BeEf

Advertencia

El autor de esta demo no se hace responsable por el uso que le den a la herramienta, ya que el propósito de este documento es con fin educativo y todo acto ilegal realizado por cualquier persona no será responsabilidad del autor.

¿Qué es BeEf?

Es una herramienta usada en el mundo de los pentesters, ya que nos permite generar vectores de ataque. Se centra básicamente en encontrar vulnerabilidades del browser, permitiendo así comprometer la seguridad de la victima, nos ofrece una lista extensa de métodos que se pueden aplicar como explotación, tenemos desde ataques de redireccionamiento, hasta activación de la cámara conectada al dispositivo.

Instalación

1. Comenzaremos instalando las dependencias, en este caso ingresaremos el siguiente comando:

```
sudo apt install ruby ruby-dev
```

```
└─$ sudo apt install ruby ruby-dev
[sudo] password for kali:
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
Suggested packages:
  ri
The following packages will be upgraded:
  ruby ruby-dev
2 upgraded, 0 newly installed, 0 to remove and 1134 not upgraded.
Need to get 23.6 kB of archives.
After this operation, 0 B of additional disk space will be used.
Get:1 http://http.kali.org/kali kali-rolling/main amd64 ruby amd64 1:3.0+1kali2 [12.2 kB]
Get:2 http://http.kali.org/kali kali-rolling/main amd64 ruby-dev amd64 1:3.0+1kali2 [11.4 kB]
Fetched 23.6 kB in 1s (32.3 kB/s)
(Reading database ... 311015 files and directories currently installed.)
Preparing to unpack .../ruby_1%3a3.0+1kali2_amd64.deb ...
Unpacking ruby (1:3.0+1kali2) over (1:3.0+1kali1) ...
Preparing to unpack .../ruby-dev_1%3a3.0+1kali2_amd64.deb ...
Unpacking ruby-dev:amd64 (1:3.0+1kali2) over (1:3.0+1kali1) ...
Setting up ruby-dev:amd64 (1:3.0+1kali2) ...
Setting up ruby (1:3.0+1kali2) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for kali-menu (2022.3.1) ...

┌──(kali㉿kali)-[~]
└─$
```

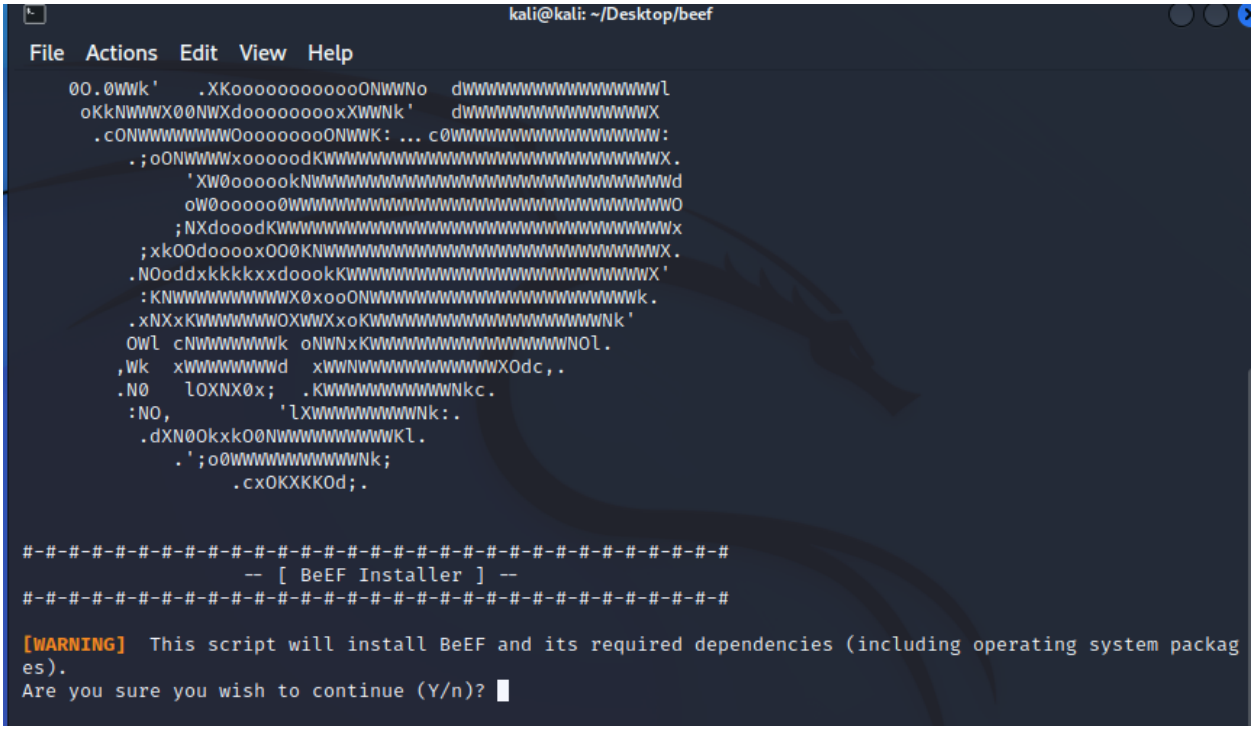
2. A continuación, nos dirigimos al Desktop y clonamos el repositorio con nuestra herramienta.

```
cd Desktop
sudo git clone https://github.com/beefproject/beef.git
```

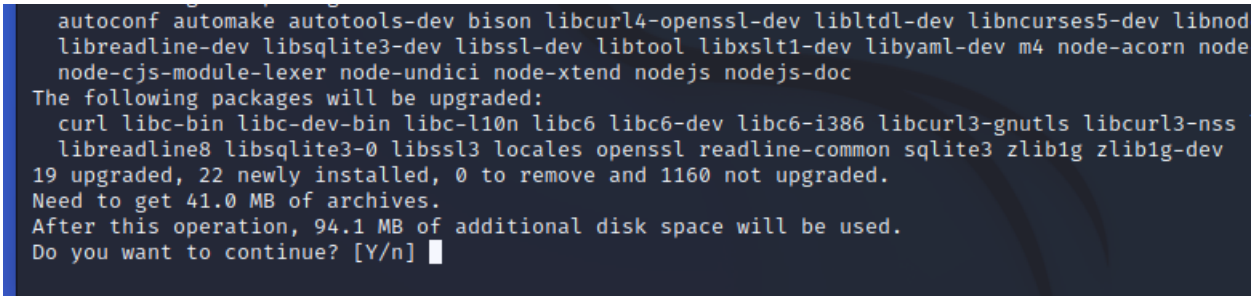
```
┌──(kali㉿kali)-[~]
└─$ cd Desktop

┌──(kali㉿kali)-[~/Desktop]
└─$ sudo git clone https://github.com/beefproject/beef.git
Cloning into 'beef'...
remote: Enumerating objects: 49397, done.
remote: Counting objects: 100% (24/24), done.
remote: Compressing objects: 100% (20/20), done.
remote: Total 49397 (delta 12), reused 12 (delta 4), pack-reused 49373
Receiving objects: 100% (49397/49397), 21.32 MiB | 6.76 MiB/s, done.
Resolving deltas: 100% (31065/31065), done.
```

3. Cambiamos de directorio al repositorio que acabamos de clonar, e insertamos el siguiente comando:

```
cd beef
sudo ./install
```



En la confirmación, insertamos yes.



En la siguiente confirmación, de igual manera insertamos yes.

4. Instalamos otra dependencia con el siguiente comando:

```
sudo bundle install
```

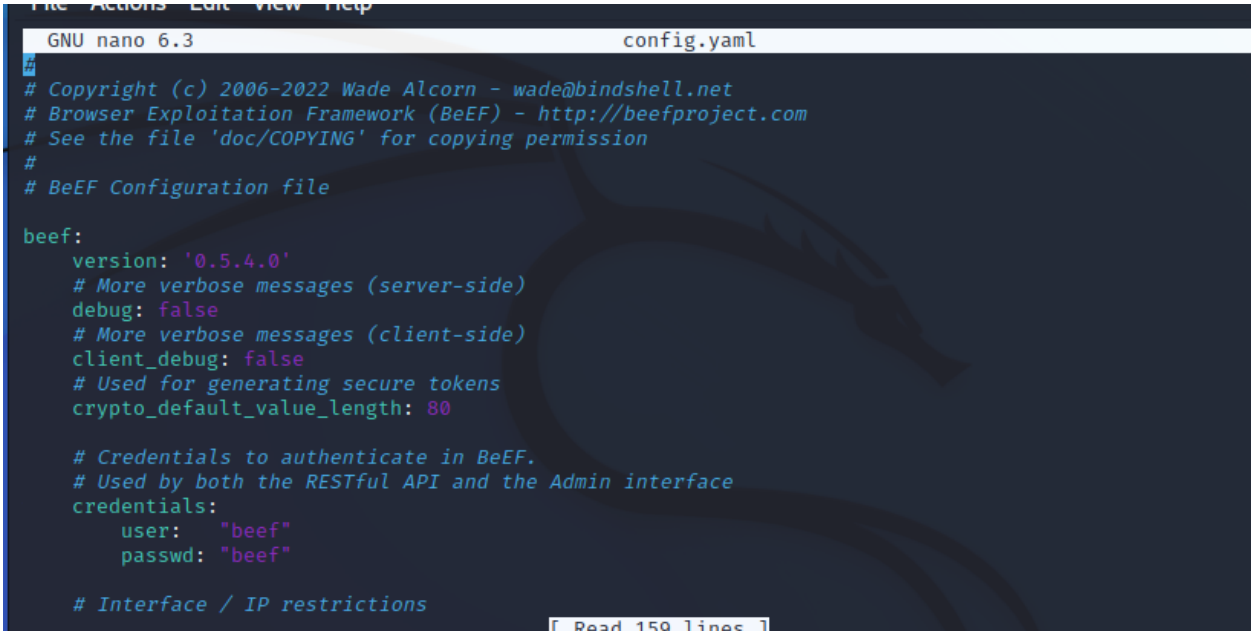
```
┌──(kali㉿kali)-[~/Desktop/beef]
└─$ sudo bundle install
Don't run Bundler as root. Bundler can ask for sudo if it is needed, and installing your bundle as r
will break this application for all non-root users on this machine.
Using rake 13.0.6
Using concurrent-ruby 1.1.10
Using i18n 1.12.0
Using minitest 5.16.3
Using tzinfo 2.0.5
Using activesupport 7.0.4
Using activemodel 7.0.4
Using activerecord 7.0.4
Using public_suffix 5.0.0
Using addressable 2.8.1
Using ansi 1.5.0
Using ast 2.4.2
Using fiber-local 1.0.0
```

5. Existe un archivo en nuestro directorio de la herramienta y debemos cambiar las credenciales de ingreso, así que listamos el contenido del directorio:

```
┌──(kali㉿kali)-[~/Desktop/beef]
└─$ ls
arerules                    docs                        Rakefile
beef                        extensions                  README.md
beef_cert.pem               Gemfile                     RESTful-API.postman_collection.json
beef_key.pem                Gemfile.lock                scripts
BeEF.postman_environment.json generate-certificate      spec
config.yaml                 googlef1d5ff5151333109.html test
_config.yml                 install                     tools
conf.json                   INSTALL.txt                 update-beef
core                        modules                     VERSION
doc                         package.json
Dockerfile                  package-lock.json
```

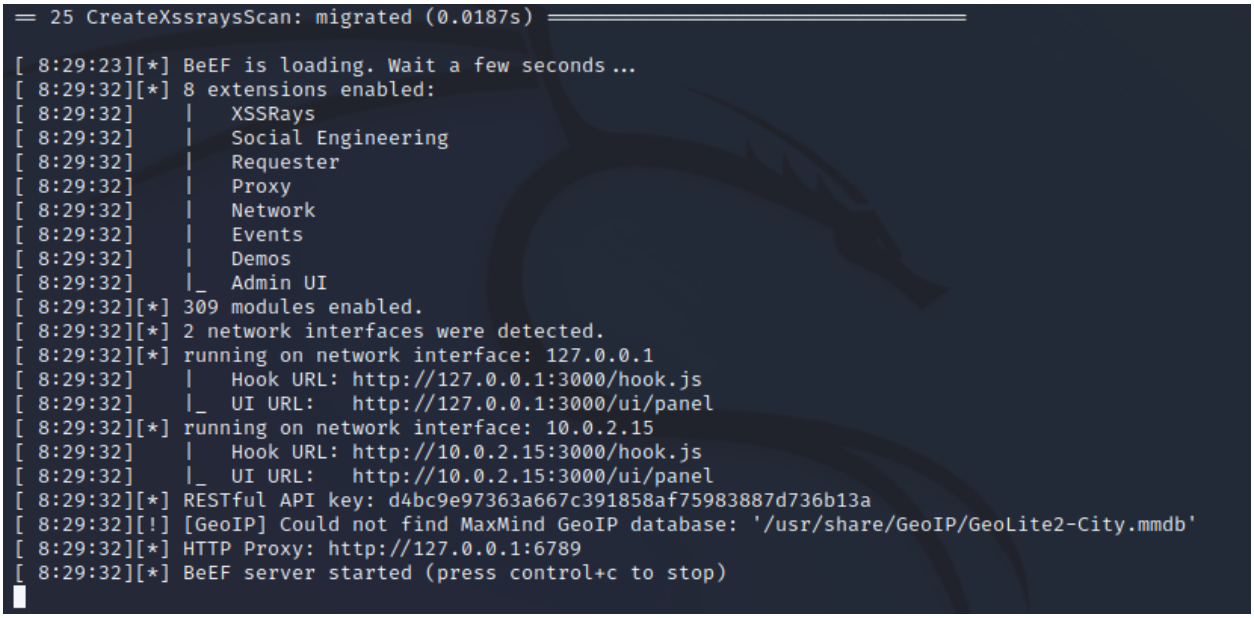
El archivo de configuración que debe ser editado es: config.yaml. En este caso usaremos nano para editarlo, de la siguiente manera:

```
sudo nano config.yaml
```



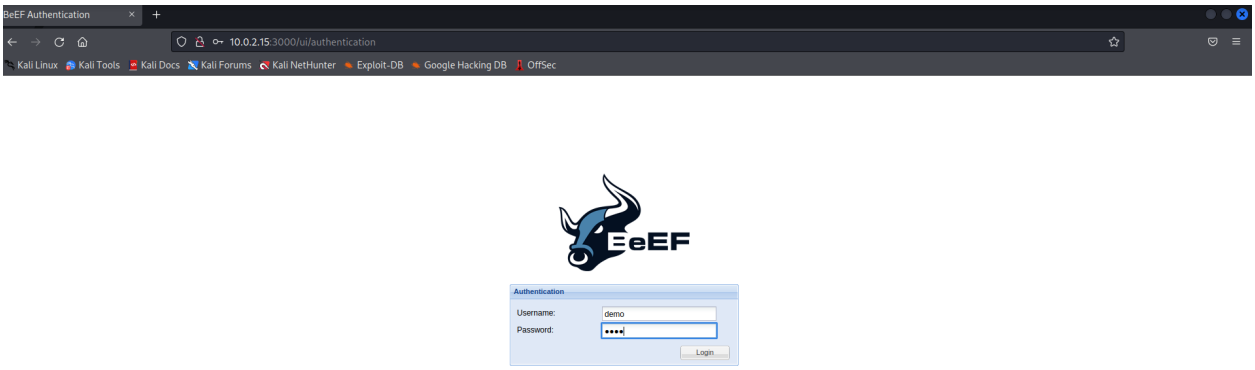
6. Editamos el usuario y contraseña en el archivo, estos cambios quedan a preferencia del lector.
7. Una vez modificado dicho archivo de configuración, podemos ejecutar la herramienta con el siguiente comando:

```
sudo ./beef
```

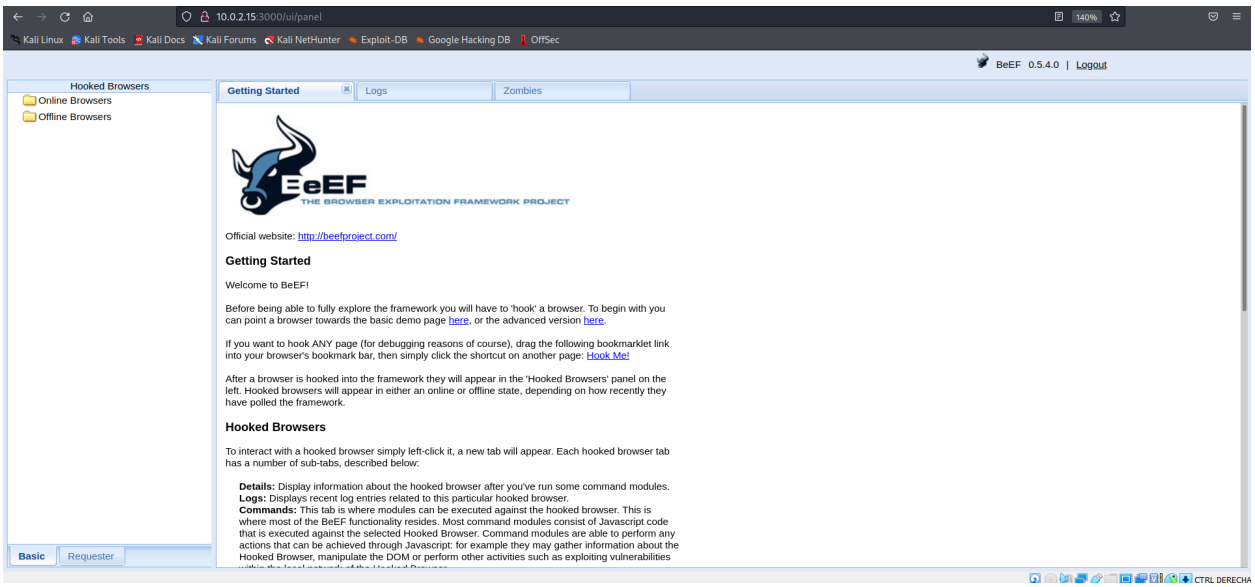


Uso de la herramienta

1. Con la ejecución de la herramienta, tenemos un link UI URL, lo abrimos en nuestro navegador e ingresamos las credenciales que modificamos en el archivo de configuración.

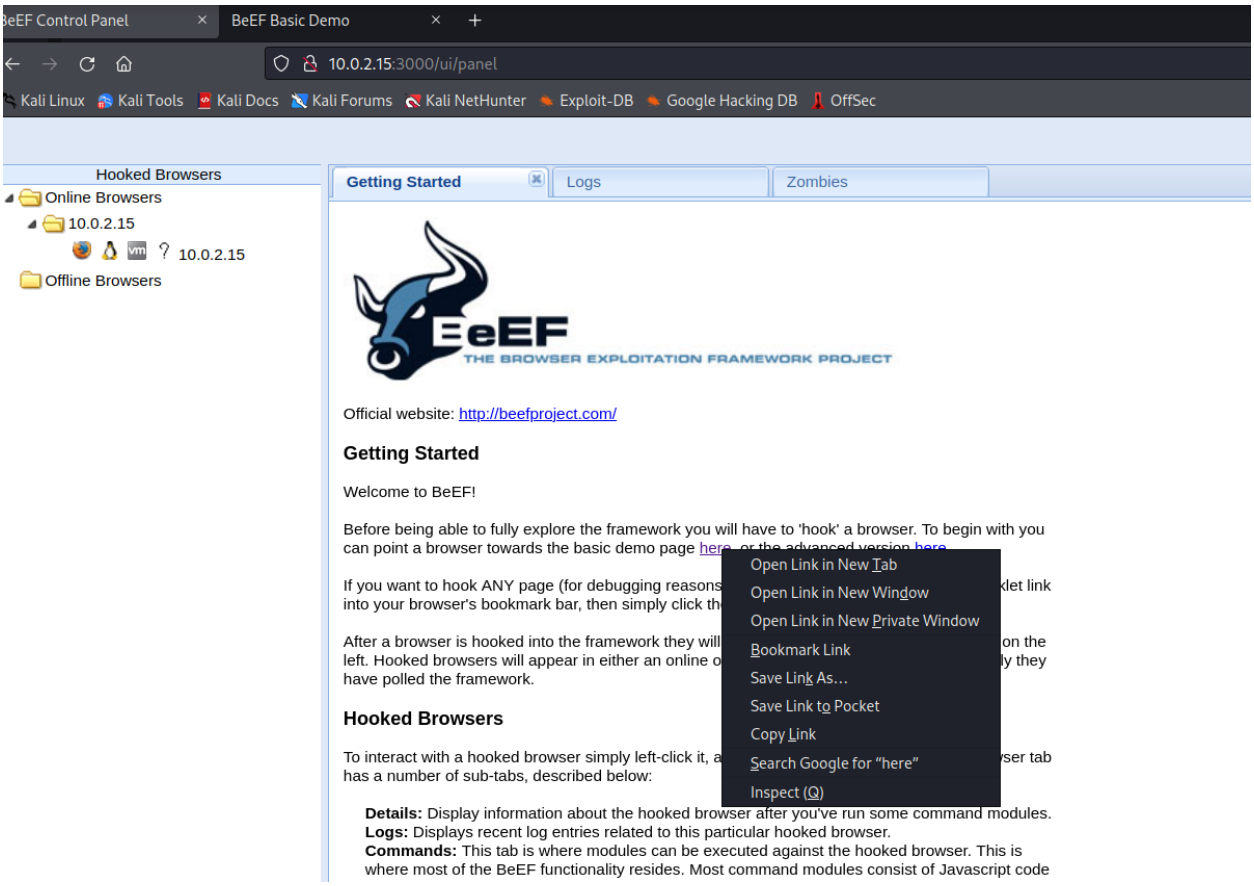


2. Una vez ingresemos podemos ver la siguiente interfaz:



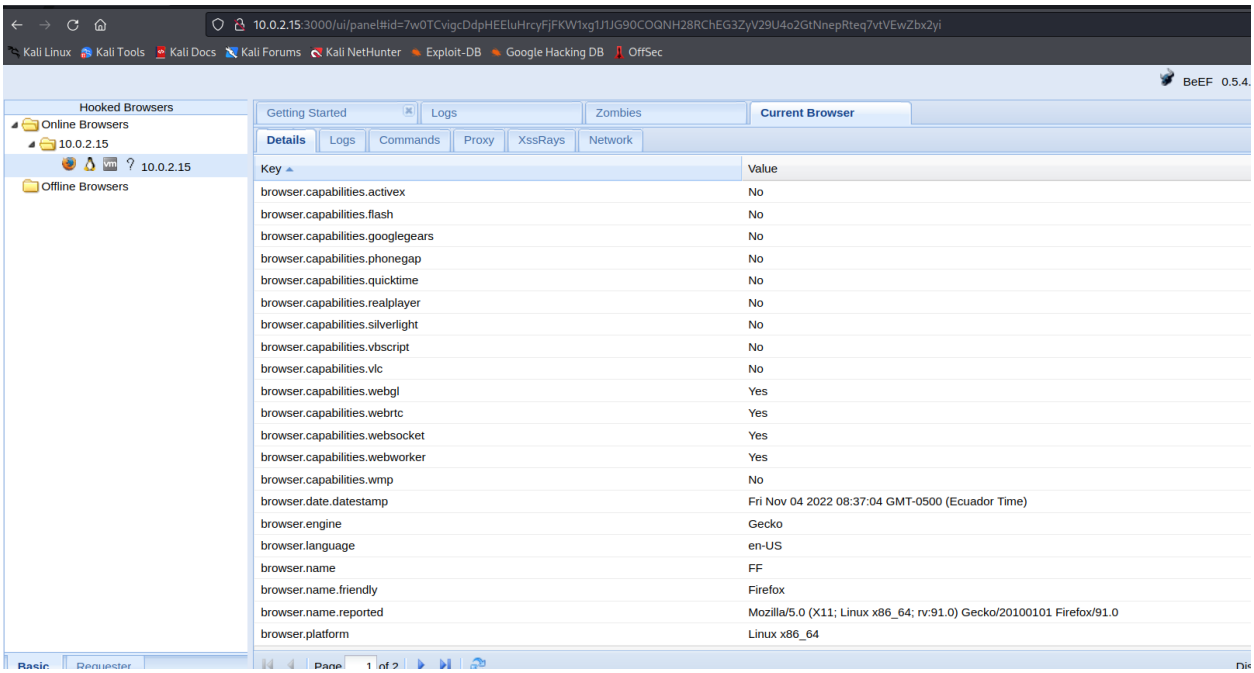
3. En la información podemos encontrar una demo básica, la cual nos genera un link y cualquier persona que ingrese a ese link, obtendremos información de tu

navegador, SO, configuraciones del browser, etc. En este caso usaremos la misma maquina virtual para acceder al link en el mismo browser.



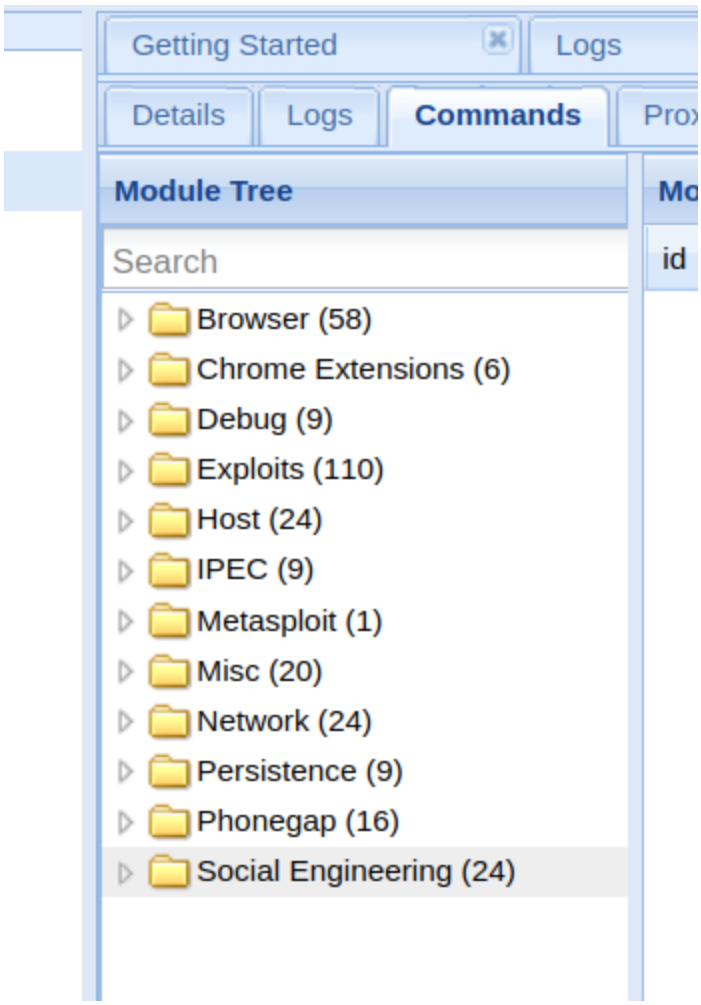
como podemos observar, abrimos el link en una nueva pestaña como indica en la parte superior, y en la sección de Online Browsers situado en la parte superior izquierda de la interfaz, vemos que ya se ha detectado una victima y adicionalmente vemos el SO y también que es una maquina virtual.

4. Hacemos clic en la ip de la victima y tendremos las siguientes opciones:

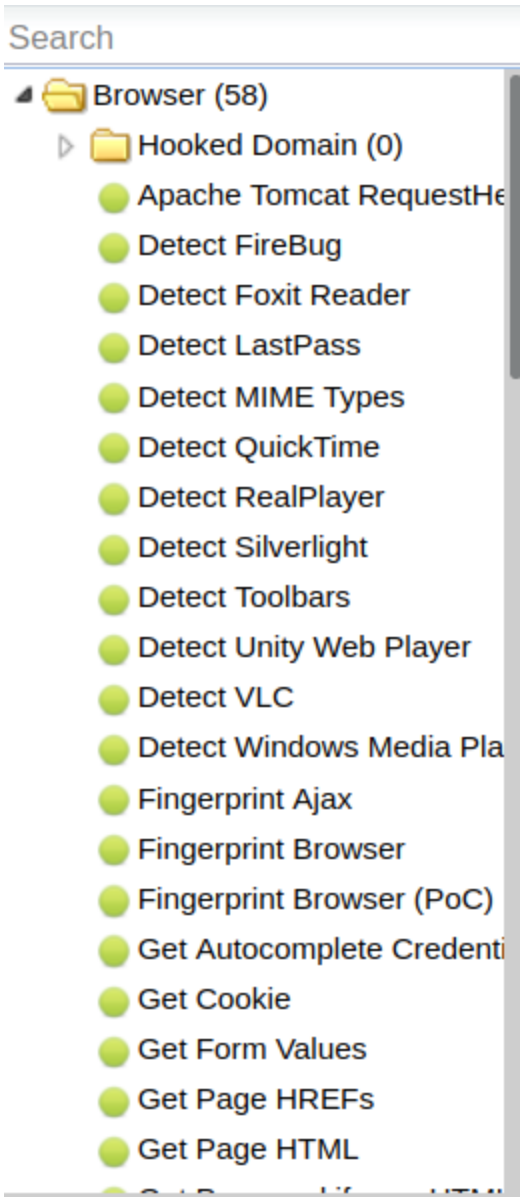


Como se menciono anteriormente tenemos mucha información de la victima, la cual nos puede ser útil para buscar una vulnerabilidad y tomar control total en el dispositivo de la victima.

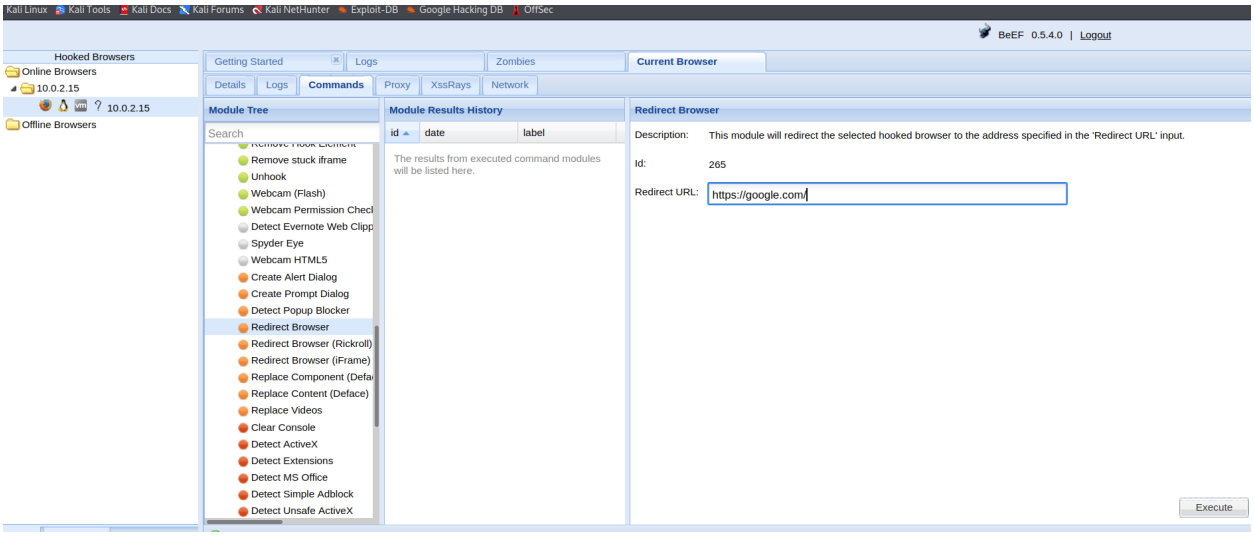
5. Para comenzar con la explotación, en la sección de “Commands” tenemos algunas opciones para vulnerar el dispositivo de la victima.



6. En este caso usaremos la primera de Browser y veremos que opciones tenemos:



Tenemos una lista extensa de posibles opciones, en este caso usaremos un redireccionamiento, la esta identificada con una viñeta de color naranja.



Vamos a enviar a nuestra victima a la pagina de google y para ello presionamos en el botón de Execute de la parte inferior derecha de nuestra interfaz.

Recordando que estamos usando la misma maquina virtual para la demo, previamente abrimos un link en el browser, abrimos dicha pestaña y en efecto hemos sido redirigidos a google.com.