Packet Inception, Dissecting Network Traffic With Wireshark

The purpose of this lab is to provide experience with dissecting traffic in Wireshark. We will have the chance to pull objects out of previously captured network traffic along with pulling data from live traffic.

We have been provided with a packet capture file that contains data from an unencrypted web session. There is an image embedded that needs to be used as evidence of improper network usage. The Security manager thinks the user is sending messages hidden behind the image. Using Wireshark, apply filters to locate and extract the evidence.

ⓘ If you wish to take a more exploratory approach to this lab, I have posted the overall tasks to accomplish. For a more detailed walkthrough of how to complete each step, look below each task in the solution bubble.

Tasks

Utilizing `Wireshark-lab-2.zip` in the optional resources, perform the lab to the best of your ability.

Task #1

`Open a pre-captured file (HTTP extraction)`

In Wireshark, Select File → Open → , then browse to Wireshark-lab-2.pcap. Open the file.

Task #2

`Filter the results.`

Now that we have the pcap file open in Wireshark, we can see quite a lot of traffic within this capture file. It has around 1171 packets total, and of those, less than 20 are HTTP packets specifically. Take a minute to examine the pcap file, become familiar with the conversations being had while thinking of the task to accomplish. Our goal is to extract potential images embedded for evidence. Based on what has been asked of us, let's clear our view by filtering for HTTP traffic only.

Apply a filter to include only HTTP (80/TCP) requests.

▶ **Click to show answer**

Please note how this removes any additional TCP or IP datagrams from the window and allows us to focus on communication solely with HTTP. From here, we can see several basic HTTP datagrams containing the GET method and 200 OK responses. These are interesting because we can now see that a client requested several files, and the server responded with an OK. If we select one of the OK responses, we can follow that stream and see the data transfer over TCP. Let's give this a shot.

Task #3

`Follow the stream and extract the item(s) found.`

So now that we have established there is HTTP traffic in this capture file, let's try to grab some of the items inside as requested. The first thing we need to do is follow the stream for one of the file transfers. With our `http` filter still applied, look for one of the lines in which the Web Server responds with a "200 OK" message which acts as an acknowledgment/receipt to a users' GET request. Now let's select that packet and follow the TCP stream.

▶ **Click to show answer**

Now that we validated the transfer happened, Wireshark can make it extremely easy to extract files from HTTP traffic.

Cheat Sheet
Resources
? Go to Questions

Table of Contents

Introduction
Network Traffic Analysis
Networking Primer - Layers 1-4
Networking Primer - Layers 5-7

Analysis
The Analysis Process
Analysis in Practice

Tcpdump
Tcpdump Fundamentals
Capturing With Tcpdump (Fundamentals Labs)
Tcpdump Packet Filtering
Interrogating Network Traffic With Capture and Display Filters

Wireshark
Analysis with Wireshark
Familiarity With Wireshark
Wireshark Advanced Usage
Packet Inception, Dissecting Network Traffic With Wireshark
Guided Lab: Traffic Analysis Workflow
Decrypting RDP connections

My Workstation
OFFLINE
Start Instance
∞ / 1 spawns left

We can check to see if an image file was pulled down by looking for the `JFIF` format in the packets. The JPEG File Interchange Format `JFIF` will alert us to the presence of any JPEG image files. We are looking for this format because it is the most common file type for images alongside the png format. With that in mind, we will likely see an image in this format for our investigation.

Check for the presence of JFIF files in the HTTP traffic.

▶ **Click to show answer**

Now that we are sure image files were transferred between the suspicious host and the server let's grab them out of the capture. To do this, we need to export the objects out of the HTTP traffic.

▶ **Click to show answer**

At this point, we should now have the image files that our security manager requested us to capture if they existed. They can now examine the file to determine if any data was hidden within it.

Live Capture and Analysis

In the scenario above, we practiced filtering on a pre-captured file. Now it's time to do some live packet captures. We will connect to the academy lab and sniff traffic live from a host in the network to complete this portion.

After we analyzed the pcap traffic, the Security Manager has come back and confirmed the user was smuggling data out of the network via the images. He is requesting that we now capture traffic to determine if anything else is going on from the user's host `172.16.10.2`. We will need to start a capture, categorize and filter the data, and extract anything significant to the investigation.

Connectivity to Lab

Access to the lab environment to complete this part of the lab will be a bit different. We are using XfreeRDP to provide us desktop access to the lab virtual machine to utilize Wireshark from within the environment.

We will be connecting to the Academy lab like normal utilizing your own VM with a HTB Academy VPN key or the Pwnbox built into the module section. You can start the FreeRDP client on the Pwnbox by typing the following into your shell once the target spawns:

Code: bash

```
xfreerdp /v:<target IP> /u:htb-student /p:HTB_@cademy_stdnt!
```

You can find the `target IP`, `Username`, and `Password` needed below:

- `Click below in the Questions section to spawn the target host and obtain an IP address.`
  - `IP ==`
  - `Username == htb-student`
  - `Password == HTB_@cademy_stdnt!`

Start a Wireshark Capture

We will be sniffing traffic from the host we logged into from our own VM or Pwnbox. Utilizing interface `ENS224` in Wireshark, let the capture run for a few minutes before stopping it. Our goal is to determine if anything is happening with the user's host and another machine on the corporate or external networks.

Self Analysis

Before following these tasks below, take the time to step through our pcap traffic unguided. Use the skills we have previously tested, such as following streams, analysis of conversations, and other skills to determine what is going on. Keep these questions in mind while performing analysis:

- `How many conversations can be seen?`
- `Can we determine who the clients and servers are?`
- `What protocols are being utilized?`

- Is anything of note happening? (ports being misused, clear text traffic or credentials, etc.)

In this lab, we are concerned with the hosts 172.16.10.2 and 172.16.10.20 while performing the following steps. In our analysis, we should have noticed some web traffic between these hosts and some FTP traffic. Let's dig a bit deeper.

FTP Analysis

When examining the traffic, we captured, was any traffic pertaining to FTP noticed? Who was the server for that traffic?

Were we able to determine if an authenticated user was performing these actions, or were they anonymous?

Filter the results

Now that we have seen some interesting traffic, let's try and grab the file off the wire.

Examine the FTP commands to determine what you need to inspect, and then extract the files from ftp-data and reassemble it

▶ **Click to show answer**

HTTP Analysis

We should have seen a bit of HTTP traffic as well. Was this the case for you?

If so, could we determine who the webserver is?

What application is running the webserver?

What were the most common method requests you saw?

Follow the stream and extract the item(s) found

Now attempt to follow the HTTP stream and determine if there is anything to extract.

▶ **Click to show answer**

Summary

By the end of this lab, we should be able to open previously captured .pcap files, apply display filters, follow streams, and extract items from the capture file. Experiment with ways to capture new traffic and applying filters to find specific traffic. To check our understanding, answer the questions below with the traffic you capture on your own.

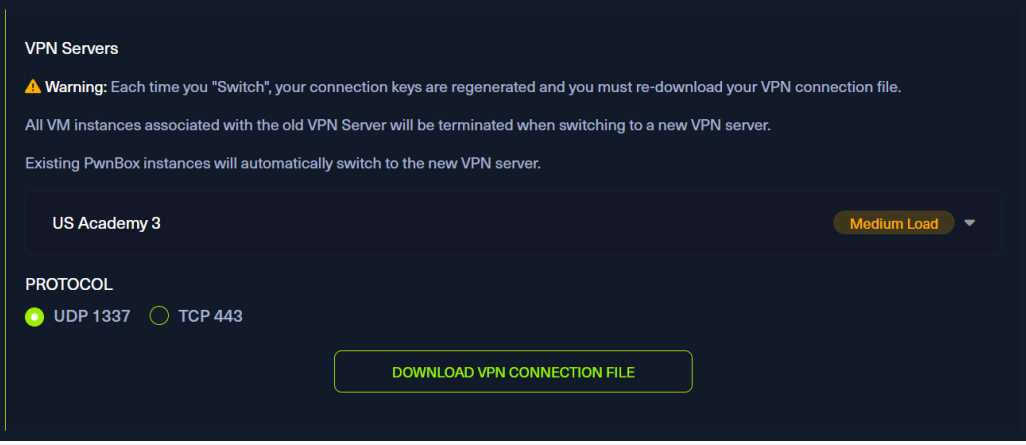
Check your understanding:

• What filters or expressions did you use? Were they effective?

• How did these filters affect the traffic you could see within the capture?

• How can utilizing these features be beneficial to you and your mission?

• What filter would you use if you wanted to only see TCP traffic from the client?



Connect to Pwnbox
Your own web-based Parrot Linux instance to play our labs.

Pwnbox Location

UK 138ms

Terminate Pwnbox to switch location

Start Instance

∞ / 1 spawns left

Waiting to start...

Enable step-by-step solutions for all questions

Questions

Answer the question(s) below to complete this Section and earn cubes!

Cheat Sheet

Download VPN Connection File

Target(s): Click here to spawn the target system!

RDP to with user "htb-student" and password "HTB_@cademy_stdnt!"

+ 2 What was the filename of the image that contained a certain Transformer Leader? (name.filetype)

Rise-Up.jpg

Submit Hint

+ 0 Which employee is suspected of performing potentially malicious actions in the live environment?

bob

Submit Hint

Previous Next

Mark Complete & Next