Jacky Guzman Nunez
Worked with Anne Lehr

Part 1 :

a. Yes, there is a cookie for this domain. The name is “theme” and the value is “default.”
b. Yes! Did change from having a value of "default" to "red."
c. We saw the HTTP headers "Cookie: theme=default" and "Set-Cookie: theme=default; Expires=Fri, 24 Jan 2025 16:26:09 GMT; Path=/" . We do see the same cookie values as we did in the Inspector because both represent the same cookie that sets the theme.
d. Yes! Our red theme is still selected.
e. Through the cookie??????????????????????????????????????
f. The browser lets the server know about the change by sending the Cookie header along with the Get /fdf/ HTTP/1.1 request. This lets the server know the current state of the website's theme and allows for it to make any appropriate changes (even when there are no changes, the cookie is always set to a specific value).
g. You simply type the name of the theme you want (default, red, blue) in the "Value" section, press Enter, and refresh the website so it displays the new change (we didn't have to use the drop-down menu). We are basically modifying the cookie's information.
h. We have to intercept the request before it gets sent out if we want to be able to modify the theme in Burpsuit's Proxy tool.
i. In a macOS, cookies are stored in the Library folder, and inside that folder are different folders that belong to different browsers, where cookies are stored (found the information in: http://allaboutcookies.org/what-is-a-cookie-file).

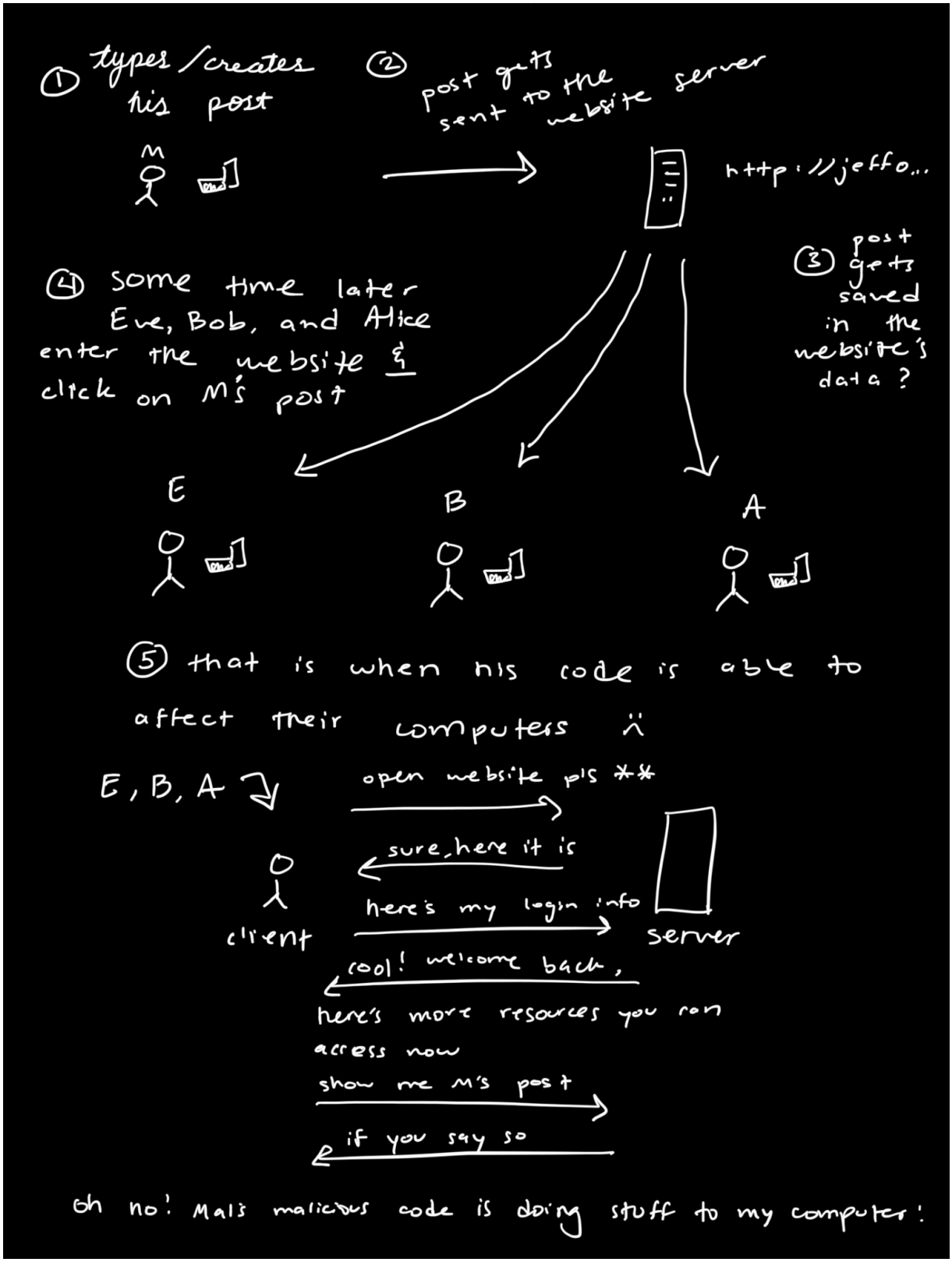
Part 2 :

- I logged in as Eve.
- I see the red text and the "alert" pop-up message.
- How I believe Moriarty made the red text and pop-up message appear:
    - used simple Javascript code to alter the color of the text (<script> allows for Javascript to be integrated into the HTML code)
- Experimenting with HTML, CSS, Javascript
    - post [JGN] Taxes?
        - redirect the user to youtube, where they get rick rolled :)

a.

- Moriarty types his evil posts and submits them to the FDF
- Once submitted, his posts "attack" anyone who clicks on them/opens them.
    - By opening his posts, the unsuspecting victims unintentionally allow for the code in Moriarty's post to run, causing "chaos". (At this point we need to find Sherlock so he can fix the mess.)

* this can happen "anytime and anywhere", as there is no specific time the code runs (like 2pm every day)



b. Javascript might have access to Alice's cookies (I read about this in a website but am confused as to how that works). It also can redirect Alice to another website, possibly even a copy of a website, where Alice would be asked to sing in again because of a 'small error in the wifi or internet connection', allowing for Mal to get Alice's password and username easily.
c. Javascript could also be used to make a paywall appear! I got this idea from a student who made a post about paywalls, which I think is brilliant! If Mal can mae a paywall appear, then they could steal a poor innocent user's credit/debit card information.
d. - Technique a browser can use to prevent Moriarty from doing any more crazy things: read input from the user as plain text, not code, so that the input (whether its code or not) will have no impact whatsoever on the browser when it has to display the input
   - Technique a server can use to prevent Moriarty from doing any more crazy things: validate and clean up user input (checking for malicious code) to prevent a reflected attack

Note: (Who knew cookies and discussion posts were dangerous???)