Introduction

For this assignment, you’ll be exploiting a number of web vulnerabilities. The teaching staff has installed a separate instance of the “Damn Vulnerable Web App (DVWA)” for each student in the class.

Setting up SSH tunnels

As the name implies, this app is highly vulnerable to manipulation, and is massively insecure. We don’t want to make it publicly available, since that will clearly lead to an owned computer. To access the webserver, you will need to create an ssh tunnel\(^1\). On Linux or Mac, you can do this via:

```
ssh netid@your_alice_machine_ip -NfxC -L9999:209.148.46.205:80
```


The above (either Mac/Linux or Windows) means that if you go to port 9999 on your local machine, you’ll be connected to port 80 (the web port) of my server (whose IP address is 209.148.46.205) through your netid-alice-HW1 machine. I’ll go over this in class, but knowing how this works is not critical to solving this homework.

For this homework assignment, you will be doing all of the work using your local web browser that runs on your computer.

---

\(^1\) For more information about ssh tunnels, [read this.](http://howto.ccs.neu.edu/howto/windows/ssh-port-tunneling-with-putty/)
Accessing DVWA

Once the tunnel is set up, you can then access your instance of DVWA by opening up http://localhost:9999/netid/setup.php in your browser, replacing netid with your netid.

This works because, as discussed above, traffic sent to port 9999 on your local machine is tunneled inside of SSH traffic to the webserver 209.148.46.205 on port 80.

Once you’ve connected to DVWA for the first time, you’ll need to reset your local database.

**WARNING:** Repeating this procedure wipes out all state saved to DVWA, so be careful if you do this more than once. Your work will be lost. To do this, click on the “Create/Reset Database” button.

If DVWA asks for a username and password, enter “admin” as the username and “password” as the password.

1. **Reading a secret file {15 points}**

Your job is to read the contents of /var/secret-files/important-secret.txt that is located on the web server. To do this, you’ll need to exploit a vulnerability in the “Command Execution” page of DVWA.

In other words, using only your web browser, you’ll need to exploit the vulnerability to cause DVWA to read this file for you and report its contents back through the web page.

For this part of the writeup, include the exact string that you entered in the “Command Execution” page to exploit the vulnerability. Also, your writeup should specify the contents of /var/secret-files/important-secret.txt.

**HINT:** The “View Source” button on each page is your friend. You don’t need to speak PHP to understand the gist of what it’s trying to do.

**WARNING:** You do not have permission to do anything other than read the contents of /var/secret-files/important-secret.txt. Do not exceed what’s allowed in this homework assignment. Keep in mind that the server is a shared resource, shared amongst all students in this class.

2. **Annoying users {15 points}**

Next, you will annoy visitors to the website with an annoying popup. Go to the “XSS stored” page and add a comment that causes a popup window that reads: “Hi!”

Here’s a big hint: Google for the `alert()` JavaScript function.
No need to turn in anything for this question. Just leave the popup in place and we’ll check that it works.

3  **Resetting a password {20 points}**

Finally, you will conduct a cross-site request forgery (CSRF) attack to reset the currently logged in user’s password to turing. You will do this by exploiting both the XSS vulnerability on the “XSS stored” page and the password reset function that’s on the “CSRF” page.

This attack is definitely trickier than the others. Since the web form imposes a maximum length of 50 (see the `textarea` HTML tag in the XSS page’s html source), you’ll need to override that limit by using Google Chrome’s Developer Tools, or some other plugin (for the particular browser that you are using) that lets you manually override the HTML that’s received.

You will also want to become familiar with the `document.location` JavaScript command.

In the writeup, please include the exact string that you entered in the XSS page to enable the CSRF attack.

**Submission Instructions**

To upload the various parts of your assignment, navigate to the COSC235 course on Blackboard, click the “Assignments” link on the left hand side, and select the “hw5” link. Include either a single PDF document or a single ASCII document (no RTF or Word files please!) that answers the above questions.

Please post questions (especially requests for clarification) about this homework to Piazza.