

Protocol Analysis: Capturing Packets

This lab should take 1 – 1.5 hours. Level: intermediate

Objectives

This lab will introduce you to “packet sniffing,” a method by which we can capture packets being sent between computers as they communicate. As a network administrator you can use this method to help evaluate the performance of your network by identifying bottlenecks and slower performing servers or sections of your network. You can also use it to check the security of your network. As a graphic demonstration of this, you will configure an FTP server and observe the login packet interchange. You will see that each communication may consist of several packets that are exchanged between the two computers and you will see the potential for security leaks and how to gauge potential abuse of the network by users.

Overview & Prerequisites

You will first install a program called Wireshark. This is an open source application freely available on the Internet that allows you to capture packets as they appear at the network adaptor card. This means that you will be able to see all header information on the packet from each of the OSI layers. (Normally these headers are stripped off so that the only portion remaining is the data payload.) You will use the software to view complete packets and locate each layer’s header, from the physical layer to the application layer. Doing so will help you to better understand network traffic and identify things that are “out of order.” Using this program you will:

- 1) Analyze simple protocols and learn about the software interface and the information it contains;
- 2) Observe, analyze and reconstruct specific packet interchanges between a computer and a server; and
- 3) Monitor the login process to an FTP server. This will include searching for the login information in the Wireshark output.

For the first two parts of this lab, you will need a single computer with an Internet connection. For the last part, you will need two computers, one of which should have an active FTP server loaded on it. Instructions are provided in part 3 for setting up an FTP server on one computer and connecting to it from a second computer using an FTP client.

Procedure

To obtain the software that you will use for this lab, go to www.wireshark.org and download it to your workstation. Once downloaded, you can install the software and accept all defaults. The program includes a helper program called WinPCap, which will install after Wireshark is installed.

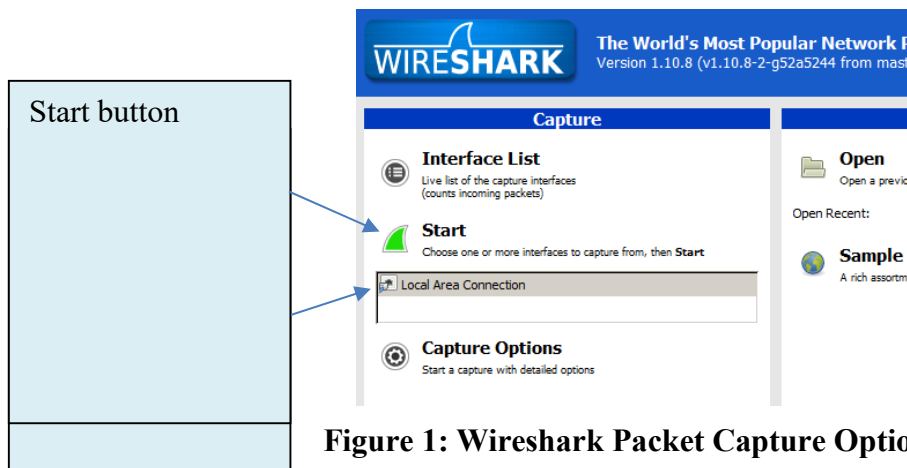


Figure 1: Wireshark Packet Capture Options

Below the menu, the capture window is divided into three distinct areas. The top is a listing of all packets received—the packet list pane; the middle provides the details of a packet selected in the packet list pane and is called the packet details pane; and the bottom, called the packet bytes pane, shows the hexadecimal details of the selected packet and will highlight its (selected) fields. Figure 2 illustrates this and shows some captured packets.

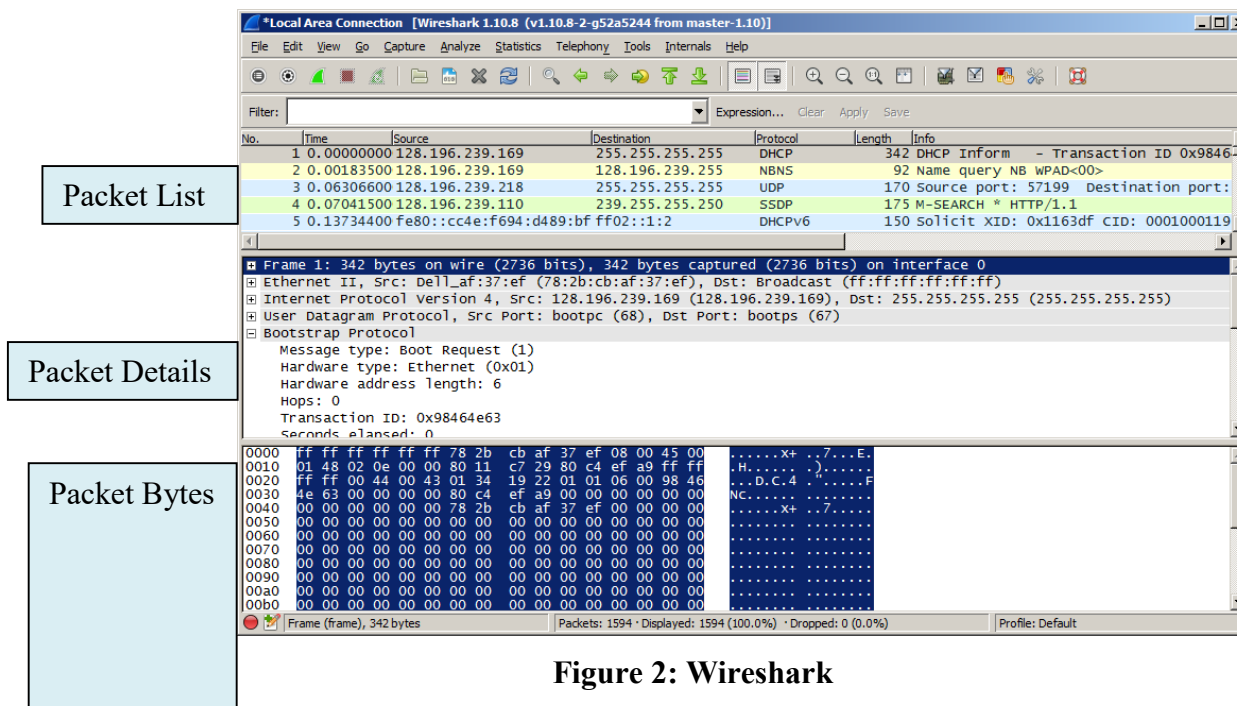


Figure 2: Wireshark

You can see in Figure 2 that multiple packets were captured and the first packet is selected in the packet list pane. In the packet details pane, you can see the Ethernet frame header, the IP

Find a TCP packet in the packet list pane and select it. In the packet details pane, click on the “+” next to the word “Frame.” When this part of the packet opens, you will see some summary information that Wireshark logs about every packet that it captures. Now open each subsequent section of the packet beginning with “Ethernet II.” You should be able to find the portions of each packet corresponding to figures 3a through 3c within the packet details section (though the sizes of each section may not always be apparent without closer examination).

Preamble	Start of Frame	Destination Address	Source Address	Type	Data	FCS	Flag
7 bytes	1 byte	6 bytes	6 bytes	2 bytes	46 - 1500 bytes	4 bytes	8 bytes

Figure 3a: An Ethernet II Frame Layout

Version Number	Header Length	Service Field	Total Length	ID	Flags	Fragment Offset	Time to Live	Next Protocol	Header Checksum	Source IP Address	Destination IP Address	Data
4 bits	4 bits	8 bits	16 bits	16 bits	3 bits	13 bits	8 bits	8 bits	16 bits	32 bits	32 bits	Variable

Figure 3b: The IP Header Layout

Source Port	Destination Port	Sequence Number	ACK Number	Header Length	Unused	Flags	Window Size	Header Checksum	Urgent Pointer	Options	Data
16 bits	16 bits	32 bits	32 bits	4 bits	3 bits	9 bits	16 bits	16 bits	16 bits	32 bits	Variable

Figure 3c: The TCP Header Layout

Figure 3a includes 20 bytes that are processed in the hardware and will not be seen in the packet details pane. These are the preamble (7 bytes), the Start of Frame (1 byte), the Frame Check Sequence (FCS, 4 bytes), and the final Flag (8 bytes).

Part 2: Finding specific packet sequences

For this part you need a workstation that is connected to the Internet and one that receives its IP address from a DHCP server. You should have Wireshark installed on your workstation from part 1. In step 1 you will observe the packets required to make and break a connection.

Step 1 Observing a TCP connection

- 1) Ensure that your capture options are set as before and begin another capture session.
- 2) After the capture session has begun, open a web browser on your workstation, allow the web page to finish loading, and then stop the packet capture session.
- 3) Look for the first three TCP packets in the packet list pane. TCP packets have a green background color (depending on your settings) and are easily recognized.

These three packets should be listed as [SYN], [SYN, ACK] and [ACK]. This 3-packet interchange builds a connection between two computers. You should notice that the destination

- 2) **NOTE: DO NOT DO THIS STEP IF YOU ARE ON A REMOTE CONNECTION OR YOU WILL LOSE YOUR SESSION. ONLY DO THIS IF YOU ARE CONDUCTING THIS LAB ON YOUR OWN PC. SKIP TO STEP 3 to release and renew in the same step.**

Begin a Command Prompt window. Next, to release the existing IP address, enter the `ipconfig /release` command at the command prompt. See Figure 4. (Note: if your computer has IPv6 configured, you will see the configured IPv6 address; you can release these using the `ipconfig /release6` command.)



```
C:\Users>ipconfig /release

Windows IP Configuration

Ethernet adapter Ethernet0:

    Connection-specific DNS Suffix  . : 
    Link-local IPv6 Address . . . . . : fe80::68cc:a3b6:43ab:aaf6%3
    Default Gateway . . . . . : 

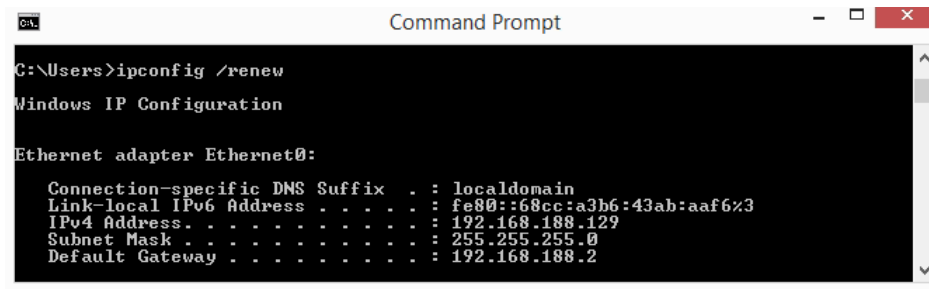
Tunnel adapter Local Area Connection* 3:

    Connection-specific DNS Suffix  . : 
    IPv6 Address . . . . . : 2001:0:9d38:6abd:28a9:622:3f57:437e
    Link-local IPv6 Address . . . . . : fe80::28a9:622:3f57:437e%4
    Default Gateway . . . . . : 

C:\Users>
```

Figure 4: Releasing a DHCP IP Address Lease

- 3) As soon as you see that your IP address was released (shown as empty or 0.0.0.0, depending on your system) enter the `ipconfig /renew` command at the command prompt. See Figure 5.



```
C:\Users>ipconfig /renew

Windows IP Configuration

Ethernet adapter Ethernet0:

    Connection-specific DNS Suffix  . : localdomain
    Link-local IPv6 Address . . . . . : fe80::68cc:a3b6:43ab:aaf6%3
    IPv4 Address . . . . . : 192.168.188.129
    Subnet Mask . . . . . : 255.255.255.0
    Default Gateway . . . . . : 192.168.188.2

C:\Users>
```

Figure 5: Renewing a DHCP IP Address Lease

- 4) Wait until the renewal process has completed (you receive an IP address). Then, stop the packet capture in Wireshark. Next, click on the column in the Packet List pane marked, "Protocol." This will sort the entries in order of protocol.

that the destination address in each of the 4 packets is a broadcast address¹. It should be obvious to you why the first two packets are broadcasted, but what about the last two? Can you explain this?

Step 3: Following an HTTP stream

Let's have a closer look at a request/response interchange that requests a web site. Follow these steps to obtain a fresh set of packets:

- 1) Ensure that your capture options are set as before and begin another capture session. You can discard the previous session or save it to a file.
- 2) Open Internet Explorer on your workstation, return to Wireshark and begin a packet capture session.
- 3) Type in a URL and after the page loads, return to Wireshark and stop the packet capture.
- 4) Find the packet with comments in the "Info" column saying "GET / HTTP/1.1" and select it. Right click this packet and click "Follow TCP stream" from the popup menu. See Figure 6.

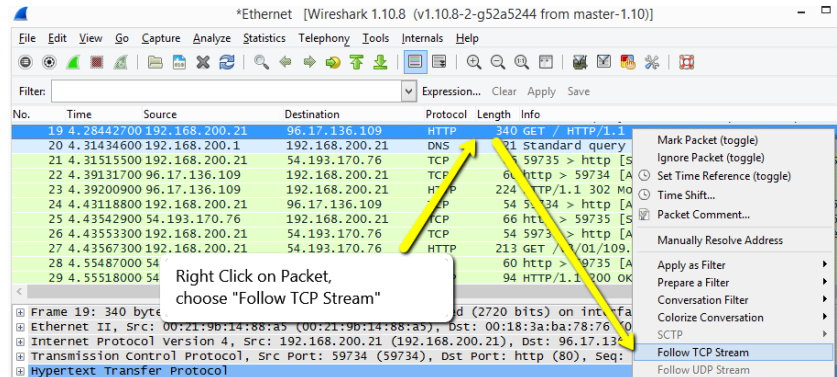


Figure 6: Follow TCP Stream

- 5) A new window will open with the details of the http exchange. The request and acknowledgements from your workstation are in red, and the responses are in blue and should resemble Figure 7.

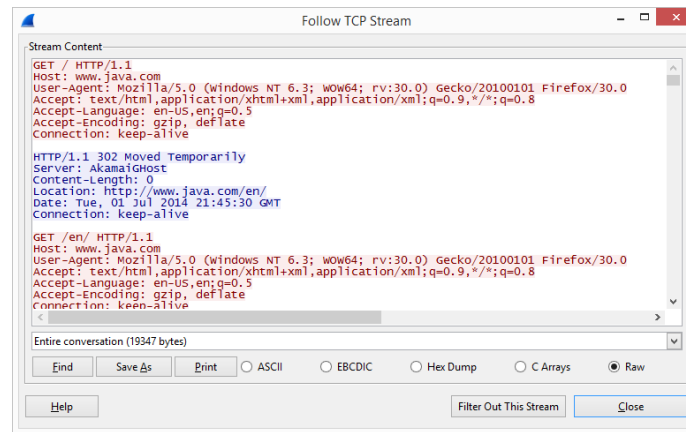


Figure 7: Raw TCP Stream Data

- 6) At the bottom of this window are some options for saving this file for later reference. Click the “Close” button to return to the main window and you will notice that only the TCP and HTTP packets have been retained, since a filter was created based on your action of following the TCP stream. Now select File > Export > Objects > HTTP. See Figure 8. In the resulting window, find the Hostname you visited (second column; in our case, it was www.java.com) and the content-type corresponding to text/html. Then, click the “Save As” button. Save the file (with a “.html” extension) on your desktop.

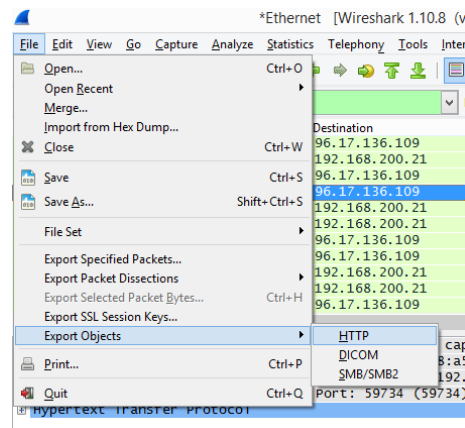


Figure 8: Exporting TCP Stream (HTTP)

- 7) Minimize all windows and find the file you just saved on the desktop and open it with a web browser. If the web page contains a number of secondary files, such as image, css,



Figure 9: Java.com (comparison)

Part 3: Viewing an FTP transfer

We will now look at the file transfer between an FTP client and an FTP server. You will need a second computer on your network capable of providing file transfer services (an FTP server). The easiest way to set up an FTP server is to download the open source program, Filezilla. It has both a server version that you can use to store files and a client version, which is used to access your server from another computer.

Step 1: Setting up the FTP Server

Download the Filezilla server from <http://filezilla-project.org> and install it on one computer. You can accept all the defaults for this demonstration, but you will need to create a user and assign a home directory to that user. Make sure you give the user a password but do not enable SSL. For this lab, we chose the username “johndoe” and a password “secret”. See Figure 10.

Note: Your firewall may need to be configured to allow connections to FileZilla Server.

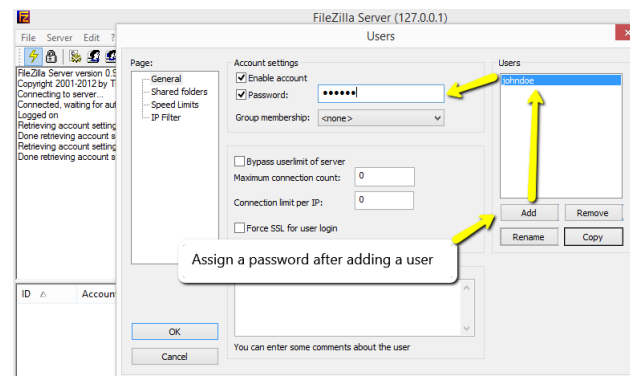


Figure 10: Creating an FTP user in FileZilla

Download the Filezilla client from the same website as above and install it on a second

- 1) Open Wireshark on the client, ensure that your capture options are set as before and begin another capture session.
- 2) Connect to the FTP server by typing in its IP address, user name and password in the text boxes at the top of the client software, then press “Quickconnect”. This is shown in Figure 11.

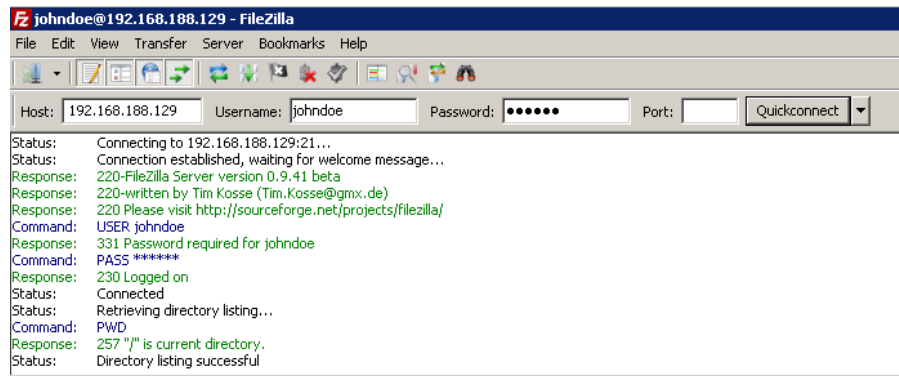


Figure 11: Connecting to the FTP Server

- 3) Stop the packet capture.

Look for the FTP packets in the Protocol column (or apply a filter to show only ftp protocol packets). In the “Info” column they will say “Request: ...” and “Response: ...” You should notice that the username and password are displayed for you in this column in clear text. This is shown in Figure 12.

If you have never seen a password revealed in a packet sniffer, it can be a real eye opener. Although we know that FTP servers are inherently not secure, this demonstration should make you think about the security of other types of logins. Try this: if you have a domain controller on your network, try logging on to it from a workstation and sniffing the packets as you do so. Are you able to find the password? (Hopefully not.) Now set up a database server for which the security setting is controlled by the operating system and do the same thing. If the security is not configured correctly, not only will you be able to find the login information (user name and password), but data will be passed in the clear also.

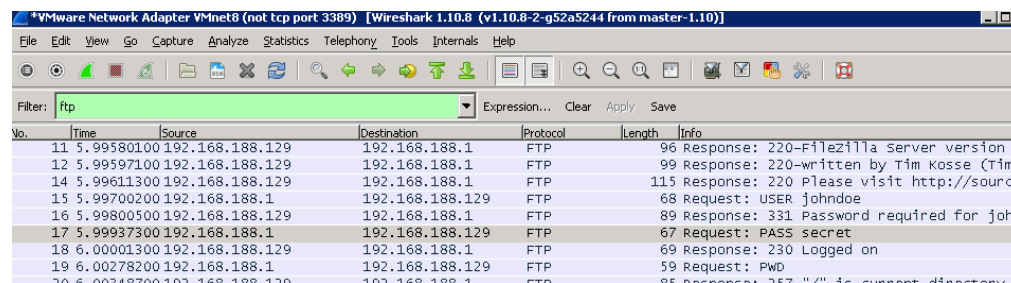


Figure 12: An FTP Login Sequence in Wireshark

Questions

1. Packet sniffing can be a controversial subject. Discuss any issues related to ethics that might arise when an organization monitors the electronic activity of its employees.
2. You looked at packets captured during a web page request. What might this be useful for?
3. Most computers are connected together with switches (rather than hubs). How does this affect the packet capturing process?
4. Discuss how sniffing packets from wireless networks might differ from wired networks. Use the Internet to search for wireless packet sniffers. Where might someone go to sniff wireless packets and possibly obtain some “juicy” information?