25 March 2025

Spring 2025

(INFT-3508 - 20332)

Cyber Security Fundamentals

Homework 1

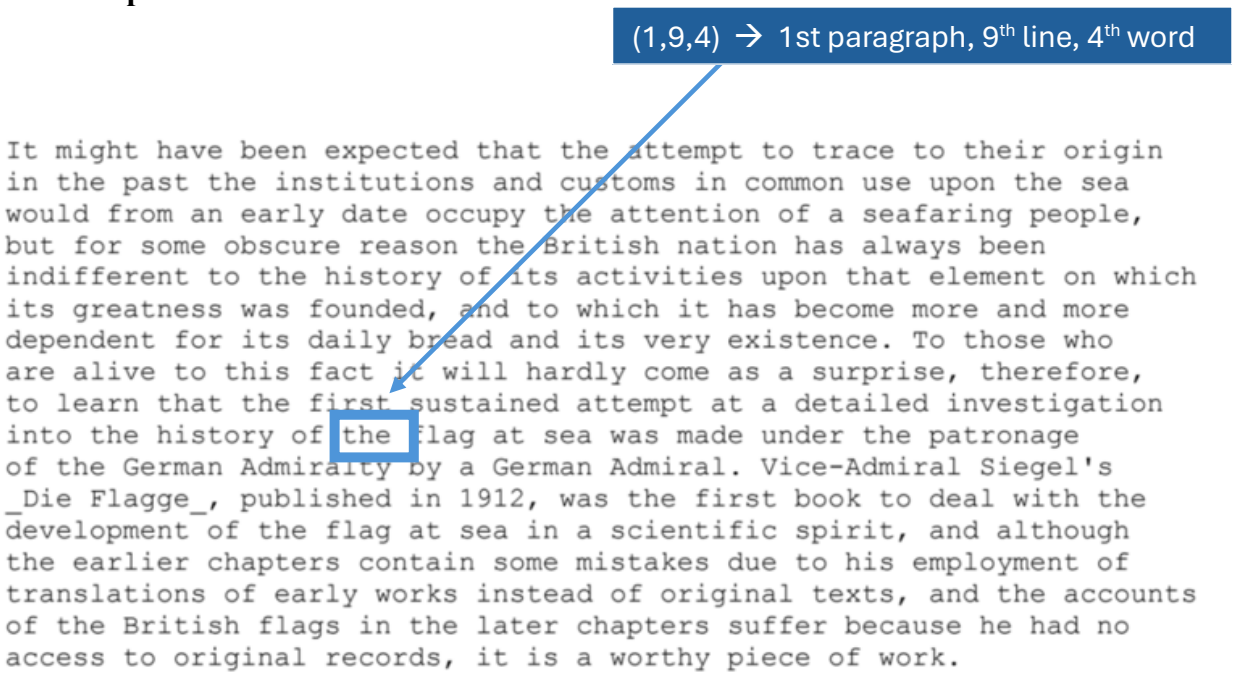
Task 1

First, I observed that the text in “book.pdf” is divided into paragraphs. Since the highest first digit in the number sets is 8 (as seen in (8,10,1)), there are at least 8 paragraphs in the document. This means the first number in each triplet likely refers to a specific paragraph. For example, in (1,9,4), the “1” points to the first paragraph.

Next, the second number in each triplet probably indicates the line within the paragraph. Paragraphs are made up of lines, and these line numbers seem to fit logically. For instance, in (1,9,4), the “9” refers to the ninth line of the first paragraph.

Finally, the third number in the triplet specifies the position of a word within that line. For example, in (1,9,4), the “4” means the fourth word in the ninth line of the first paragraph. Similarly, (7,1,5) would point to the fifth word in the first line of the seventh paragraph.

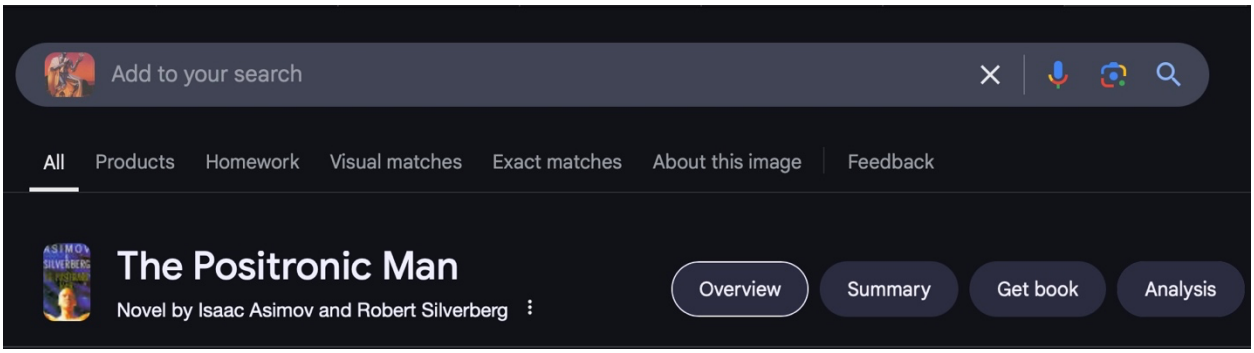
By following this pattern – (paragraph number, line number, word position), we can extract a specific hidden message from the “book.pdf” document. This message is **“The flag is Ceremonial plates”.**



Task 2

To solve this puzzle, we need to identify the name of the book depicted in the poster. Firstly, I downloaded the poster and uploaded the image to a search bar on Google. This technique is called Reverse Image Search. It works by analyzing the visual patterns in an uploaded image, comparing it against millions of images online, and identifying matching or similar images along with their associated web pages.

After this step, I found that this poster is from a science fiction novel about robots and humanity. The name of the book is “The Positronic Man” by Isaac Asimov and Robert Silverberg.

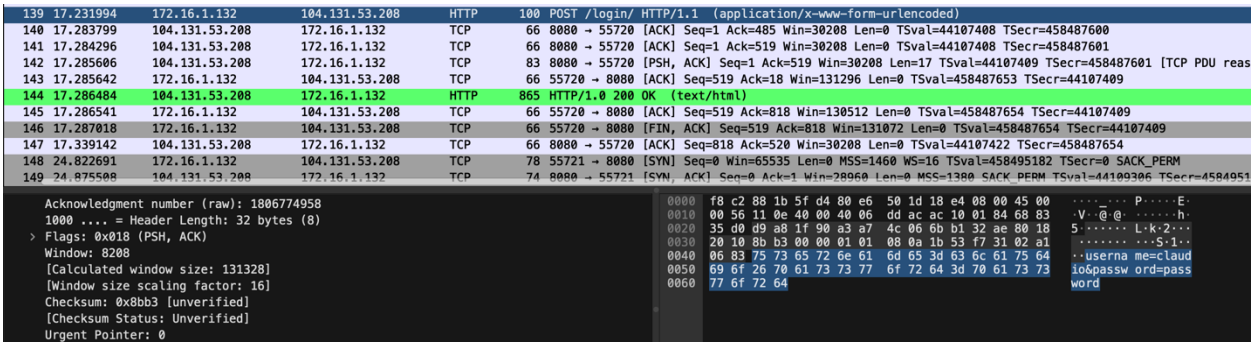


Task 3

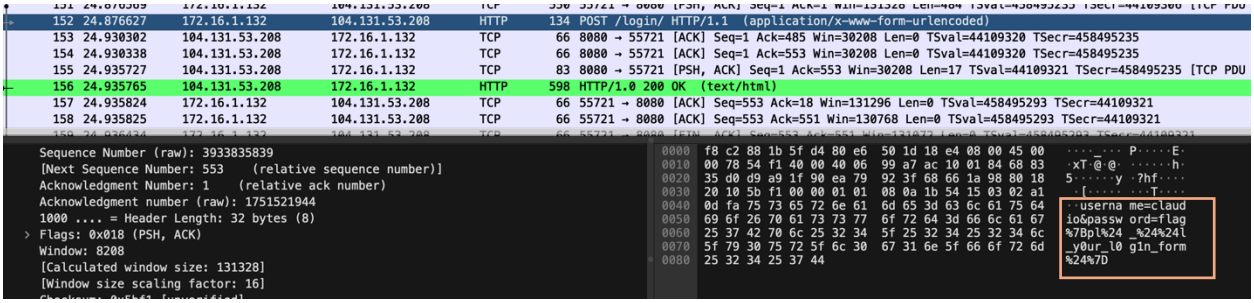
The captured network traffic reveals a sequence of communications between the local device (IP: 172.16.1.132) and different remote servers. Initially, a TCP handshake takes place, which is a standard three-step process for establishing a reliable connection. The client first sends a SYN packet, the server responds with a SYN-ACK, and the client completes the process with an ACK. This confirms that both devices are ready to exchange data.

After establishing the connection, multiple HTTP GET and POST requests follow. The GET requests indicate that the client is fetching resources from a website, likely for loading scripts and other webpage components. An essential detail in this traffic capture is an HTTP POST request that contains sensitive information. Since HTTP does not encrypt data, credentials are visible in plain text.

First Attempt



Second Attempt



These pictures represent an HTTP POST request to "/login," indicating that a user is attempting to log into a website or system by sending their username and password through this request to the "/login" endpoint. The content type is "application/x-www-form-urlencoded," which signifies that it transmits form data, including a username and password.

Here, we can see two login attempts. The first attempt is in packet 139, where the password is "password," which is incorrect. After this failed attempt, there is a second login attempt in packet 152. When we click on the packet, we can clearly see both the username and the correct password in the right-bottom section. (highlighted with orange color)

- Username: claudio
- Password: flag%7Bpl%24_%24%24l_y0ur_l0g1n_form%24%7D

If we decode the existing password in hexadecimal values, we get the password: **redflag{p1$_$$l_y0ur_l0g1n_form$}**

Task 4

In this task, we are trying to decode an encrypted text. The structure of the text looks like a song, but at the top, it seems like there is a separate sentence. This sentence is likely important; therefore, I decided to focus on decoding it first.

First, I looked for short, frequently repeated words in the text. I found “wcn,” which appears many times throughout the text and starts the first sentence and song. Since “the” is a very common word (article) in English and often appears at the beginning of sentences, I guessed that “wcn” might mean “the.” In this case “w” is “t”, “c” is “h”, and “n” is “e”. To know that this way is true I started testing on different words and started decoding.

Next, I searched for other short words, such as prepositions and articles, because they often follow predictable patterns. By identifying these words, I could decode more letters. Each time I decoded a new letter, I wrote it down, allowing me to apply it to other words and gradually reveal them. This step-by-step process made it easier to decode the rest of the text.

1. aw (3rd line 3rd word) - _ t → can be “it”
2. at (28th line) – i _ → can be “if”
3. ht (14th line 7 th word) - _f → can be “of”
4. enn (4th line 5th word) - _ee → can be “see”
5. cze (11th line 1st word) - _ as → can be “has”
6. oatn (11th line 3rd word) - _ ife → can be “life”
7. oayn (13th line 8th word) – li _ e → can be “like”
8. wcady (1st line 2nd word) – thi _ k → can be “think”
9. ewaoo (4th line 2nd word) – sti _ _ → “still”
10. zoo (18th line 5th word) – a _ _ → can be “all”
11. jawc (18th line 4th word) - _ ith → can be “with”
12. zdg (2nd line 1st word) – an _ → can be “and”
13. thvnew - (21st line 9th word) – fo _est → can be “forest”
14. ihohve - (19th line 7th word) - _ olors → can be “colors”
15. izd (9th line 9th word) - _ an → can be “can”
16. dzrn - (11th line 9th word) – na _ e → can be “name”
17. rhkdwzade (18th line 10th word) – mo _ ntains → can be “mountains”

As I noted the letters, it looked like this:

w-t
c-h
n-e
z (is used very often and separately) - a
a-i
t-f
h-o
e-s
o-L
y-k
j-w
d-n
g-d
v-r
r-m
i-c
k-u

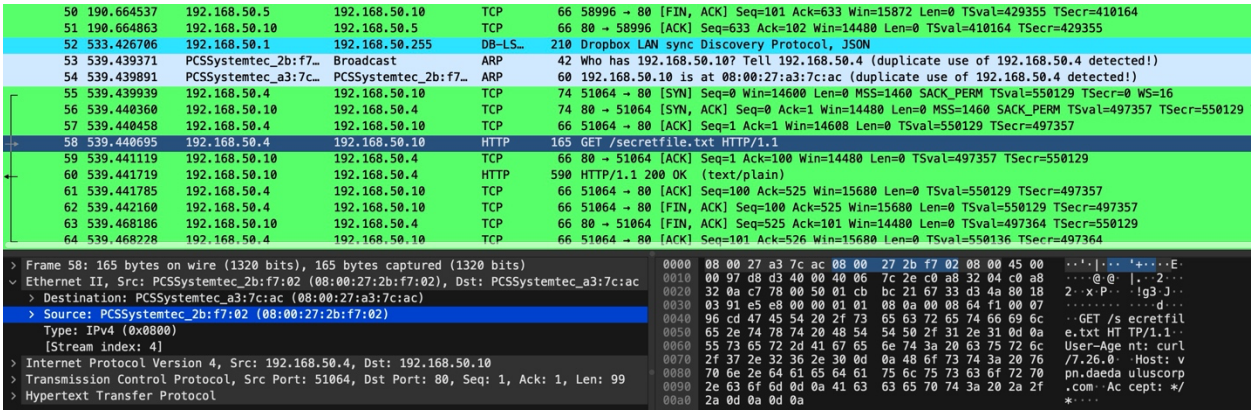
With these letters identified, I could finally decode the first sentence "**wcn zkwchvapzwahd ihgn ae jawczoowcnihohvehtwcnjadg**"

- wcn = "the"
- zkwchvapzwahd = "authorization" (Since "p" was not yet decoded, I identified the missing letter logically as "z")
- ihgn = "code"
- ae = "is"
- jawczoowcnihohvehtwcnjadg = "withallthecolorsofthewind"

By putting all the decoded words together, I found the full sentence: "**The authorization code is withallthecolorsofthewind**." From this sentence, it is immediately clear that the actual authorization code is "**withallthecolorsofthewind.**"

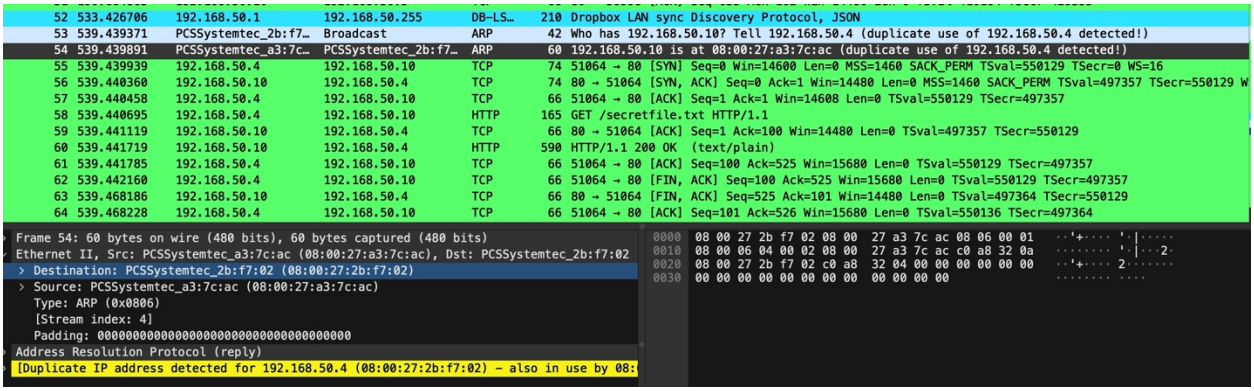
Task 5

In order to determine the attacker's last name, we need to carefully analyze packets. One key observation is in packet 58, where we see an HTTP GET request for a file named "secretfile.txt." The source IP address is 198.168.50.4, and the destination is 192.168.50.10. However, there is a suspicious activity. Before the broadcast, in packet 51 we can see that the expected destination is 192.168.50.5, not 192.168.50.4.



To uncover the actual IP address used by the attacker, we need to examine the last ARP request. The reason is that when an IP conflict or unauthorized device appears on the network, ARP requests help to detect the attack. By checking the last ARP request, we find a message "Duplicate use of 192.168.50.4 detected!". This confirms that an attacker is masquerading as 192.168.50.4.

Additionally, we identify that this attacker has a MAC address of **08:00:27:2b:f7:02**. This is a key piece of information because MAC addresses are unique in each device and cannot be easily changed like IP addresses.



Now, we need to analyze the user activity to find the attacker. Observing the packets, we can see that IP 192.168.50.3 requested /john.johnson endpoint. How do we know this is suspicious? The source MAC address (08:00:27:2b:f7:02) of this request matches the attacker's MAC address. This means that the attacker is pretending to be (masquerading) to be a real user.

Therefore, the answer is: "**johnson, 192.168.50.3, 192.168.50.4**"