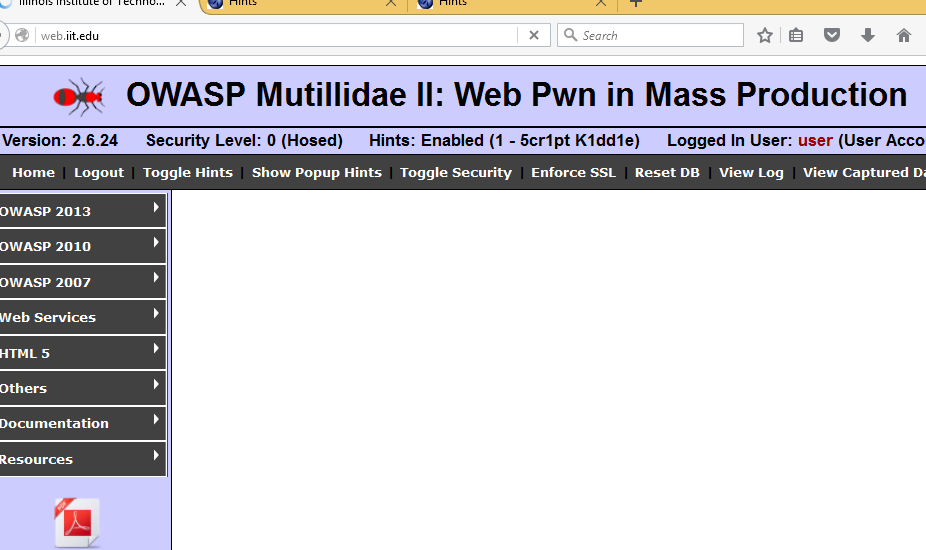
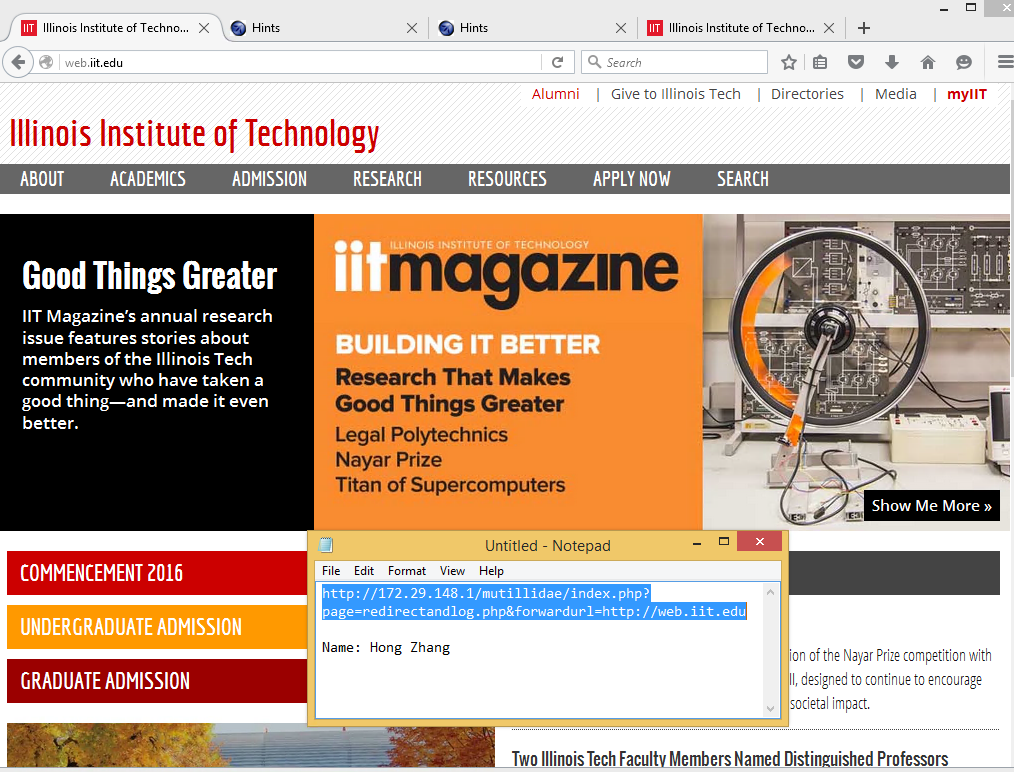
**Exercise 1: “OWASP 2013”/ “A10- Unvalidated Redirects and Forwards”/ “Credits”**

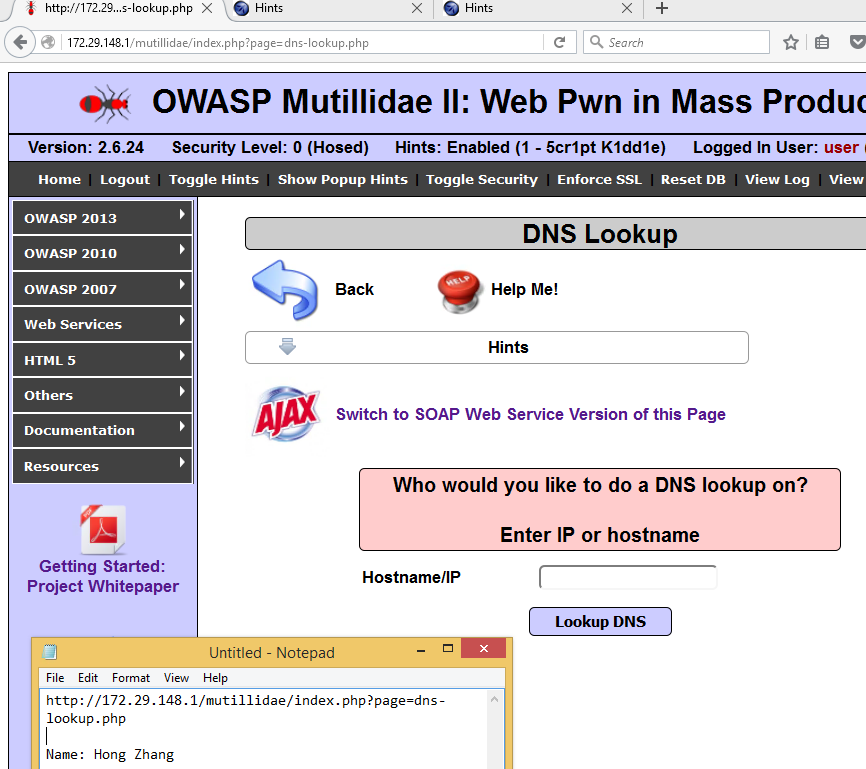
In this exercise, this website has four unvalidated redirects. Because the URL is passed as a parameter, I can replace it. For example, the first web link use <http://172.29.148.1/mutillidae/index.php?page=redirectandlog.php&forwardurl=http://www.owasp.org> to redirect users to the website, <http://www.owa>. Hence, I will change it to <http://172.29.148.1/mutillidae/index.php?page=redirectandlog.php&forwardurl=web.iit.edu> and the website will go to web.iit.edu.

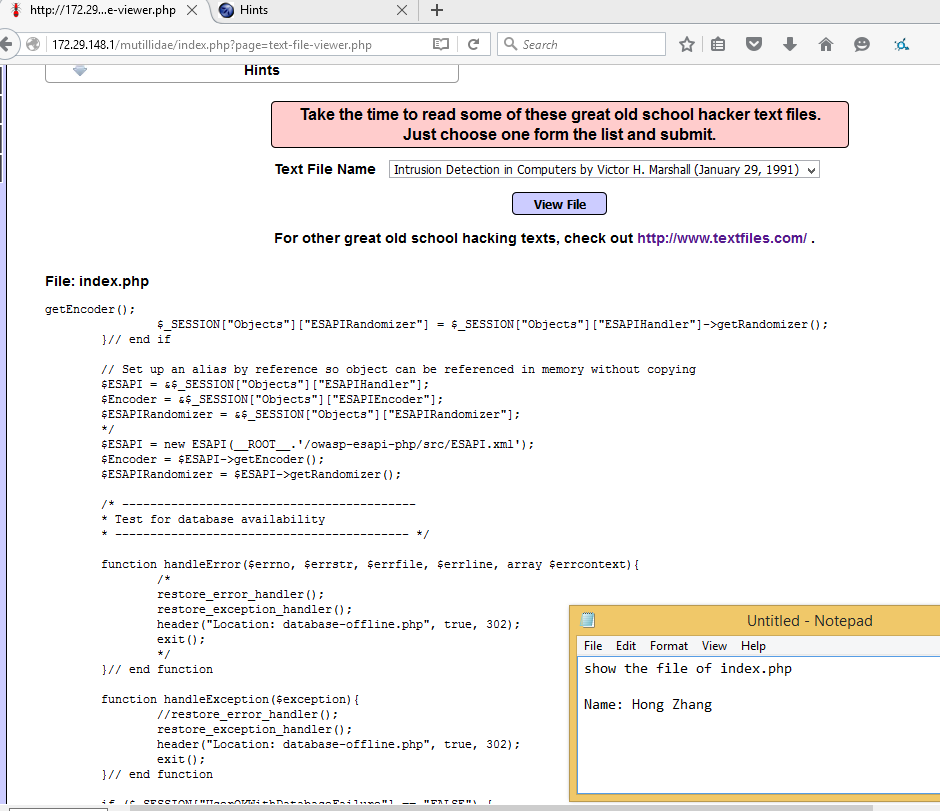
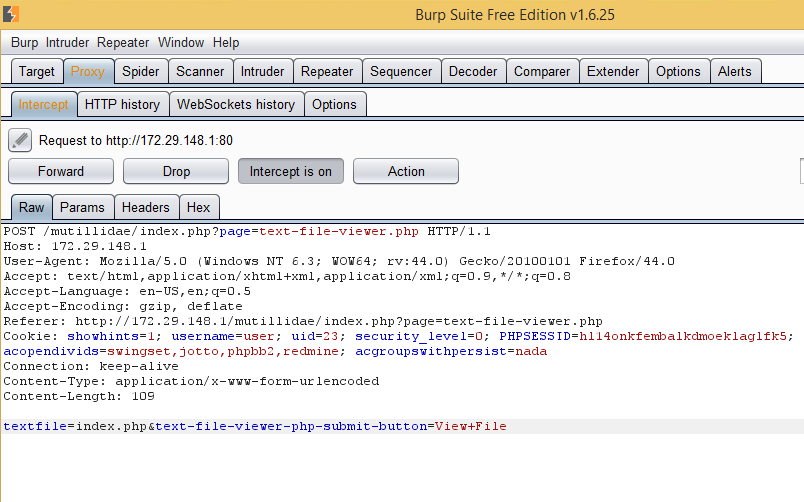




**Exercise 2: “OWASP 2013”/ “A4-Insecure Direct Object References”/ “Text File Viewer”**

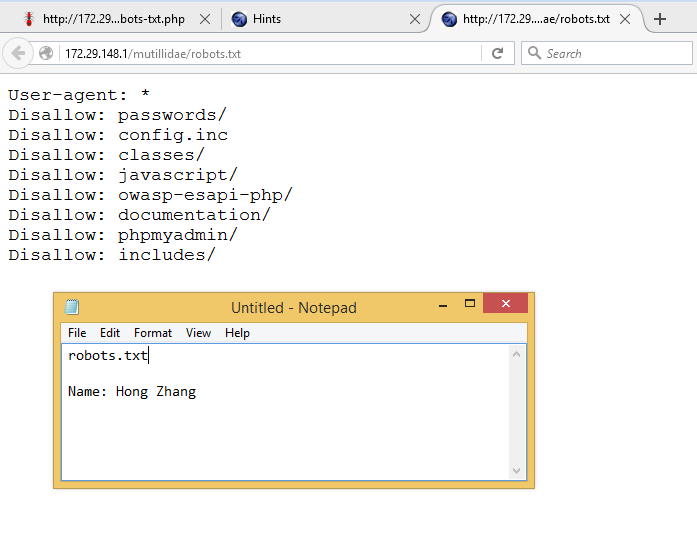
In this exercise, the webpage exposes a reference to an internal implementation object, such as a file. Hence, I can change it and get another one. For example, this webpage is <http://172.29.148.1/mutillidae/index.php?page=text-file-viewer.php>. I can change it to <http://172.29.148.1/mutillidae/index.php?page=dns-lookup.php>.



Another test is using Burp-Suite to get the content of index.php in this webpage.

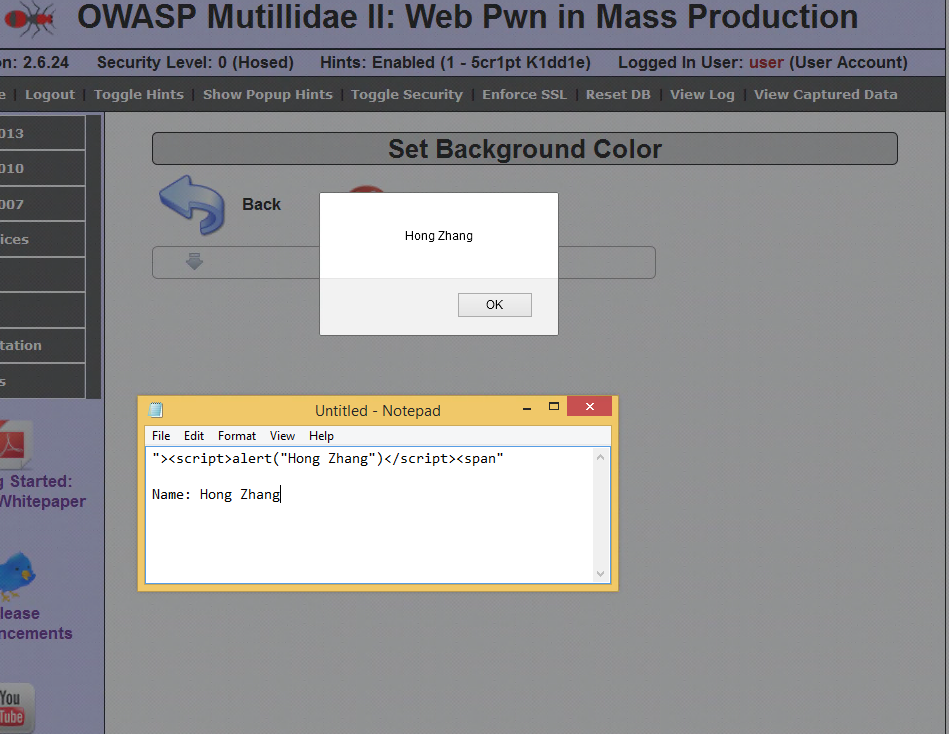
**Exercise 3: “OWASP 2013”/ “A7- Missing Function Level Access Control”/ “Robots.txt”**

The Robots.txt file is web accessible file and used by Google search engine. It should be verified before users see it. However, in this exercise, I use <http://172.29.148.1/mutillidae/robots.txt> and can see it without authorization or request.

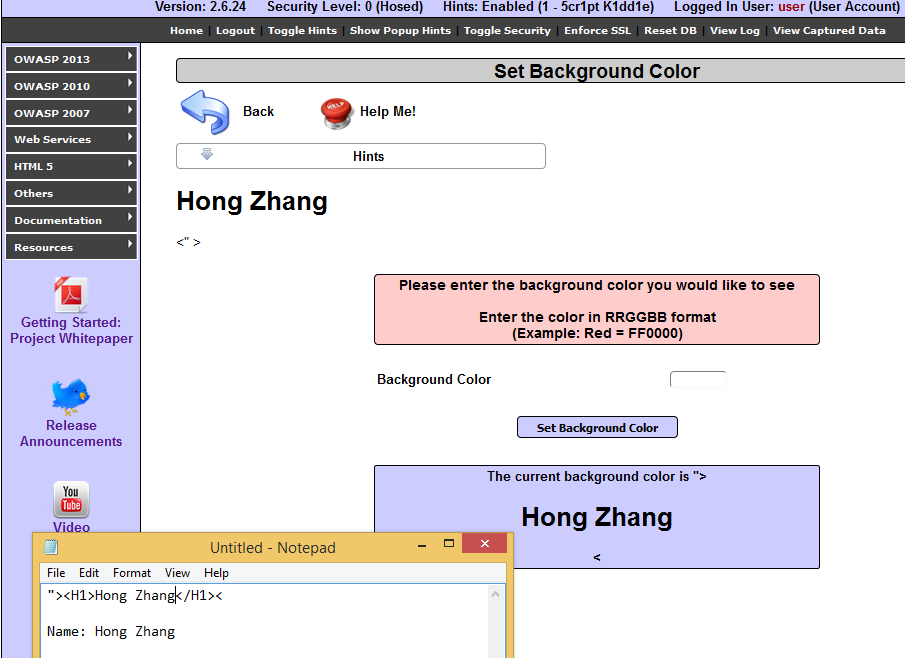


**Exercise 4: “OWASP 2013”/ “A1 – Injection (Other)”/ “HTML Injection (HTML)” / “Set Background Color”**

This exercise is a cascading style injection. I can send untrusted data as part of a command or query.

In the first test, I input a alter box, “><script>alert(“Hong Zhang”)</script><span”.

In the second test, I input some html codes, “><H1>Hong Zhang</H1><” and the interpreter could not configure out.



**Exercise 5: “OWASP 2013”/ “A1 – Injection (Other)”/ “XML Injection” / “XML Validator”**

In this exercise, XML input in this webpage contains a reference to an external entity and I can use injection in it. For example, I can disclose the file of /etc/passwd.

