# Wireshark Advanced Usage

In this section, we will cover some advanced usage with Wireshark. The project developers have included many different capabilities ranging from tracking TCP conversations to cracking wireless credentials. The inclusion of many different plugins makes Wireshark one of the best traffic analysis tools.

## Plugins

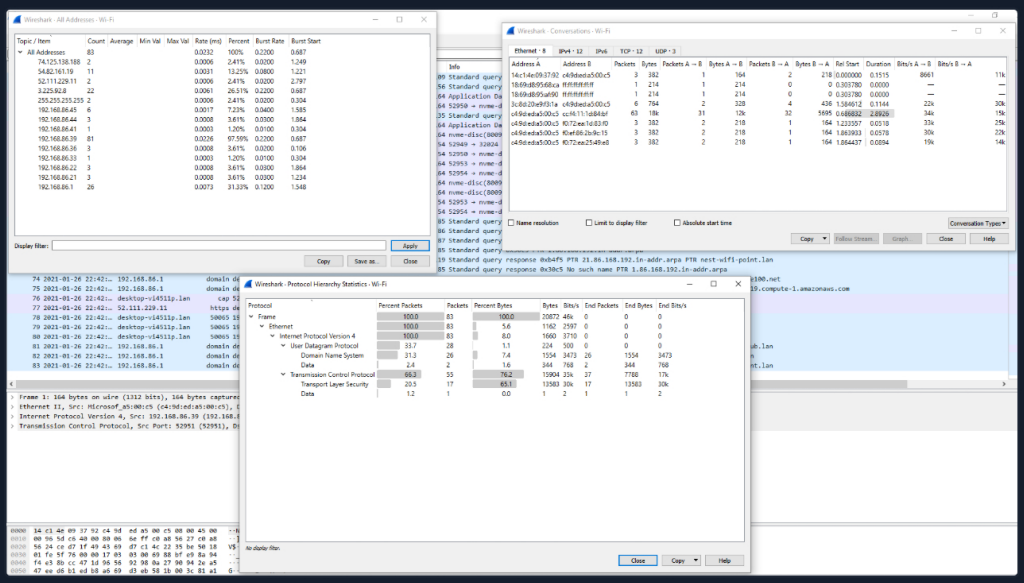
The analyze and statistics radials provide a plethora of plugins to run against the capture. In this section, we will work through a couple of them. We would cover all of which Wireshark offers, but sadly, it is simply not achievable in an introductory module. I urge everyone to experiment and play as we go through this journey.

### The Statistics and Analyze Tabs

The Statistics and Analyze tabs can provide us with great insight into the data we are examining. From these points, we can utilize many of the baked-in plugins Wireshark has to offer.

The plugins here can give us detailed reports about the network traffic being utilized. It can show us everything from the top talkers in our environment to specific conversations and even breakdown by IP and protocol.

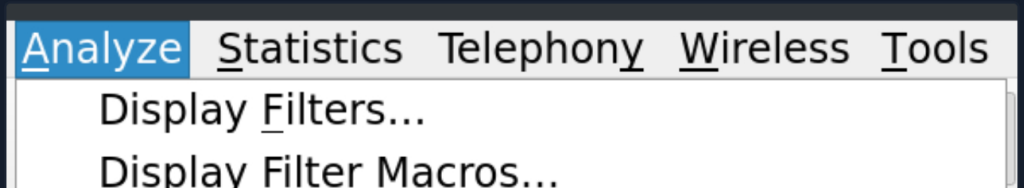
### Statistics Tab



### Analyze

From the Analyze tab, we can utilize plugins that allow us to do things such as following TCP streams, filter on conversation types, prepare new packet filters and examine the expert info Wireshark generates about the traffic. Below are a few examples of how to use these plugins.

### Analyze Tab




Cheat Sheet
Resources
Go to Questions

Table of Contents

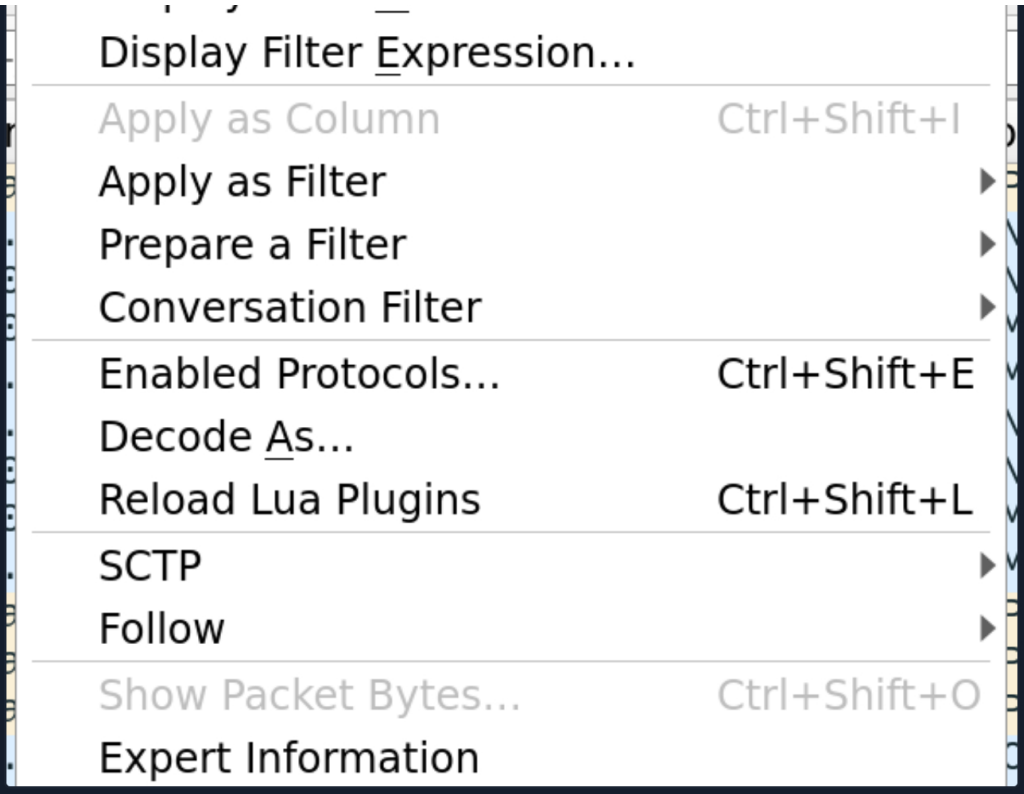
Introduction
Network Traffic Analysis
Networking Primer - Layers 1-4
Networking Primer - Layers 5-7

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



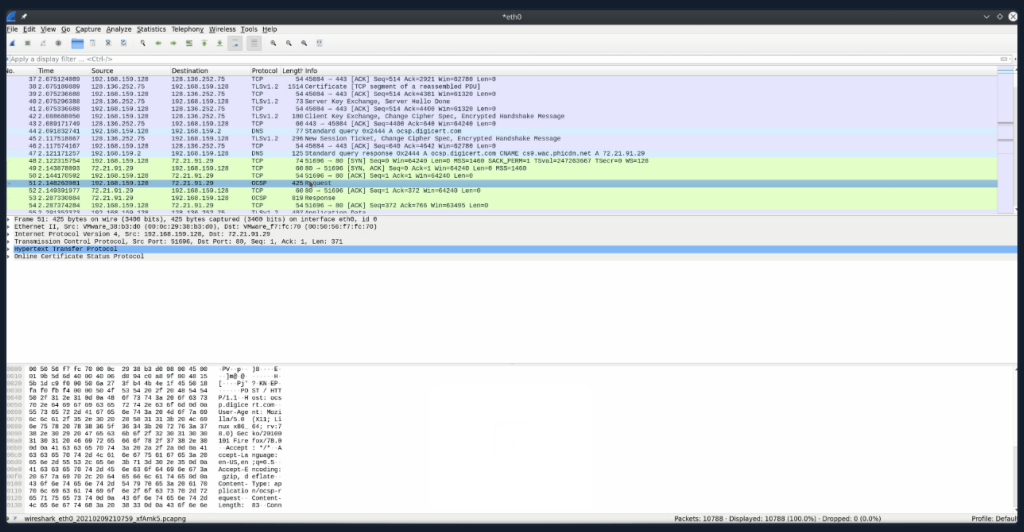

### Following TCP Streams

Wireshark can stitch TCP packets back together to recreate the entire stream in a readable format. This ability also allows us to pull data (`images, files, etc.`) out of the capture. This works for almost any protocol that utilizes TCP as a transport mechanism.

To utilize this feature:

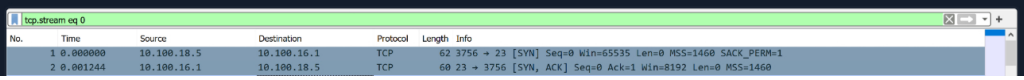
- right-click on a packet from the stream we wish to recreate.
- select follow → TCP
- this will open a new window with the stream stitched back together. From here, we can see the entire conversation.

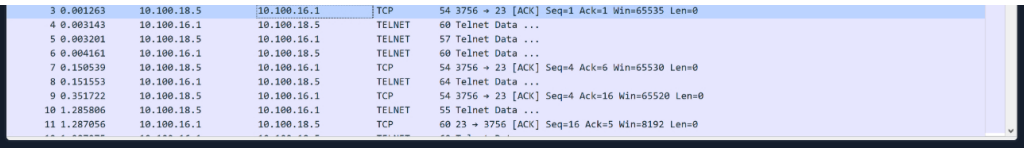
### Follow A Stream Via GUI



Alternatively, we can utilize the filter `tcp.stream eq #` to find and track conversations captured in the pcap file.

## Filter For A Specific TCP Stream





Notice that the first three packets in the image above have a full TCP handshake. Following those packets, we can see the stream transferring data. We have cleared anything not related out of view by utilizing the filter, and we now can see the conversation in order.

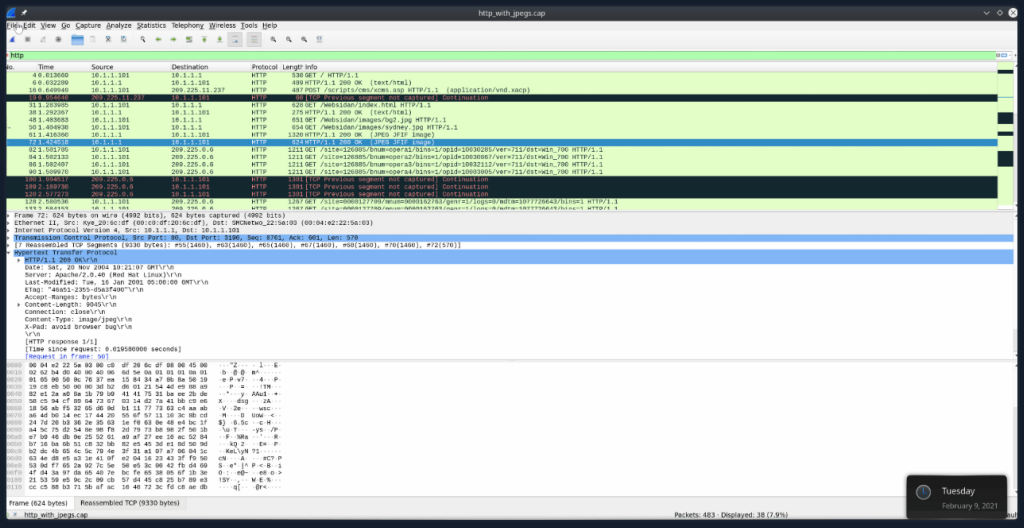
### Extracting Data and Files From a Capture

Wireshark can recover many different types of data from streams. It requires you to have captured the entire conversation. Otherwise, this ability will fail to put an incomplete datagram back together. If we want a more in-depth understanding of how this capability works, check out the Networking 101 Module or research TCP/IP fragmentation.

To extract files from a stream:

- `stop your capture.`
- `Select the File radial → Export → , then select the protocol format to extract from.`
- `(DICOM, HTTP, SMB, etc.)`

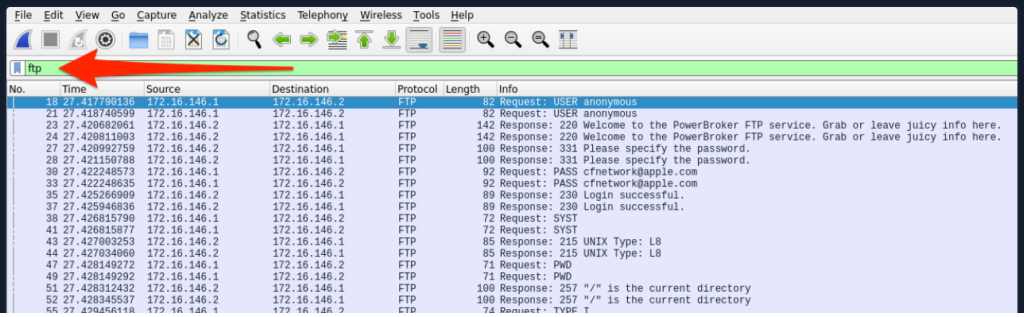
### Extract Files From The GUI

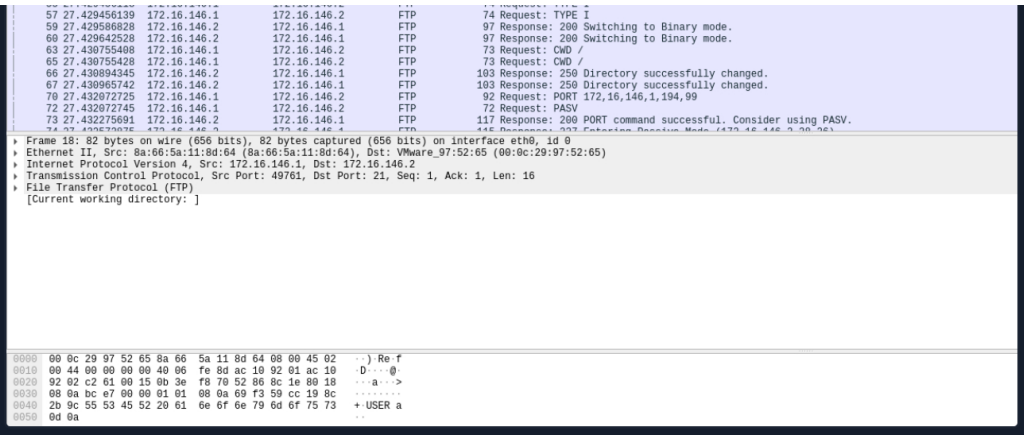


Another exciting way to grab data out of the pcap file comes from FTP. The File Transfer Protocol moves data between a server and host to pull it out of the raw bytes and reconstruct the file. (image, text documents, etc.) FTP utilizes TCP as its transport protocol and uses ports `20 & 21` to function. TCP port 20 is used to transfer data between the server and host, while port 21 is used as the FTP control port. Any commands such as login, listing files, and issuing download or uploads happen over this port. To do so, we need to look at the different FTP display filters in Wireshark. A complete list of these can be found here. For now, we will look at three:

- `ftp - Will display anything about the FTP protocol.`
  - `We can utilize this to get a feel for what hosts/servers are transferring data over FTP.`

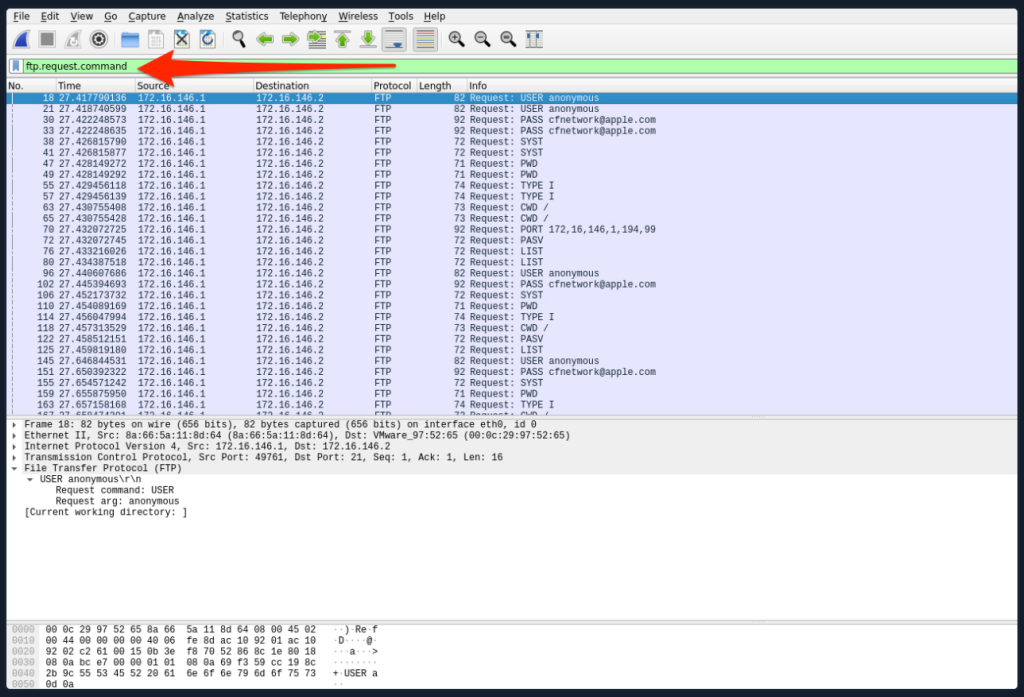
### FTP Disector





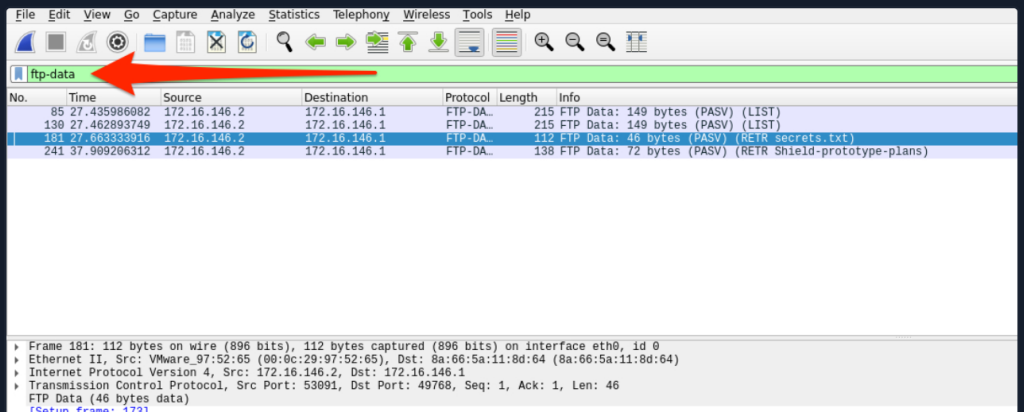

- ftp.request.command - Will show any commands sent across the ftp-control channel ( port 21 )
    - We can look for information like usernames and passwords with this filter. It can also show us filenames for anything requested.

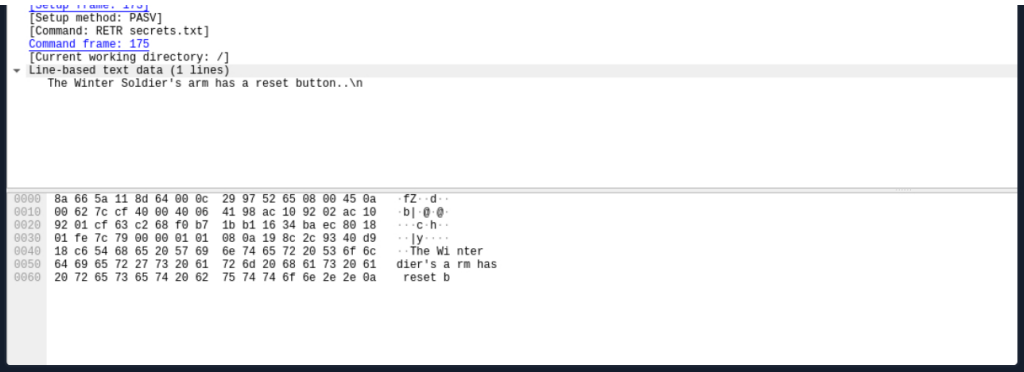
## FTP-Request-Command Filter




- ftp-data - Will show any data transferred over the data channel ( port 20 )
    - If we filter on a conversation and utilize ftp-data, we can capture anything sent during the conversation. We can reconstruct anything transferred by placing the raw data back into a new file and naming it appropriately.

## FTP-Data Filter






Since FTP utilizes TCP as its transport mechanism, we can utilize the `follow tcp stream` function we utilized earlier in the section to group any conversation we wish to explore. The basic steps to dissect FTP data from a pcap are as follows:

1. Identify any FTP traffic using the `ftp` display filter.
2. Look at the command controls sent between the server and hosts to determine if anything was transferred and who did so with the `ftp.request.command` filter.
3. Choose a file, then filter for `ftp-data`. Select a packet that corresponds with our file of interest and follow the TCP stream that correlates to it.
4. Once done, Change `"Show and save data as"` to `"Raw"` and save the content as the original file name.
5. Validate the extraction by checking the file type.

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

Pwnbox Location

UK 137ms

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

+ 0 Which plugin tab can provide us with a way to view conversation metadata and even protocol breakdowns for the entire PCAP file?

Statistics

Submit Hint

+ 0 What plugin tab will allow me to accomplish tasks such as applying filters, following streams, and viewing expert info?

Analyze

Submit Hint

+ 0 What stream oriented Transport protocol enables us to follow and rebuild conversations and the included data?

TCP

Submit Hint

+ 0 True or False: Wireshark can extract files from HTTP traffic.

True

Submit Hint

+ 0 True or False: The ftp-data filter will show us any data sent over TCP port 21.

False

Submit Hint

Previous Next

Mark Complete & Next

Powered by HACKTHEBOX