Cross Site Request Forgery
IN618 Security

Introduction

Cross site request forgery (XSRF) is an exploit that is somewhat related to XSS, but it's different enough to warrant exploration on it's own. Basically, XSS vulnerabilities arise when a user's browser trusts the response from a server and gets exploited. In XSRF, the server trusts the request from the browser.

1 Examine the exploit

A simple application at `http://xss.foo.org.nz/lab5.1` is vulnerable to XSRF exploits. Try out the site. Registered students can log in using their surnames as user names and first names as passwords. Anyone else can log in with user name `user` and password `user`. See if you can deduce how it can be exploited.

After you try exploiting it on your own, make sure that you have a saved message. Keep a tab to the message page open while visiting `http://sec-student.foo.org.nz/~tclark` in a second tab. Now go back and reload `http://xss.foo.org.nz/lab5.1/home.php` in your other tab. What happened? Can you see how it happened?

2 Explore the exploit further

If you haven't already done so, create an exploit that deletes the message that a user has previously saved.

3 The problem

We can exploit this vulernability because anytime we can make a logged-in user's browser send a correctly formed request to `http://xss.foo.org.nz/lab5.1/home.php`, the server carries out whatever action the request specifies. It doesn't matter that the request did not come from a user using the form on the web site. How can we make sure that the server will only carry out requests submitted by users using the correct form?