Hotkeys / Keyboard Shortcuts

The fields at the top can be navigated using CTRL+1, CTRL+2, etc, up to CTRL+8 (send button).
On Mac, the shortcut key is Command.

Logging and Saving

To save the log of your Packet Traffic area, click on the “Save Log” button
To save a selected packet, select the packet and click “Save Traffic Packet”
To copy the ASCII data of a packet, select the packet and click “Copy to Clipboard”

Intense Traffic Generator

Go to Tools => Intense Traffic Generator on the menu bar
This will continuously send packets to the specified destination
Please be aware that a log of this intense traffic will not be recorded.

Documentation

To view the Pocket Sender documentation, please navigate to the following url:
https://packetsender.com/documentation

D. Sending TCP Packets to Your Machine Using Packet Sender

Open WireShark and capture all traffic for either your Wi-Fi Interface or Ethernet Interface (depending on your Network connection)

Determine your local machines network public IPv4 address (go to whatismyip.com)

Using Packet Sender send the following 2 TCP packets

1. Send your first name to the port 10,000 to the public IPv4 address of your local machine
2. Send your last name to the port 80 to the IPv4 address of your local machine

Take a screenshot of the Packet Traffic Panel of your Packet Sender application.
Name the screenshot **packetsender_tcp_1.png**

Go to WireShark and apply the filter log to filter only captured TCP traffic to the destination IP address of your local machine's public IPv4 address

Take a screenshot of the applied filter. Name the screenshot **wireshark_tcp_1.png**

E. Analyzing TCP Packets

Take 1 TCP packet from step D and view details. If you do not have any TCP traffic because they have been blocked, choose any TCP packet

Take a screenshot and highlight the following information in any way you choose

Source
Source Port
Destination
Destination Port
Sequence number
Acknowledge number

Name the screenshot **analyzing_tcp_packets_1.png**

F. Sending UDP Packets to Your Machine Using Packet Sender

Using Packet Sender send the following 2 UDP packets

1. Send your favorite course name to any port between 10,000 and 15,000 to the IPv4 address of your local machine
2. Send your favorite sport to any to a port greater than 50,000 to the IPv4 address of your local machine

Take a screenshot of the Packet Traffic Panel of your Packet Sender application.
Name the screenshot **packetsender_udp_1.png**

Go to WireShark and apply the filter log to filter only captured UDP traffic to the destination IP address of your public IPv4 address and to ports greater or equal to 10,000

Take a screenshot of the applied filter. Name the screenshot **wireshark_udp_1.png**

G. Analyzing UDP Packets

Take 1 UDP packet from step F and view details. If you do not have any UDP traffic because they have been blocked, choose any UDP packet
Take a screenshot and highlight the following information in any way you choose

Source
Source Port
Destination
Destination Port

Name the screenshot **analyzing_udp_packets_1.png**

H. Deeper Look at TCP Packets

Complete the following:

- In WireShark, filter all TCP packets where the reset flag has been set
  - Take a screenshot named **wireshark_advanced_tcp_1.png**
- Start a new Capture. Apply the *capture filter* to only capture tcp packets on the http protcol
  - Take a screenshot of the WireShark application
    Name the screenshot **wireshark_advanced_tcp_2.png**

I. Simplified Firewall Rules: UFW

Overview

Iptables provide a kernel level ip filtering mechanism which allow you to make routing decisions on IP packets.

There is a simplified firewall tool named Uncomplicated Fire Wall (UFW). It is not as flexible but is easier to configure for common scenarios.

UFW comes pre-installed in most Linux distributions and provides a basic default firewall which allow you to easily turn on and off basic services.

Using UFW

To view firewall rules, type the following command:

```
ufw status
```

To view which applications are using which ports, type the following command

```
ufw app list
```

To view the manual of how to use, go to the following page
http://manpages.ubuntu.com/manpages/bionic/man8/ufw.8.html

J. Changing the SSH Port

One action you can take to secure your server is to change the default ssh port number. The following steps will outline you can do so.

1) Connect to your Remote Server
2) Open the file /etc/ssh/sshd_config
3) Find the line with the value **Port 22** or **#Port 22**
4) Change the port value to **2222**
    a. Take a screenshot of this change. Name is **ssh_port_change_1.png**
5) Add the port to your server firewall. Associate it with the appropriate protocol.
6) Restart the sshd service
    a. Remember, to action a service, it takes the following format
        i. **service {service_name} {action}**
        service is a command

        {service_name} is the service you'd like to change such as apache2, mysql, etc

        {action} is a list of support actions such start, stop, etc
7) Open a new GitBash window and log into your ssh service using the new port. Research how to specify the port using the ssh command
8) Create a text file named **ssh_port_change.txt** and copy and paste the two commands for Step 6) and 7)
9) Change the ssh port back to its default value of 22
10) Remove the added firewall rule from Step 5)
11) Confirm that you can log into ssh using the default port and NOT by the changed port

K. Commit and Upload Changes to GitHub repo

Commit the changes to your repo by:

1. Opening a GitBash window and ensure that it is connected to your local machine
2. Navigate to local repository directory location
3. Add all the files completed in this Lab Exercise
4. Commit the changes
5. Push the changes to your GitHub course repo

Please refer to the instructions in the last section of Lab Exercise 1