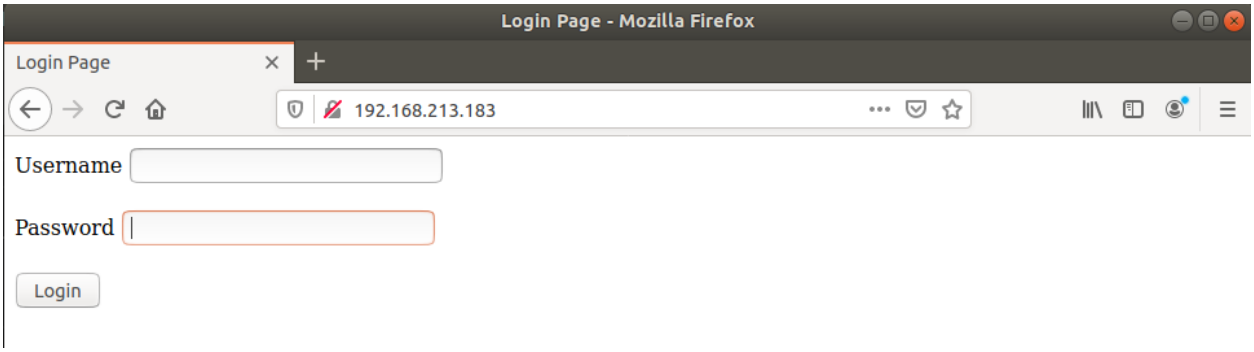
Exploiting CSRF Against my PHP Scripts

Setting up my Environment

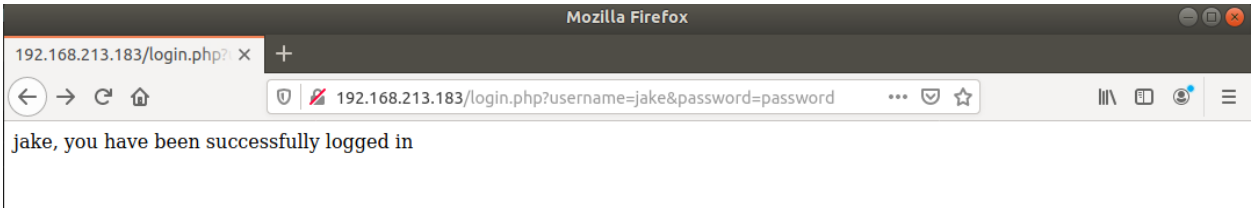
1. Install apache2 and libapache2-mod-php using the install.sh script
2. Move the php and html files given to the /var/www/html directory
3. Create two additional files in /var/www/html: username.txt and password.txt
    a. Set the permissions of both of these files to -rw-rw-rw-
4. Enter a username in username.txt and a password in password.txt
    a. This will act as the only user for my application

How my Scripts Work

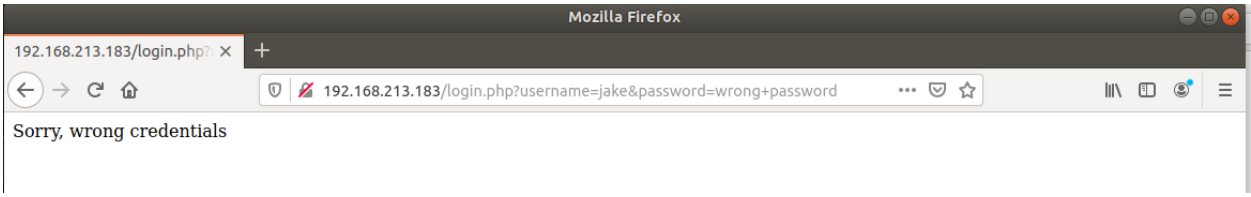
1. Navigate to my website at http://192.168.213.183. You will be met with this screen:



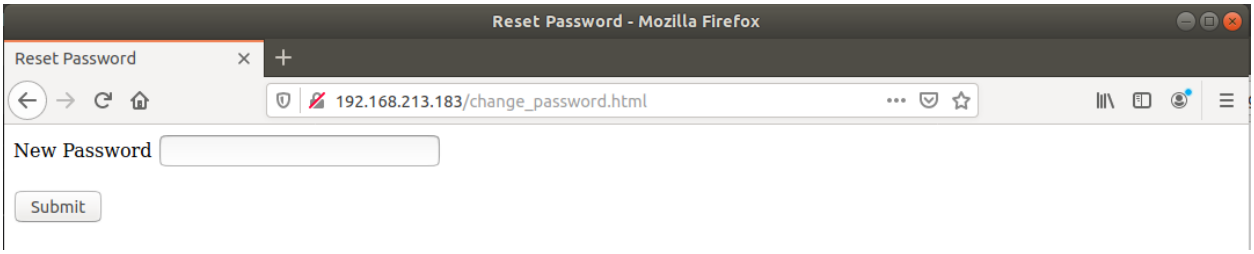
    a. For this step and the rest of the steps below, I will be using the IP address of my testing server which is 192.168.213.183. If your testing environment differs, please substitute accordingly.
2. Log in with the credentials in the username.txt and password.txt files. After you log in, a message will be returned telling you if your log in was successful or not. Both are pictured below.
    a. Successful login



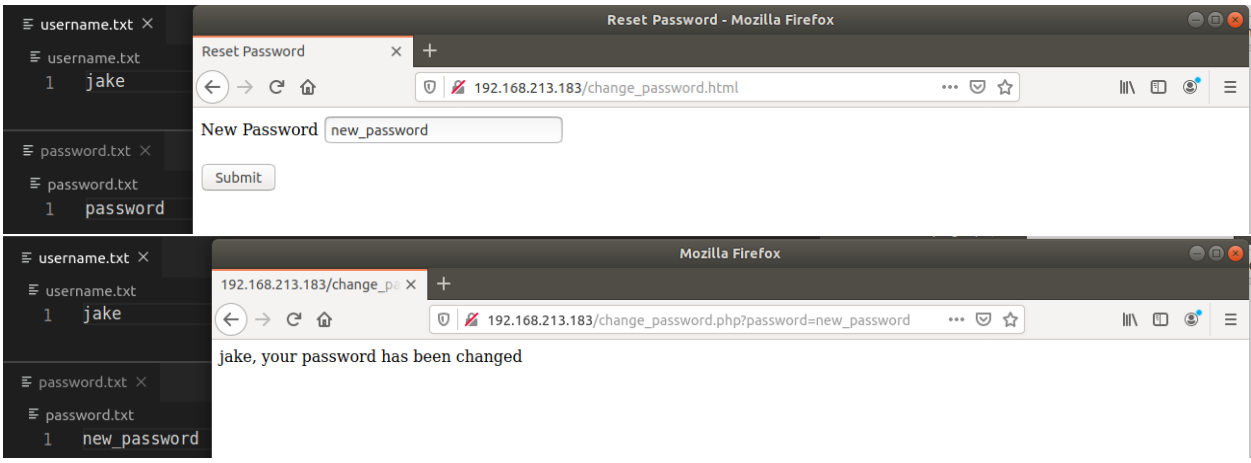
    b. Unsuccessful login



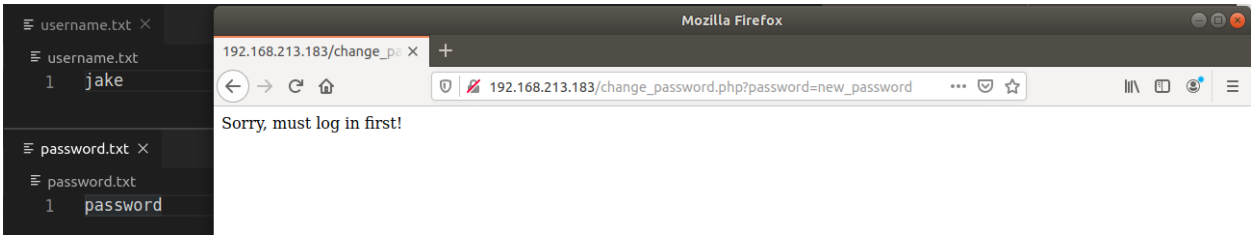
3. To change your password, navigate to http://192.168.213.183/change_password.html. You will be met with this screen.



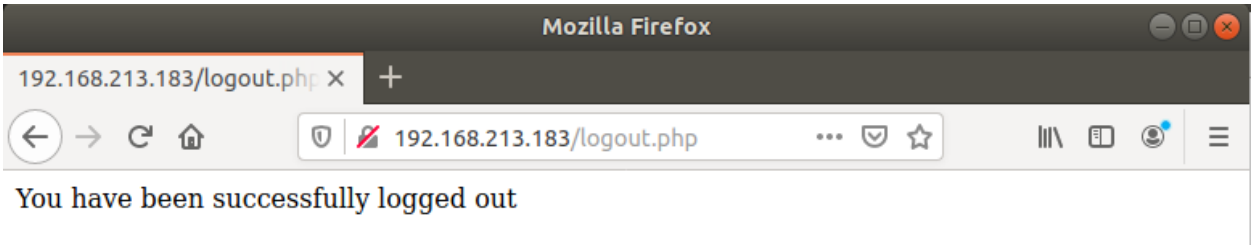
4. If you are logged in, you may enter a password and it will change in password.txt. If you are not logged in, the password will not be changed. Both scenarios are pictured below.
    a. Logged in



    b. Not logged in



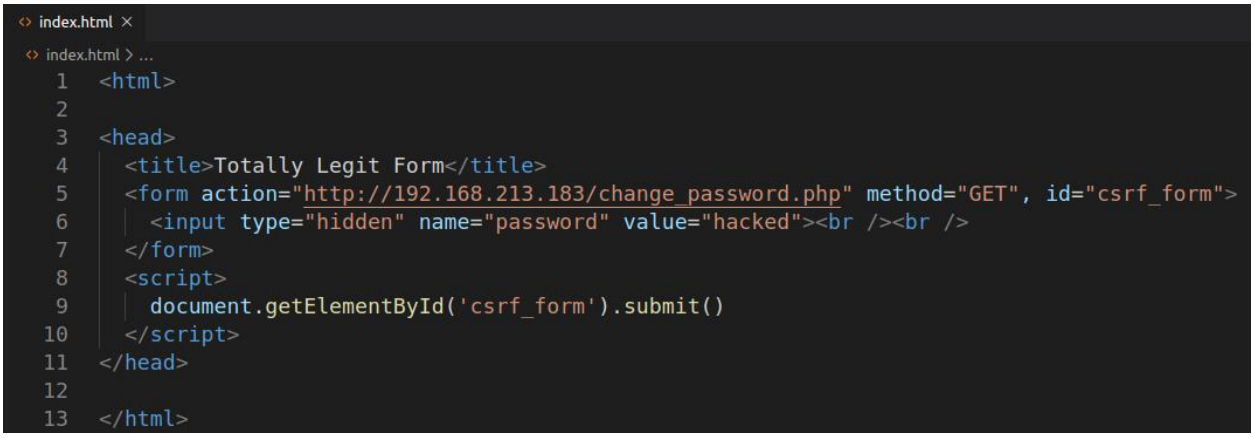
5. Log out of your account by navigating to http://192.168.213.183/logout.php. Below is what you will see when logging out.



How to Exploit my PHP Scripts

1. Log in to my application using step 1 and 2 above to mimic a user logging in.

2. Start an apache server on a remote host using the install.sh script and paste the following code in /var/www/html/index.html. This code will be attached with my PHP code as malicious_index.html for easy pasting.



3. On the host housing the PHP application, navigate to the malicious page to mimic a user navigating to a malicious website. Below is a screenshot from my testing environment.