Alec Howard

Portswigger CSRF Lab 1 Writeup

This lab poses a few challenges to those using Burp Suite community edition as opposed to the professional version. The first step is to open the lab in Burp Suite and to intercept the "Update email" request in the proxy tab. If one was using Burp Suite Pro, they could simply use a tool to generate a CSRF PoC. However, those using community edition do not have that luxury. I was able to find a html template to use for this lab on Portswigger:

- ```
  <form method="POST" action="https://YOUR-LAB-ID.web-security-academy.net/my-account/change-email">
  <input type="hidden" name="email" value="anything%40web-security-academy.net">
  </form>
  <script>
      document.forms[0].submit();
  </script>
  ```

The first line redirects to the session's change email request. The next line auto-fills the malicious email address, and the final line automatically submits the form. The next step is to right click on the intercepted request and copy URL to paste into the placeholder one in the template above. The final step is to paste this html into the body section of the exploit server, store it, and then test it out. This is where I ran into a problem. The malicious email was not overriding the normal one. It took me longer than it should have to troubleshoot, however I finally realized that the given template had an issue at this line:

```
<input type="hidden" name="email" value="anything%40web-security-academy.net">
```

It turns out that the UTF—8 encoding of the '@' sign was throwing an error somewhere and so the email was not actually being replaced. I simply replaced it with the actual character and the lab was solved:

```
<input type="hidden" name="email" value="anything@web-security-academy.net">
```
```