ETHICAL HACKING V2
LAB SERIES

Lab 11: Client Side Exploitations

Document Version: 2020-08-24

Material in this Lab Aligns to the Following	
Books/Certifications	**Chapters/Modules/Objectives**
All-In-One CEH Chapters ISBN-13: 978-1260454550	6: Web-Based Hacking: Servers and Applications
EC-Council CEH v10 Domain Modules	13: Hacking Webservers 14: Hacking Web Applications 15: SQL Injection
CompTIA Pentest+ Objectives	2.4: Explain the process of leveraging information to prepare for exploitation 3.2: Given a scenario, exploit network-based vulnerabilities 3.4: Given a scenario, exploit application-based vulnerabilities 4.2: Compare and contrast various use cases of tools 4.3: Given a scenario, analyze tool output or data related to a penetration test
CompTIA All-In-One PenTest+ Chapters ISBN-13: 978-1260135947	5: Mobile Device and Application Testing 6: Social Engineering 7: Network-Based Attacks

Contents

Introduction

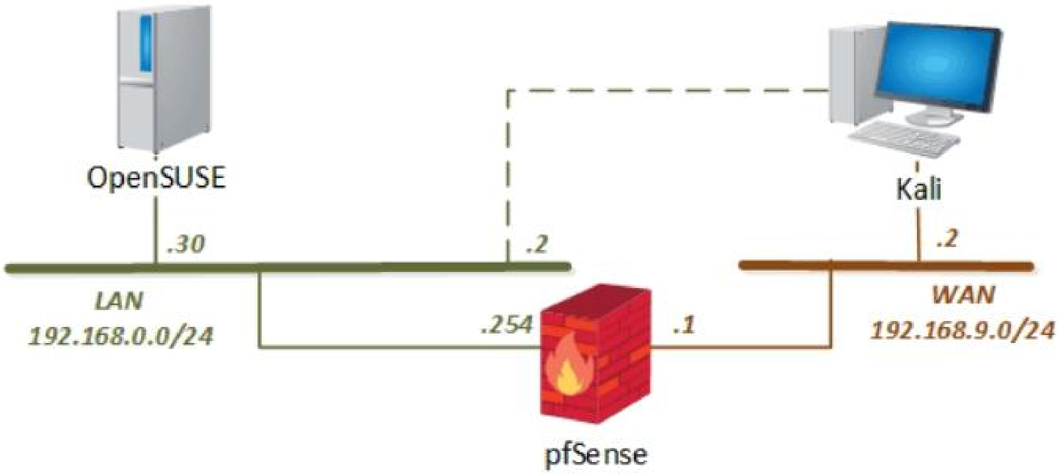
Browsers are susceptible to exploitation and can be used to gain access to the computer system and network. In this lab, we will use the *BeEF framework* to specifically target the browser and exploit the browser.

Objective

In this lab, you will be conducting ethical hacking practices using various tools. You will be performing the following tasks:

1. Hooking Browsers with BeEF Framework
2. Client Exploitation with BeEF Framework

Pod Topology



Lab Settings

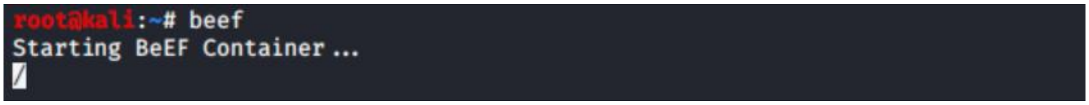
The information in the table below will be needed in order to complete the lab. The task sections below provide details on the use of this information.

Virtual Machine	**IP Address**	**Account** (if needed)	**Password** (if needed)
Kali Linux	192.168.9.2 192.168.0.2	root	toor
pfSense	192.168.0.254 192.168.68.254 192.168.9.1	admin	pfsense
OpenSUSE	192.168.0.30	osboxes	osboxes.org

1 Hooking Browsers with BeEF Framework

1. Click on the **Kali** tab.
2. Click within the console window and press **Enter** to display the login prompt.
3. Enter **`root`** as the *username*. Press **Tab**.
4. Enter **`toor`** as the *password*. Click **Log In**.
5. Open a new terminal by clicking on the **Terminal** icon located at the top of the page if the terminal is not already opened.
6. Enter the following command to start BeEF:

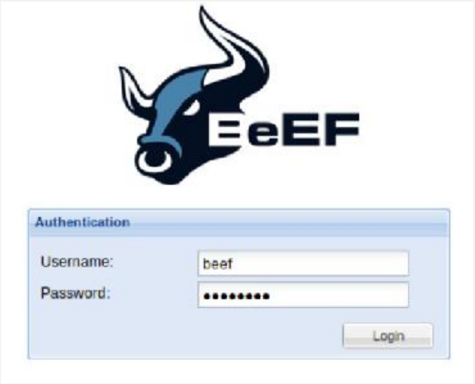
```
beef
```



```
root@kali:~# beef
Starting BeEF Container ...
```

Wait about 1-2 minutes until a web browser appears with a *BeEF* login page. This environment runs BeEF in a docker container.

7. Log in with **`beef`** as the *username* and **`password`** as the *password*. Click **Login**.

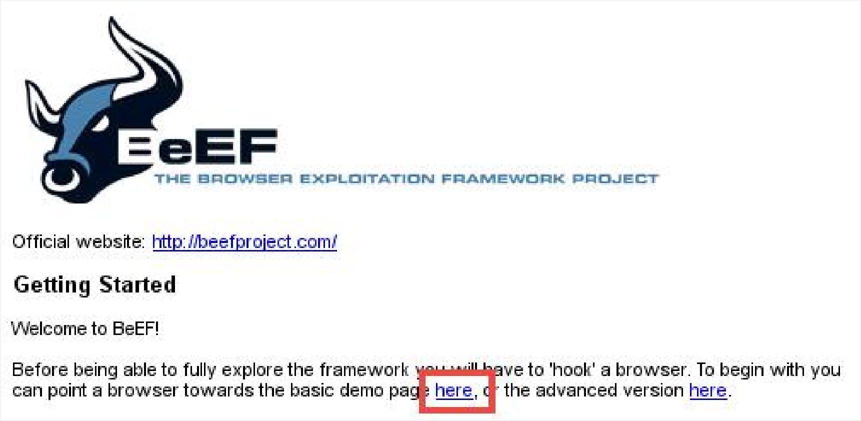


Authentication

Username: beef

Password: ••••••••

8. Working with *BeEF*, a victim is needed to hook their browser. In the middle pane, there are two demo links, click on the **here** hyperlink to navigate to the *BeEF Basic Demo* page.



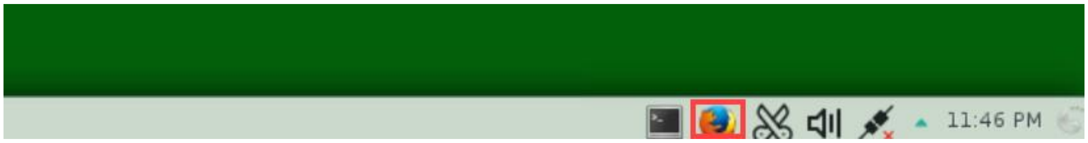
Official website: http://beefproject.com/

Getting Started

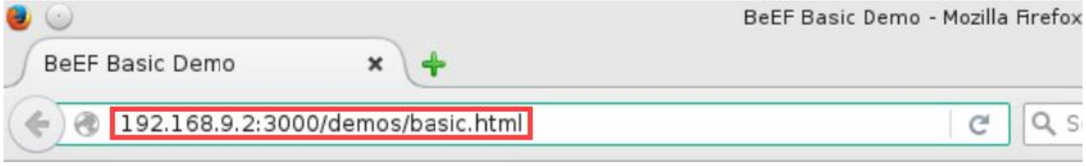
Welcome to BeEF!

Before being able to fully explore the framework you will have to 'hook' a browser. To begin with you can point a browser towards the basic demo page here, or the advanced version here.

9. Leave the *BeEF Demo* page open and click on the **OpenSUSE** tab.
10. Log in with `osboxes` as the *username* and `osboxes.org` as the *password*. Press **Enter**.
11. Once logged in, launch *Firefox* by clicking on the **Firefox** quick launch icon located on the bottom panel.

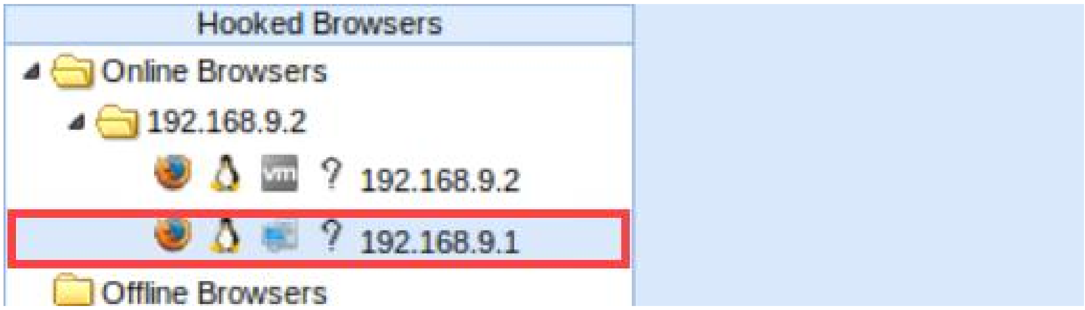


12. When viewing the *Firefox* browser, enter `192.168.9.2:3000/demos/basic.html` into the *address* field. Press **Enter**.

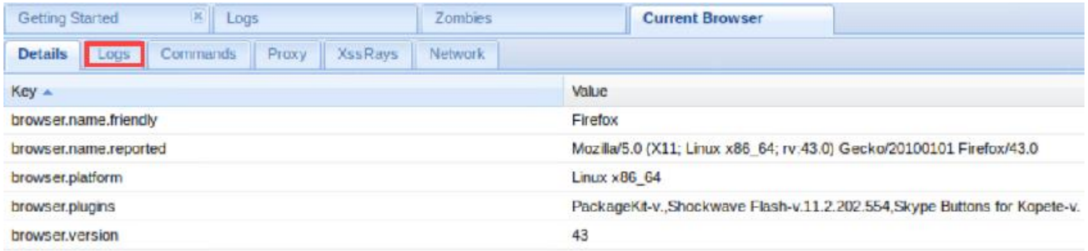


192.168.9.2:3000/demos/basic.html

13. Leave the Firefox browser open and navigate back to the **Kali** VM.
14. Make sure to view the **Mozilla Firefox** browser and click on the **BeEF Control Panel** tab.
15. Notice on the *Hooked Browsers* list towards the left, a new online browser appears. Click on **192.168.9.1**.

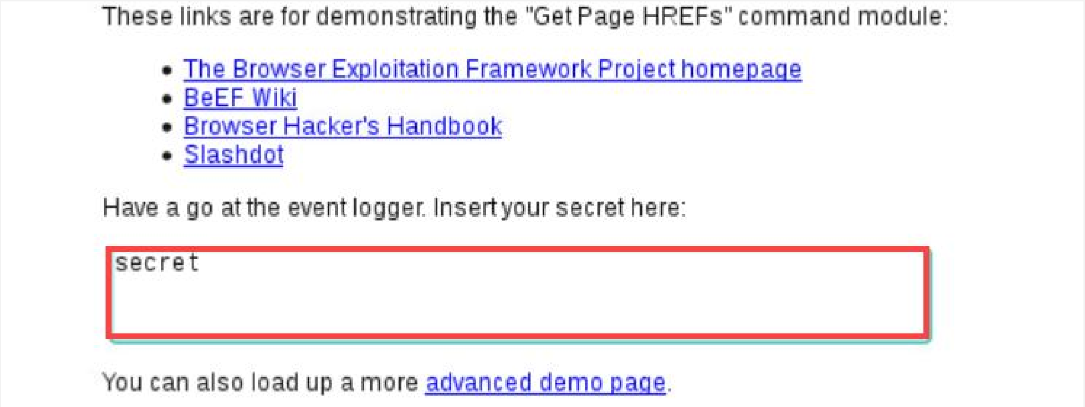


16. Once the hooked browser is selected, notice the given information in the middle pane. It appears that the browser that is hooked is running *Firefox* version 43 based on the *Browser UA String*. Click on the **Logs** tab.



Key ▲	Value
browser.name.friendly	Firefox
browser.name.reported	Mozilla/5.0 (X11; Linux x86_64; rv:43.0) Gecko/20100101 Firefox/43.0
browser.platform	Linux x86_64
browser.plugins	PackageKit-v.,Shockwave Flash-v.11.2.202.554,Skype Buttons for Kopete-v.
browser.version	43

17. Generate some events so they can be analyzed on the *Logs* tab. Navigate back to the **OpenSUSE** tab.
18. While viewing the *Firefox* browser, type **`secret`** into the *Insert your secret here:* text field. Press **Enter**.



These links are for demonstrating the "Get Page HREFs" command module:

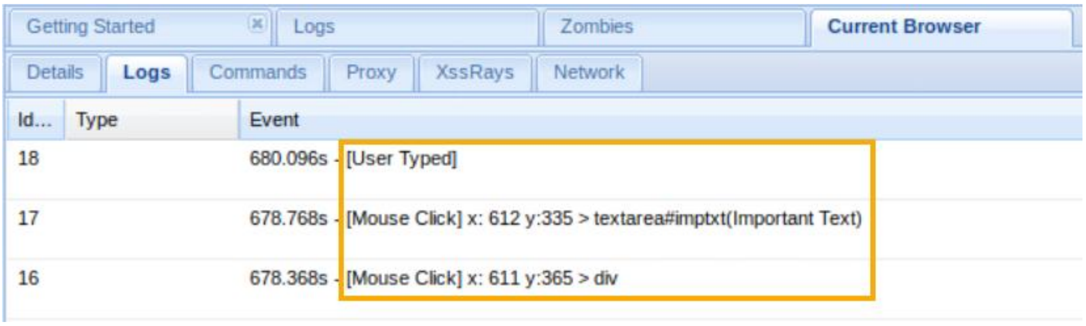
- The Browser Exploitation Framework Project homepage
- BeEF Wiki
- Browser Hacker's Handbook
- Slashdot

Have a go at the event logger. Insert your secret here:

secret

You can also load up a more advanced demo page.

19. Navigate back to the **Kali** VM.
20. While viewing the *BeEF Control Panel* tab, press the **F5** key to refresh the page.
21. Click on **192.168.9.1** from the *Hooked Browsers* pane underneath *Online Browsers*.
22. In the middle pane, click on the **Logs** tab.

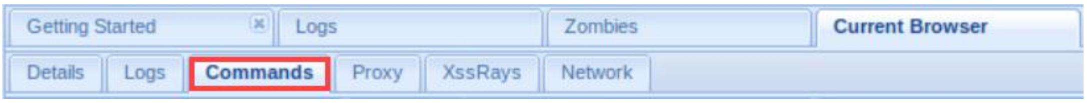


Id...	Type	Event
18		680.096s - [User Typed]
17		678.768s - [Mouse Click] x: 612 y:335 > textarea#imptxt(Important Text)
16		678.368s - [Mouse Click] x: 611 y:365 > div

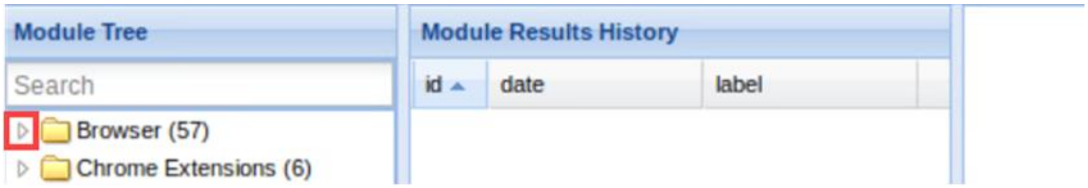
Notice some events are shown here, including captured keystrokes.

2 Client Exploitation with BeEF Framework

1. Click on the **Commands** tab in the middle pane.



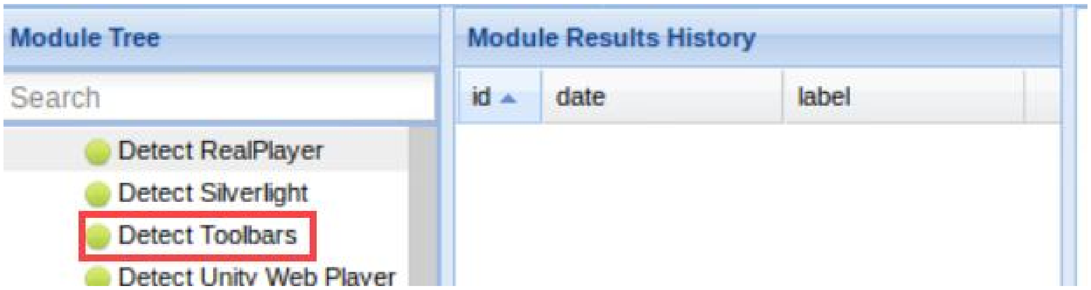
2. In the *Module Tree* pane, expand the **Browser** inventory.



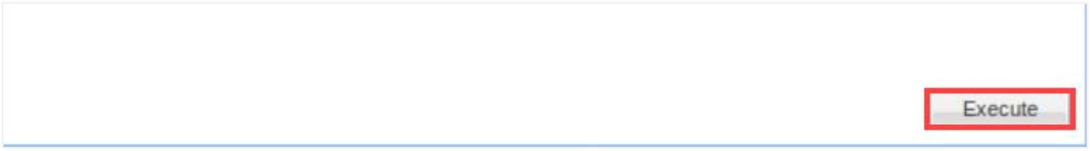
Once expanded, notice the different colors presented.

The color *green* means that those commands can be used.
The color *orange* means that the commands may not work.
The color *red* means that the commands do not work against the current browser.

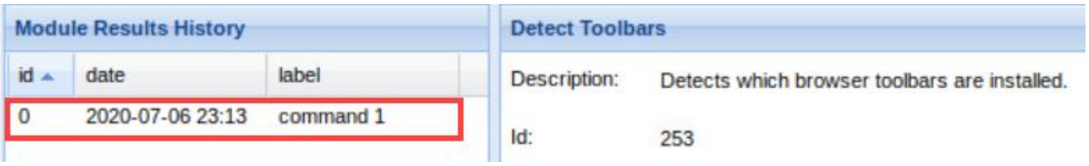
3. From the same pane, click on **Detect Toolbars**.



4. Once selected, click the **Execute** button located on the bottom-right corner.



5. Notice the *Module Results History* pane populates. Click on the **command 1** result.

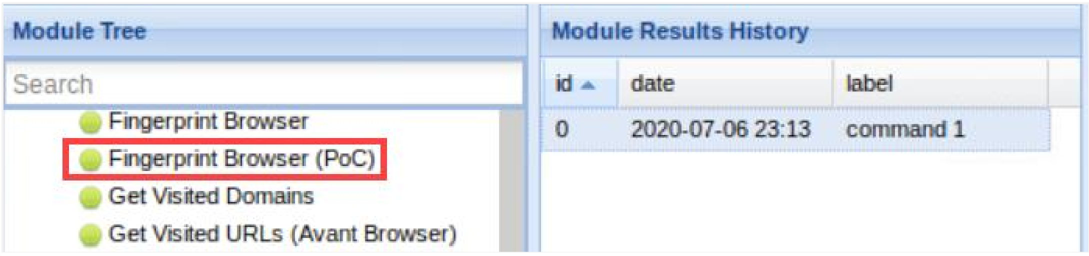


id	date	label
0	2020-07-06 23:13	command 1

Description: Detects which browser toolbars are installed.

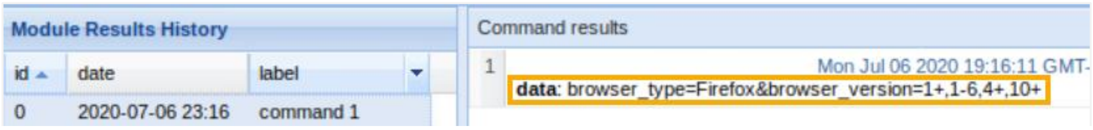
Id: 253

6. Notice in the *Command results* pane that no toolbars have been detected.
7. Focus on the *Module Tree* pane and select **Fingerprint Browser (PoC)**.



id	date	label
0	2020-07-06 23:13	command 1

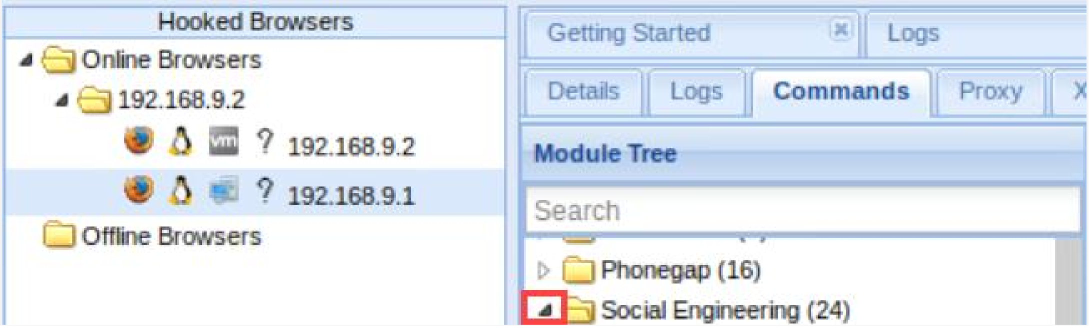
8. Once selected, click the **Execute** button.
9. Notice the *Module Results History* pane populates. Select the **command 1** entry.
10. The given results show that the hooked browser has been successfully fingerprinted.



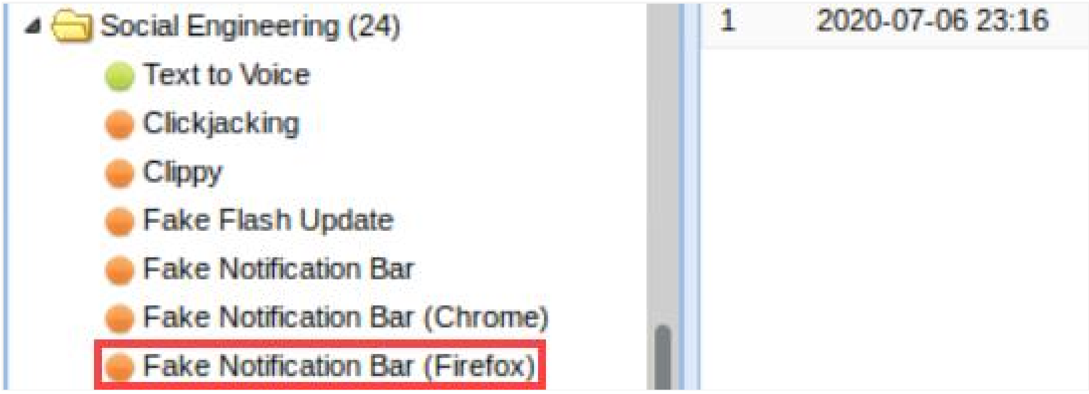
id	date	label
0	2020-07-06 23:16	command 1

```
1                                Mon Jul 06 2020 19:16:11 GMT-
data: browser_type=Firefox&browser_version=1+,1-6,4+,10+
```

11. In the *Module Tree* pane, expand the **Social Engineering** inventory.

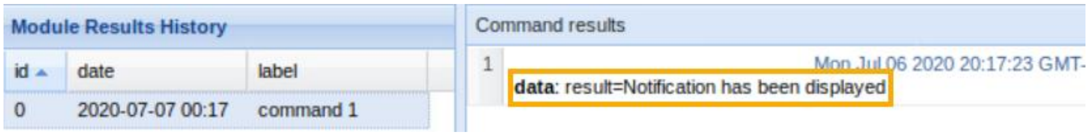


12. Select **Fake Notification Bar (Firefox)** from the same pane.



13. Once selected, click the **Execute** button.
14. The *Module Results History* pane should populate, click the **command 1** entry.

15. Notice the result indicates that a notification has been displayed. Switch to the **OpenSUSE** VM.



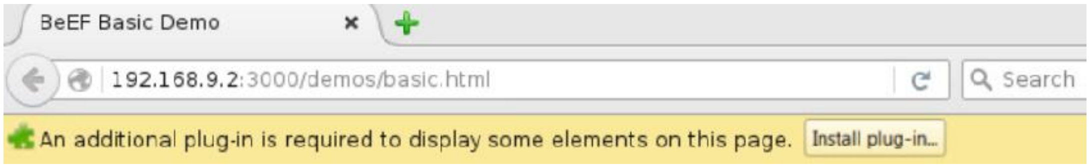
Module Results History

id	date	label
0	2020-07-07 00:17	command 1

Command results

1 Mon Jul 06 2020 20:17:23 GMT-
data: result=Notification has been displayed

16. Focus on the *Firefox* web browser and notice a notification is present, asking to install a plug-in. Don’t install the plug-in.

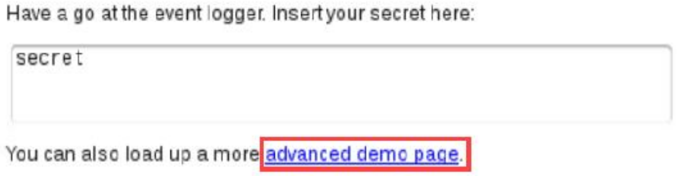


BeEF Basic Demo

192.168.9.2:3000/demos/basic.html

An additional plug-in is required to display some elements on this page.

17. While viewing the *BeEF Basic Demo* tab on the *Firefox* window, click the **advanced demo page** link.

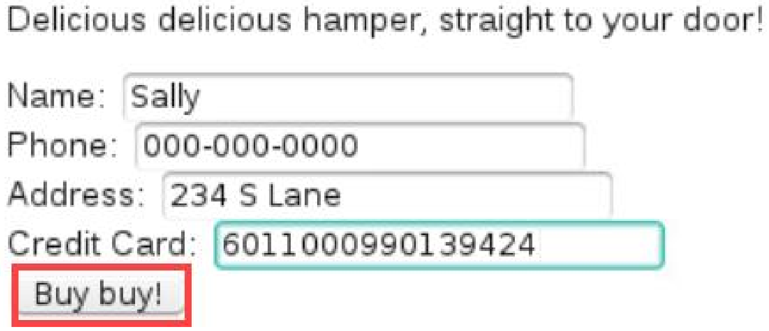


Have a go at the event logger. Insert your secret here:

secret

You can also load up a more advanced demo page.

18. Once the webpage redirects, click the **Order Your BeEF-Hamper** button.
19. Notice that a few text fields appear. Fill in each text field using the information below:
    a. *Name*: **`Sally`**
    b. *Phone*: **`000-000-0000`**
    c. *Address*: **`234 S Lane`**
    d. *Credit Card*: **`6011000990139424`**

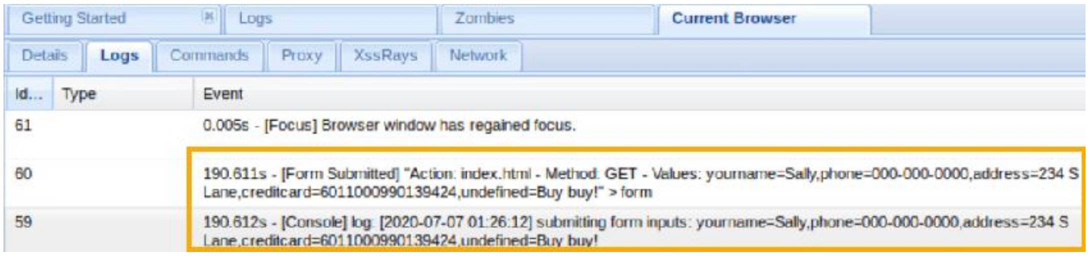


Delicious delicious hamper, straight to your door!

Name: Sally
Phone: 000-000-0000
Address: 234 S Lane
Credit Card: 6011000990139424

20. Click the **Buy buy!** button.
21. Switch to the **Kali** VM.
22. While viewing the *BeEF Control Panel* tab, press **F5** to refresh the page.
23. Click on **192.168.9.1** from the *Hooked Browsers* pane underneath *Online Browsers*.

24. In the bottom-middle pane, click on the **Logs** tab.



Id...	Type	Event
61		0.005s - [Focus] Browser window has regained focus.
60		190.611s - [Form Submitted] "Action: index.html - Method: GET - Values: yourname=Sally,phone=000-000-0000,address=234 S Lane,creditcard=6011000990139424,undefined=Buy buy!" > form
59		190.612s - [Console] log: [2020-07-07 01:26:12] submitting form inputs: yourname=Sally,phone=000-000-0000,address=234 S Lane,creditcard=6011000990139424,undefined=Buy buy!

Notice the captured keystrokes from the hooked browser.

25. You may now end your reservation.