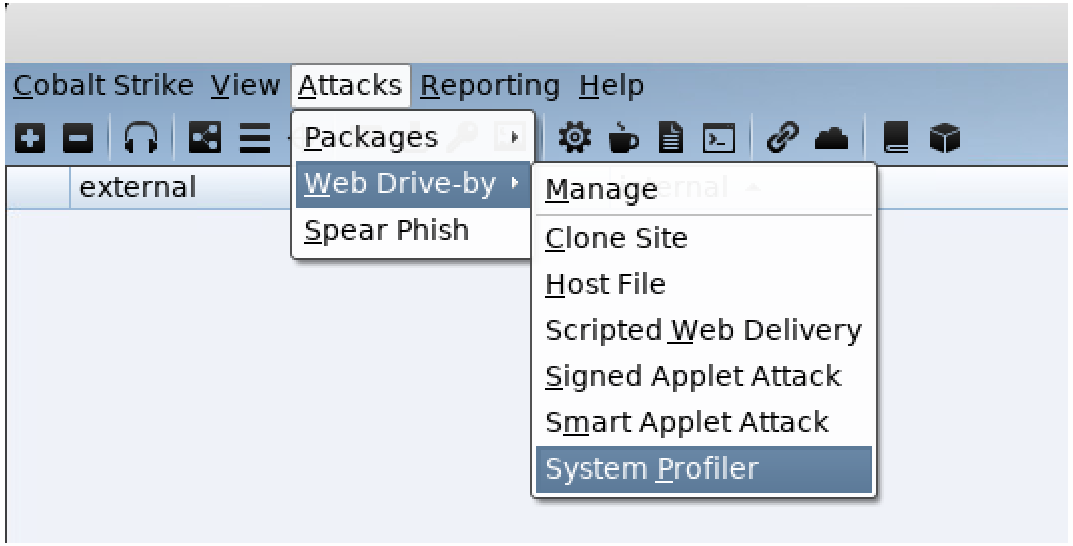
# Information Gathering With Cobalt Strike

RYAN SMITH / AUGUST 16, 2019

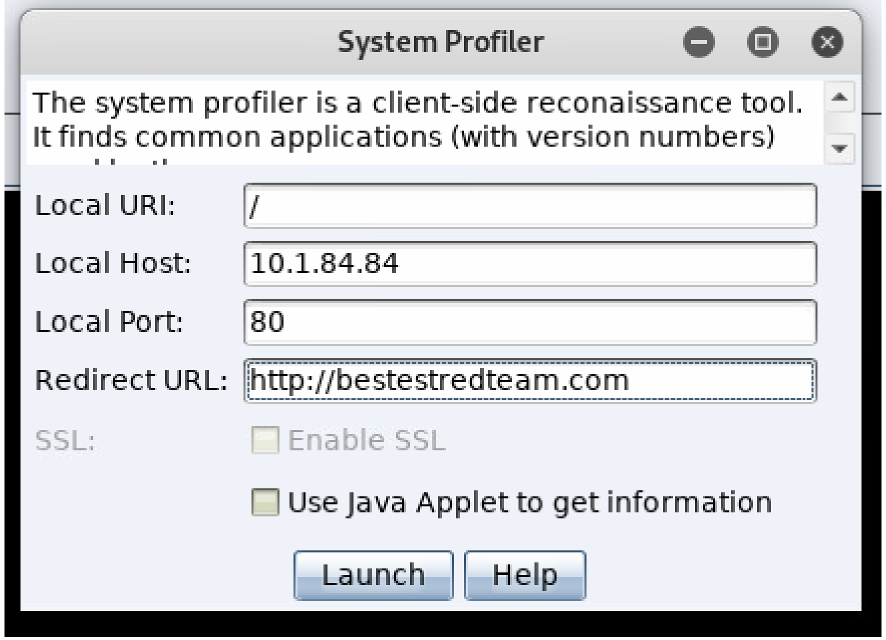
In a previous post, I began my exploration of Cobalt Strike. That post served as a sort of general overview of the tool. However, the more I look in to the tool, the more capabilities I learn about. Watching the fantastic Advanced Threat Tactics videos has been tremendously helpful. Taking heavy inspiration from them, I wanted to dig in to these capabilities. For this post, I wanted to learn more about the **System Profiler**. I figured as long as I was learning about it, I should write it up to share with the class.

## Setup

In the earlier post, I went over how to get started with the tool, so reference that for getting your teamserver and client up and running. For this post, I'll start from the main screen of Cobalt Strike. I'll first navigate to Attacks -> Web Drive-By -> System Profiler.



The dialogue box will pop up to configure our profiler. As with most modules within Cobalt Strike, it is pretty self-explanatory. Set up your URI, host, port, and  redirect URL.


System Profiler

The system profiler is a client-side reconaissance tool.
It finds common applications (with version numbers)

Local URI: /

Local Host: 10.1.84.84

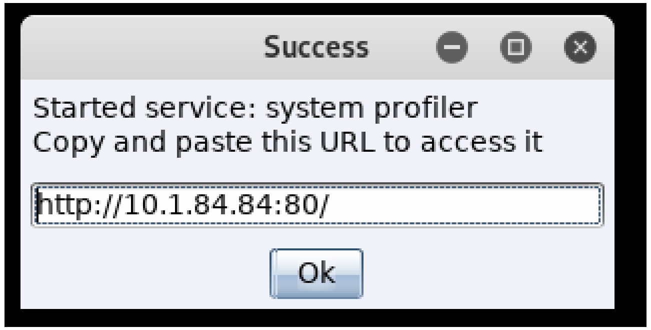
Local Port: 80

Redirect URL: http://bestestredteam.com

SSL: □ Enable SSL

□ Use Java Applet to get information

You'll also notice that I unchecked the "Use Java Applet to get information" box. In this video, it is explained that this is done because modern browsers would require a code-signing certificate in order for Java to work. Once we launch, we'll see a success box showing that the service has been started as well as the URL we should use for our attack.


Success

Started service: system profiler
Copy and paste this URL to access it

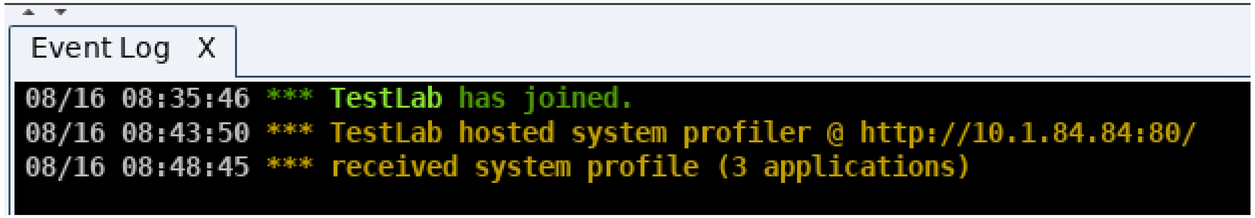
http://10.1.84.84:80/

A quick sidebar to talk about the lab setup. I'll be using a pretty stock Windows Server 2016 with Windows Defender removed. This is really just a result of me being lazy. I hope to eventually cover AV evasion techniques with Cobalt Strike, but for now I'll take the easy route.

### Usage

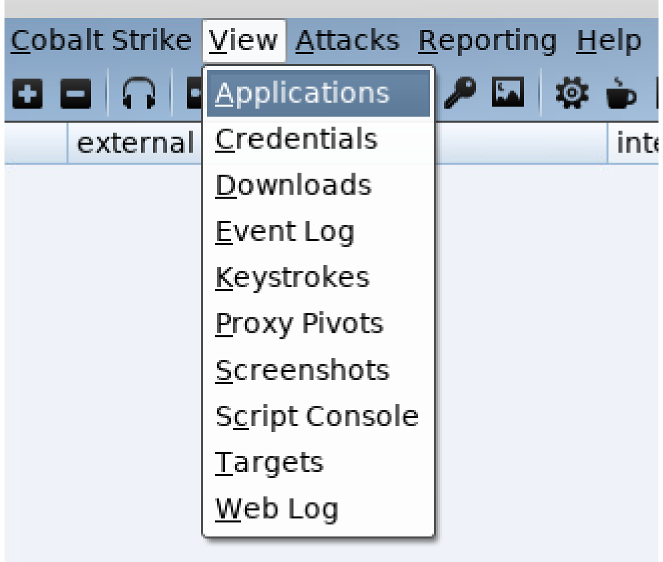
Now that we have our profiler started, we can move over to our victim machine and browse to the provided URL. Browsing to the URL will almost instantly redirect to the chosen redirect.

Back in Cobalt Strike, the connection has been logged in the event log.

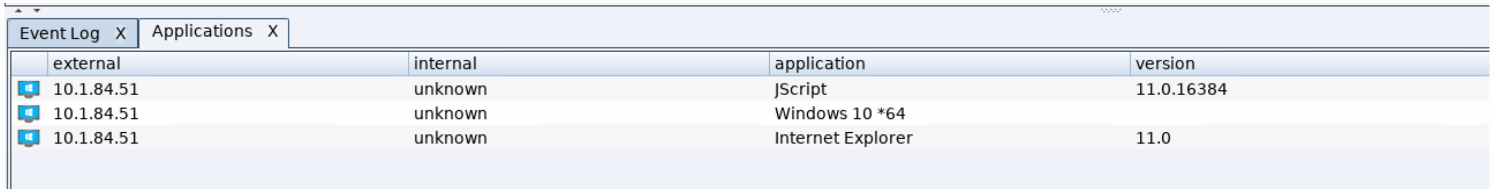


```
08/16 08:35:46 *** TestLab has joined.
08/16 08:43:50 *** TestLab hosted system profiler @ http://10.1.84.84:80/
08/16 08:48:45 *** received system profile (3 applications)
```

Notice how it mentions 3 applications. If I want to find more info about these applications, I'll browse to View -> Applications.

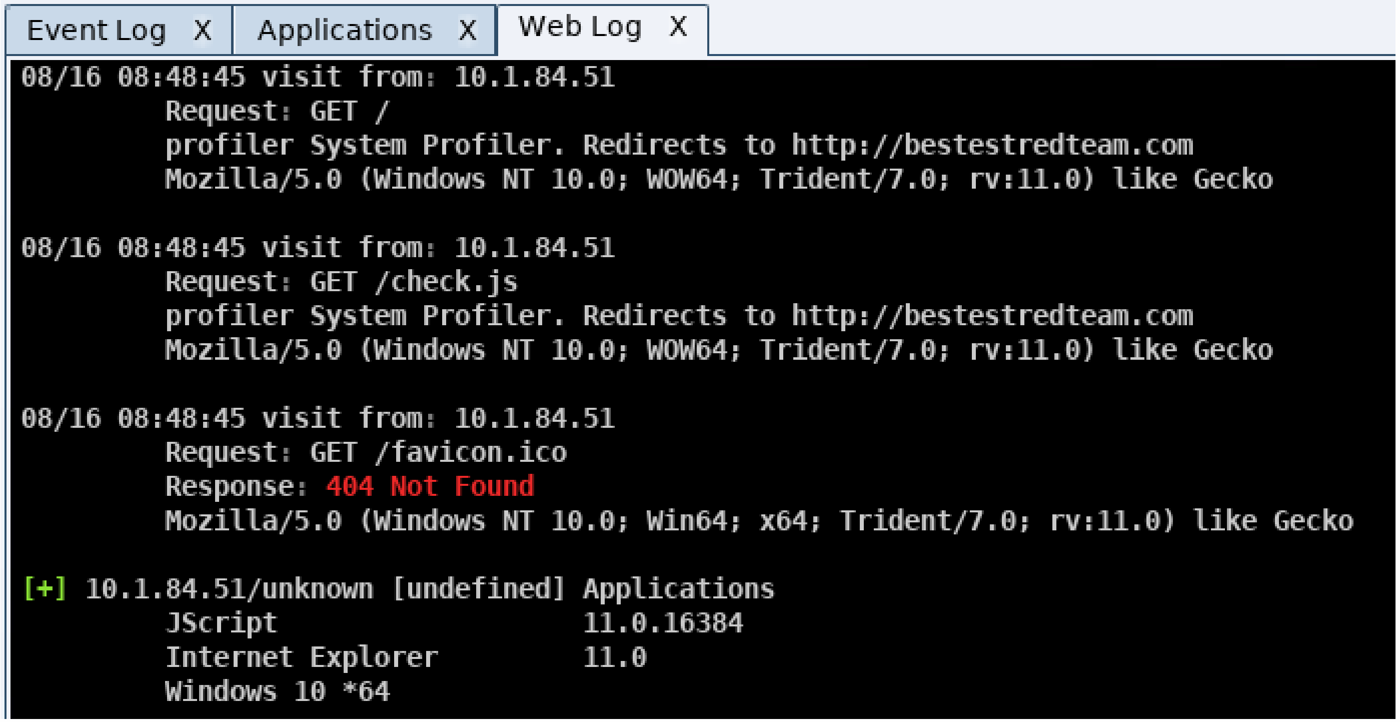


This will show me all the information the System Profiler was able to gather on the target machine's installed applications.



| external | internal | application | version |
|---|---|---|---|
| 10.1.84.51 | unknown | JScript | 11.0.16384 |
| 10.1.84.51 | unknown | Windows 10 *64 | |
| 10.1.84.51 | unknown | Internet Explorer | 11.0 |

Another source of information is the Web Log. Just go to View -> Web Log and there will be a full log of the actions the victim browser took.



```
08/16 08:48:45 visit from: 10.1.84.51
        Request: GET /
        profiler System Profiler. Redirects to http://bestestredteam.com
        Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko

08/16 08:48:45 visit from: 10.1.84.51
        Request: GET /check.js
        profiler System Profiler. Redirects to http://bestestredteam.com
        Mozilla/5.0 (Windows NT 10.0; WOW64; Trident/7.0; rv:11.0) like Gecko

08/16 08:48:45 visit from: 10.1.84.51
        Request: GET /favicon.ico
        Response: 404 Not Found
        Mozilla/5.0 (Windows NT 10.0; Win64; x64; Trident/7.0; rv:11.0) like Gecko

[+] 10.1.84.51/unknown [undefined] Applications
        JScript                     11.0.16384
        Internet Explorer           11.0
        Windows 10 *64
```

### Conclusion

I realize that this has been a rather short post, but that really goes to show just how easy the System Profiler is to use. In a real-world scenario, this would have to be paired with some social engineering to convince your victim to visit the URL. Set up well enough, the user might not even realize they've been tricked into sending over this information. Once the information has been obtained, it can be used to craft a very specific payload based on the running applications and their version numbers. If you're at all interested in learning more about Cobalt Strike, I highly recommend watching the Advanced Threat Tactics videos. Thanks for reading and never stop gathering information!

SHARE

TAGS: COBALT STRIKE INFORMATION GATHERING

## — ABOUT RYAN SMITH

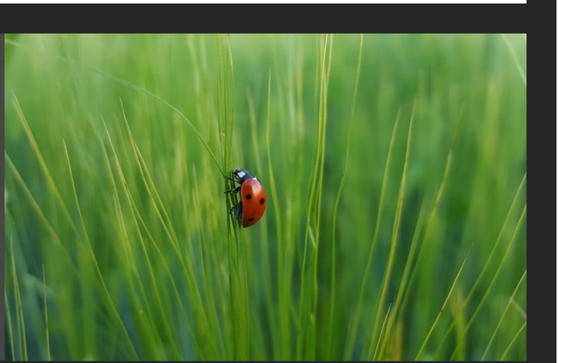
Ryan Smith is an information security professional specializing in penetration testing. He has years of experience both as an in-house pen tester and as a consultant.

SOUTH CAROLINA HTTPS://WWW.LINKEDIN.COM/IN/RYAN-SMITH-24A2B1127/
TWITTER

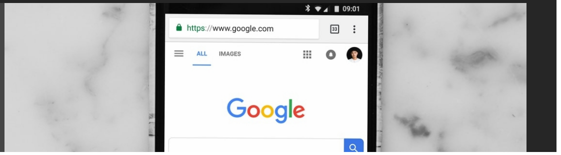
NEXT

**Stepping Into Debugging with GDB!**

SEPTEMBER 14, 2019

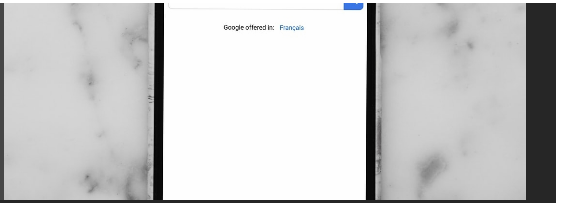


PREVIOUS



## Navigating To A Web Site Step By Step

AUGUST 01, 2019



### — ABOUT —

Two cybersecurity professionals trying to get better at all things security.

language_attributes(); ?>>
<head>
<meta charset="<?php bloginfo( 'charset' ); ?>" />
<meta name="viewport" content="width=device-width" />
<title><?php wp_title( '|', true, 'right' ); ?></title>
<link rel="profile" href="http://gmpg.org/xfn/11" />
<link rel="pingback" href="<?php bloginfo( 'pingback_url' ); ?>" />
<?php fruitful_get_favicon(); ?>
<!--[if lt IE 9]><script src="<?php echo
<?php wp_head(); ?>
</head>
<body <?php body_class();?>>
<div id="page-header" class="hfeed site">
<?php
$theme_options = fruitful_get_theme_options
$logo_pos = $menu_pos = '';
if (isset($theme_options['logo_position']))
$logo_pos = esc_attr($theme_options['logo_position']);
if (isset($theme_options['menu_position']))
$menu_pos = esc_attr($theme_options['menu_position']);
$logo_pos_class = fruitful_get_class_pos
$menu_pos_class = fruitful_get_class_pos

## — LATEST POSTS —

Container Escaper
SEPTEMBER 17, 2019

Stepping Into Debugging with GDB!
SEPTEMBER 14, 2019

Information Gathering With Cobalt Strike
AUGUST 16, 2019

## — AUTHORS —

- Bestest RedTeam
- Ryan Smith
- Ryan Villarreal

## — TAGS —

802.11 802.1X ACTIVE DIRECTORY ANTI-CSRF ASSEMBLY AUTOMATE AUTOMATION AWS BETA BETTERCAP BGP BITCOIN BLOODHOUND BLUE TEAM BURPSUITE BYPASS BYT3BL33D3R C PROGRAMMING C2 CA CAPTURE THE FLAG CERTIFICATES CLOUD CLUSTER CME

COBALT STRIKE CODING COMMAND AND CONTROL COMMAND LINE CONTAINER CORS CRACKMAPEXEC CSRF CTF CYBERSECURITY DEBUG

DEBUGGER DETECTION DOCKER DOMAIN ADMIN DOMAIN CONTROLLER DVWA ELEARNSECURITY ELK ELKSTACK ENUMERATION EWPT

EXECUTIONPOLICY FREERADIUS GDB GHOST GNU GNU RADIO GOOGLE CLOUD GOPHISH GRAPH THEORY HACKING HACKRF HASHCAT

HIJACKING HTTP HTTP/2 IMPACKET INFORMATION GATHERING INTERNAL NETWORK INTERNET OF THINGS JAVASCRIPT JUICESHOP JWT KALI LINUX

KALI TOOLS KERBEROS LATERAL MOVEMENT LINUX MERLIN MICROSOFT MICROSOFT OFFICE MINING NE0ND0G NEO4J NETWORKING

NULL SESSION OFFENSIVE SECURITY OFFSEC OPEN REDIRECT OSCE OSWP OWASP PASSWORD CRACKING PENETRATION TEST PENTEST PHISHING

PHP PINEAPPLE PIXEL TRACKING PORTAINER POST EXPLOITATION POWERSHELL PROTOCOLS PYTHON RADIO FREQUENCY RECON RED TEAMING

RED-BARON REDTEAMING REPORTING REVIEW RF RFC RTL-SDR S3 SAMBA SCANS SCAPY SCRIPTING SERVICE PRINCIPAL NAME SERVICES

SHODAN SMB SMBCLIENT SOCIAL ENGINEERING SOFTWARE DEFINED RADIO SPN SWARM SYSADMIN TERRAFORM TERRAFORMFUN TRAINING

UUID VULNERABILITY SCANNING WARDRIVING WEB APP WEB APPLICATION WEB TOKEN WEBAPP WIFI WIFU WIGLE WINDOWS WIRELESS

WPA XSS

OPINIONS EXPRESSED ARE SOLELY OUR OWN AND DO NOT EXPRESS THE VIEWS OR OPINIONS OF OUR EMPLOYERS.