Conversation with:
Noah Westerlund
Incident Analyst

Cyber Defense Analyst Challenge Chat Transcript

Noah sent:
Hello, @playerone. I'm Noah Westerlund, an Incident Analyst here at the TryCyber Consulting.

I'll be your mentor while you're interning with us. 😎

Noah sent:
I'm going to be showing you how to perform the tasks of a **Cyber Defense Analyst**. I usually perform my daily tasks using a workstation like the **Ubuntu 22.04 Linux** workstation you are looking at now.

Noah sent:
The tasks you will be doing today will require you to use **Wireshark** and the **Report Form**. Wireshark can be intimidating if you have not used it before. So let me know if you want a short tutorial on it.

Noah sent:
Are you ready to get started?

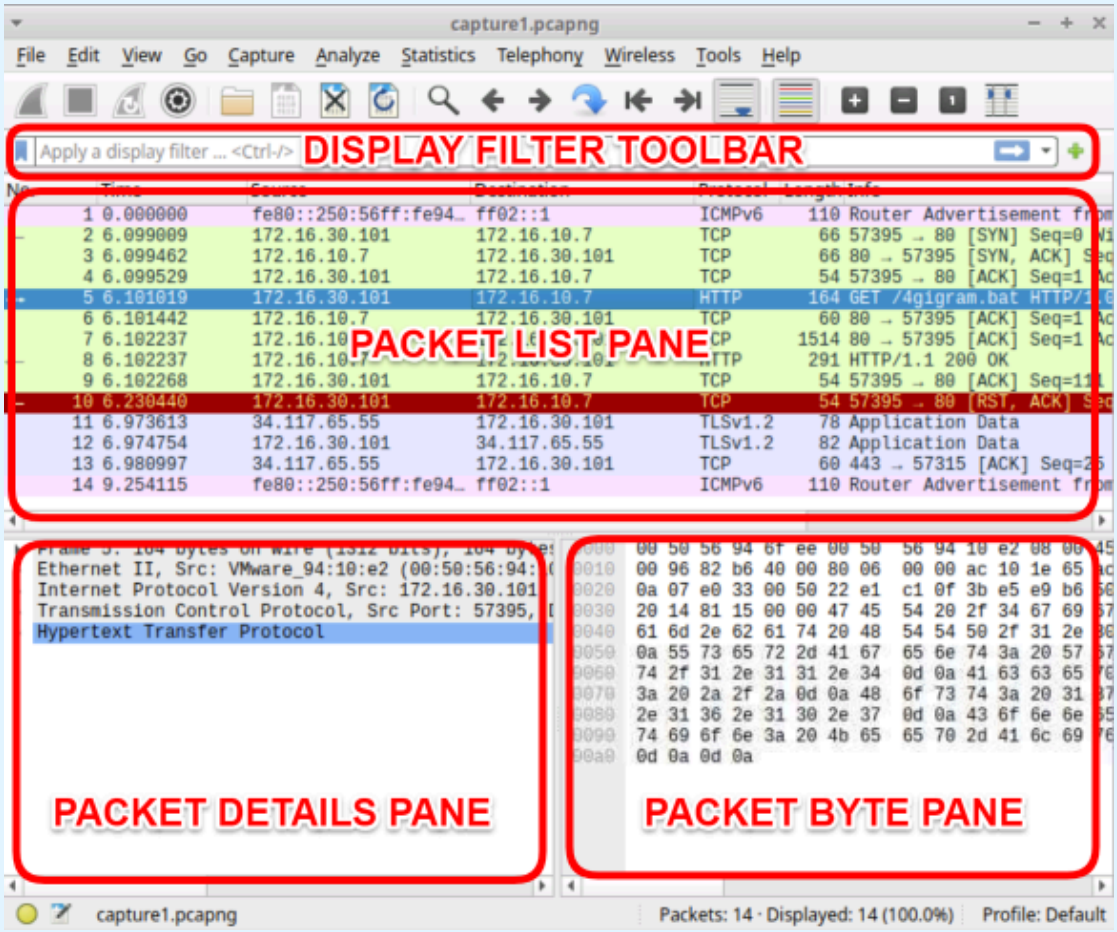
Participant sent:
Wireshark tutorial, please!

Noah sent:
Wireshark is an open-source **packet analyzer**. It is an incredibly versatile and powerful tool that is commonly used for network analysis, debugging, and troubleshooting.

Noah sent:
However, we will use only a small subset of Wireshark's features today. We will only be using it to review previously captured network traffic that has been flagged as suspicious. As such, I'm only going to go over Wireshark's main user interface so you can confidently navigate it for today's tasks.

Noah sent:
Here is a screenshot of Wireshark's main user interface that I've annotated...

Noah sent:



Noah sent:
While I've highlighted and named all the main parts of the user interface, we will only be using the **Display Filter Toolbar** and **Packet List Pane** today.

Noah sent:
The **Packet List Pane** is a scrollable, sortable, and color-coded pane of all the packets in a packet capture. The **Display Filter Toolbar** allows us to filter what packets in the packet capture are displayed in the Packet List Pane.

Participant sent:
This looks complicated...

Noah sent:
Well, I won't lie to you; packet analyzers and networking can be complicated.

Noah sent:
But for what we are doing today, we will be barely scratching the surface of all the information you see in that screenshot. So, if it looks like a bunch of alien text to you, that's fine! Trust me, that is how everyone feels when they are just starting.

Noah sent:
Today, we will only be looking at small packet captures and filtering for the type of network traffic that tends to be easier to quickly figure out, FTP traffic, or more plainly, the traffic associated with the good ol' file transfer protocol.

Noah sent:
And don't worry! I'll walk you through what I want you to do today step-by-step and give you all the details you'll need once we get into the tasks.

Participant sent:
Thanks! I'm ready to get started!

Noah sent:
Alright, to give you some background on what you'll be helping me with today, a client sent over two **packet captures** from one of their FTP servers that contain what they believe to be evidence of **brute force attacks**.

Noah sent:
Our job is to **analyze the traffic** in the packet captures and confirm if either or both contain evidence of a brute force attack, and if so, we will provide the **IP addresses of any potential threats in our report** to the client.

Noah sent:
Analyzing network traffic, characterizing suspicious activity, and identifying threats are some of the core duties of a Cyber Defense Analyst.

Noah sent:
Today, I'm going to show you how to review these two simple packet captures in **Wireshark** and identify if a brute force attack was likely occurring based on the traffic in the capture.

Noah sent:
Regardless of whether a packet capture shows evidence of a brute force attack, I want you to record report entries for each of the two packet captures.

Participant sent:
Sounds good.

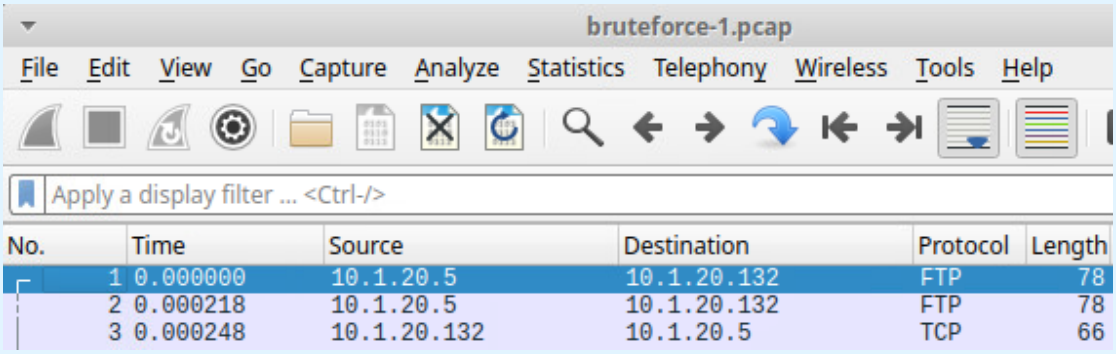
Noah sent:
Alright, let's get into the packet captures then.

Noah sent:
You can find the **two packet captures** in the `Materials` **folder** on the Desktop of this Ubuntu Linux workstation. The packet capture files are named `bruteforce-1.pcap` and `bruteforce-2.pcap`.

Noah sent:
To open the first packet capture, `bruteforce-1.pcap`, open the `Materials` folder and **double-click** on `bruteforce-1.pcap`. That will automatically open the packet capture in Wireshark.

Noah sent:
It should look something like this...

Noah sent:



Noah sent:
Let me know when you've got it open.

Participant sent:
I've got it open!

Noah sent:
Okay, let me briefly explain what you are seeing in the packet list pane (the colorful pane in the center).

Noah sent:
There are 120 packets in this packet capture. Each packet is listed in a separate row in the packet list pane. If you scroll to the bottom and look at the packet capture `Time`, you will notice that the packets were captured within roughly 5 seconds.

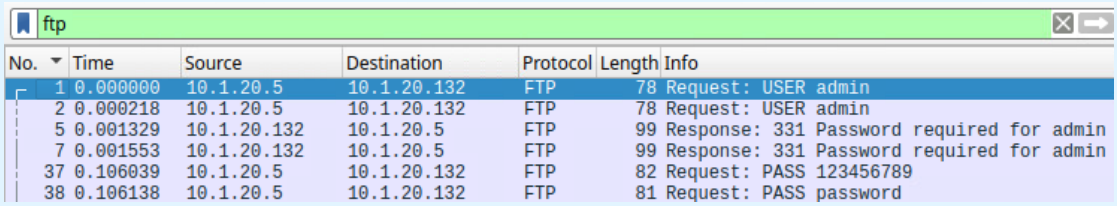
Noah sent:
This capture mostly contains a burst of **TCP/FTP exchanges** between a client (at the IP address `10.1.20.5`) and our client's FTP server (at the IP address `10.1.20.132`). The parts of the capture that we are interested in are color-coded blue (rows 1-8, row 10, row 12, and so on in the packet list pane).

Noah sent:
To filter this down and make the interesting part easier to see, let's apply a display filter via the display filter toolbar, the text box near the top of the application.

Noah sent:
Let's apply a display filter limiting it to just `ftp` packets. To do that, select the display filter toolbar, type in `ftp`, and hit `Enter` or `Return` on your keyboard.

Noah sent:
Now you should have something that looks like this...

Noah sent:



Participant sent:
I see the filtered results.

Noah sent:
Okay, now that we've removed the noise, we can clearly see malicious activity here.

Noah sent:
Look at the `Info` section for the top eight packets in our filtered capture (i.e., packets 1, 2, 5, 7, 37, 38, 43, and 46). What you are seeing here is the start of a textbook **dictionary attack**, a specific kind of brute force attack, on our client's FTP server.

Noah sent:
Essentially, those eight packets are showing that in the span of less than 2 seconds, someone tried to log in to the FTP server twice via the username `admin` with passwords that are obviously from common password *wordlists*.

Noah sent:
If you keep scrolling down, you'll see that this pattern continues. The same client (at the IP address `10.1.20.5`) tries to log in as `admin` with very different and obviously bad passwords (e.g., password, 123456789, 111111, etc.).

Noah sent:
As a note, this is not someone typing super fast. This is from a brute-forcing tool/software of some kind. The attacker was hoping our client's FTP server had a weak password on a commonly used highly privileged account.

Noah sent:
You'd be surprised how often these kinds of attacks actually work. 😟

Noah sent:
Alright, let's add our first entry to our report for the `bruteforce-1.pcap` packet capture. We will need to confirm the client's suspicion about the brute force attack and include the IP address performing the brute force attack.

Noah sent:
To **add an entry** to our report, you must use the **Report Form**. (Note: You can find the Report Form on the Report Tab)

Noah sent:
We will be doing **one report entry per packet capture**. So, for the **first entry**, you need to input the following via the Report Form...

Which packet capture is this report entry for? **bruteforce-1.pcap**

Is there a brute force attack in this packet capture? **Yes**

If so, what IP address performed the brute force attack? `10.1.20.5`

Noah sent:
Let me know when you've successfully added the entry to the report with the Report Form.

Participant sent:
Done. What's next!

Noah sent:
Okay, we only have one more **packet capture to review**, `bruteforce-2.pcap`, for **evidence of a brute force attack** against our client's FTP server (at the IP address `10.1.20.132`).

Noah sent:
Hmm... scratch that, you only have one more packet capture to review. Looks like I have to meet with another client urgently.

Noah sent:
But don't worry, I've skimmed the `bruteforce-2.pcap` packet capture. It is a fairly simple packet capture, just like `bruteforce-1.pcap`. I'm certain you will be able to identify if there are the trappings of a brute force attack in the second capture.

Noah sent:
All you need to do is repeat what we did with the first packet capture. Open `bruteforce-2.pcap` in Wireshark and apply the `ftp` display filter so you can more easily review the contents.

Noah sent:
Once you've reviewed the second capture, be sure to add the last entry to the report with the **Report Form** just like you did for the first entry.

Noah sent:
Thanks for your help today. Good Luck!

TRY CYBER