Fork Sparta, Join The Legion

RYAN SMITH / MARCH 11, 2019

I've been given the directive at work to try to automate the things that I can. One thing I'd like to investigate automating is the discovery and recon portions of a pen test. I came across a tool that claims to do just that. Legion is "an open source, easy-to-use, super-extensible and semi-automated network penetration testing framework that aids in discovery, reconnaissance and exploitation of information systems." So let's dive in and see if Legion can help achieve my goals.First off, Legion is a fork of Sparta. Some of the key changes are:

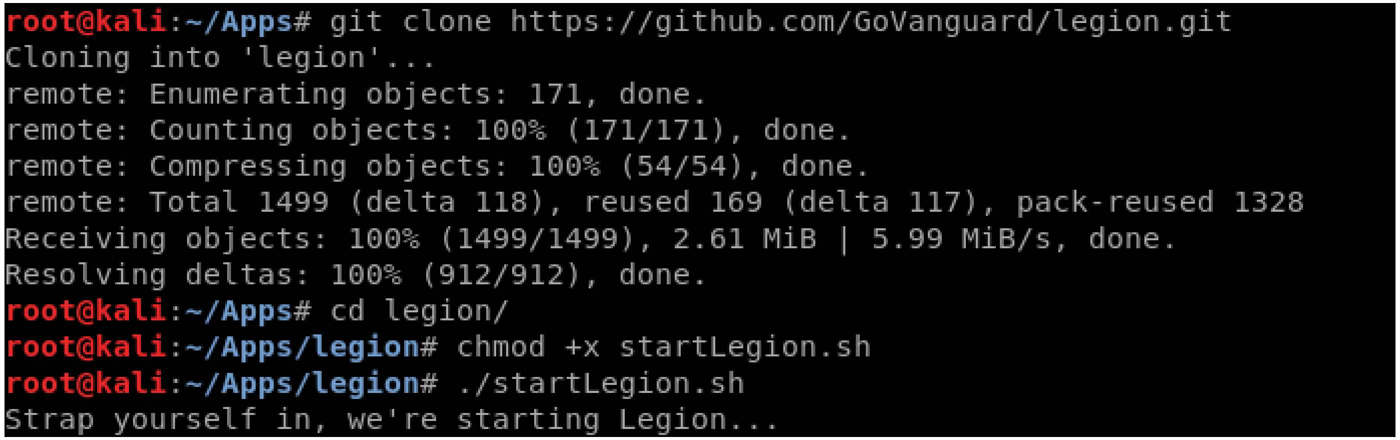
- Moving to Python 3.6.
- More intuitive GUI with things like task completion estimates, 1-click scans, and granular nmap scanning options.
- Simplification of installation (including a Docker container!).
- An active development team.

Installation really is pretty simple. Here are the steps:

```
git clone https://github.com/GoVanguard/legion.git
```

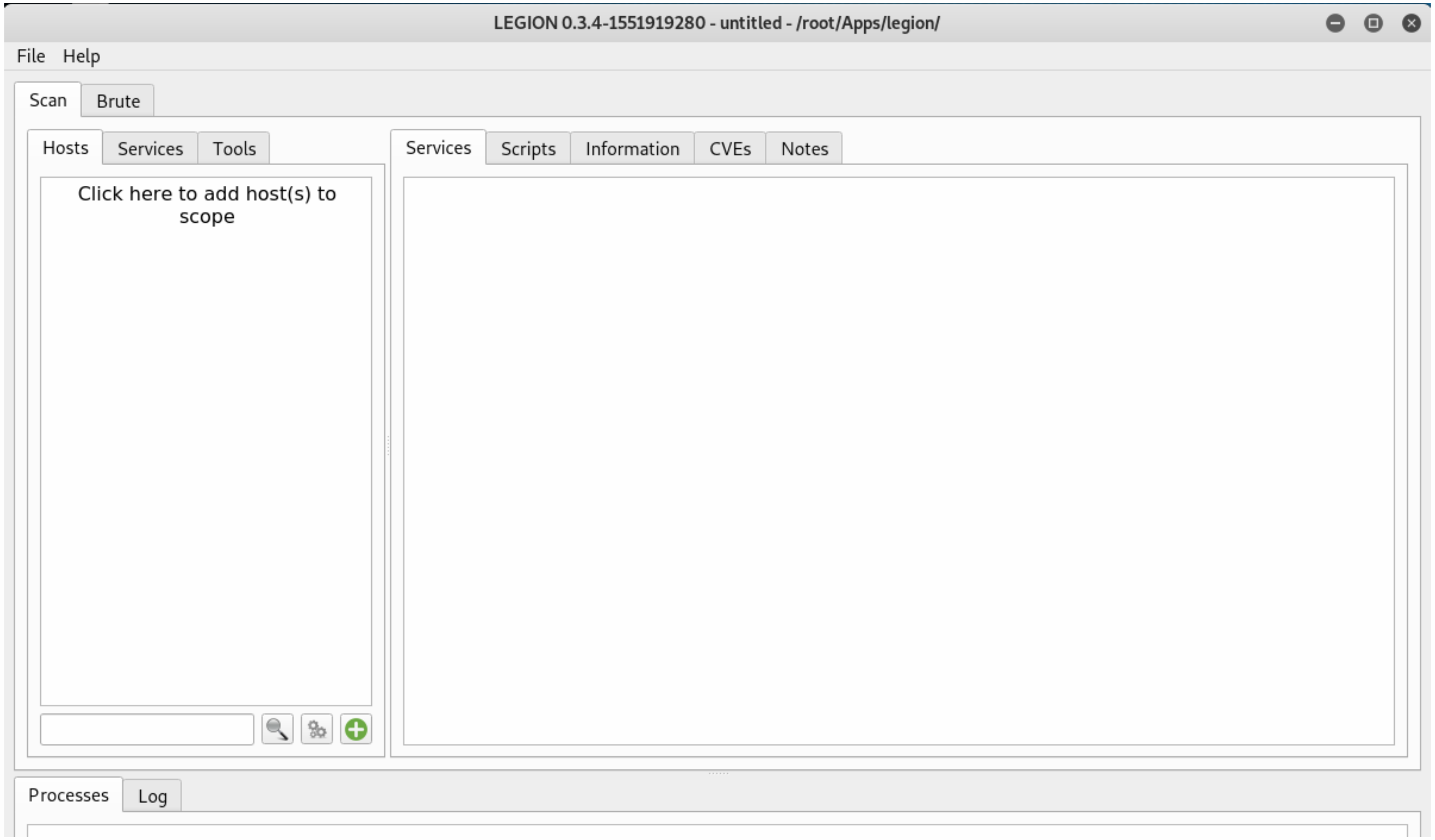
```
cd legion/
```

```
sudo chmod +x startLegion.sh
sudo ./startLegion.sh
```

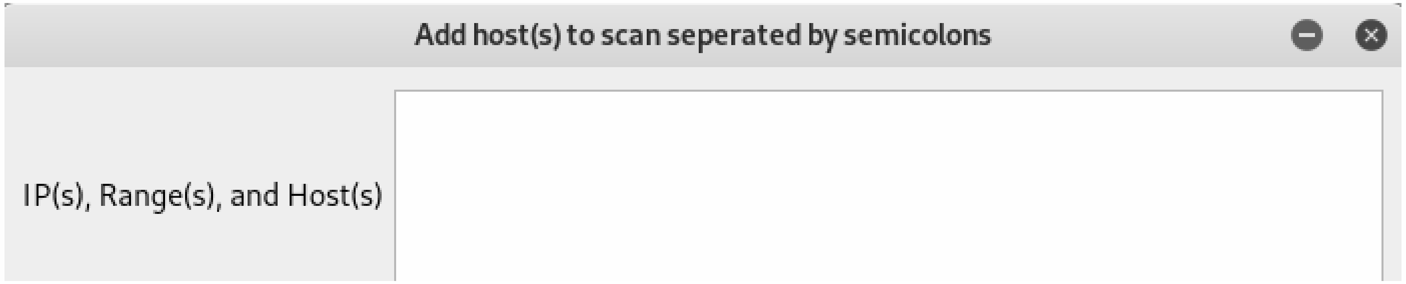


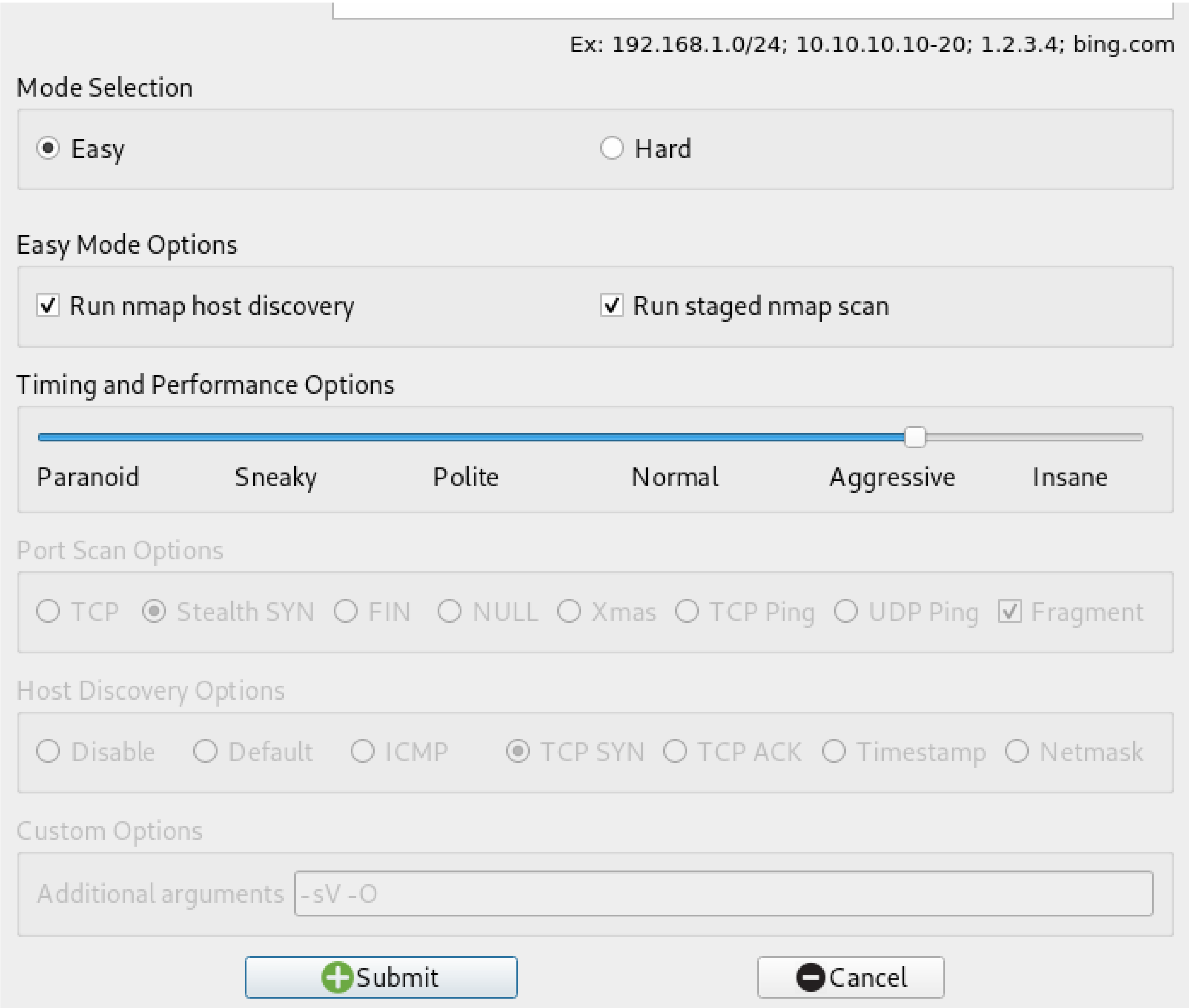
```
root@kali:~/Apps# git clone https://github.com/GoVanguard/legion.git
Cloning into 'legion'...
remote: Enumerating objects: 171, done.
remote: Counting objects: 100% (171/171), done.
remote: Compressing objects: 100% (54/54), done.
remote: Total 1499 (delta 118), reused 169 (delta 117), pack-reused 1328
Receiving objects: 100% (1499/1499), 2.61 MiB | 5.99 MiB/s, done.
Resolving deltas: 100% (912/912), done.
root@kali:~/Apps# cd legion/
root@kali:~/Apps/legion# chmod +x startLegion.sh
root@kali:~/Apps/legion# ./startLegion.sh
Strap yourself in, we're starting Legion...
```

Once the installation is completed, the application will launch itself. Here is what you will be greeted with:


LEGION 0.3.4-1551919280 - untitled - /root/Apps/legion/
File Help
Scan
Brute
Hosts
Services
Tools
Click here to add host(s) to scope
Services
Scripts
Information
CVEs
Notes
Processes
Log

Starting up is pretty intuitive, simply click the box under the "Hosts" tab and add some targets (IP addresses, hostnames, CIDR ranges).


Add host(s) to scan seperated by semicolons
IP(s), Range(s), and Host(s)



Ex: 192.168.1.0/24; 10.10.10.10-20; 1.2.3.4; bing.com

Mode Selection

● Easy ○ Hard

Easy Mode Options

☑ Run nmap host discovery ☑ Run staged nmap scan

Timing and Performance Options

Paranoid Sneaky Polite Normal Aggressive Insane

Port Scan Options

○ TCP ● Stealth SYN ○ FIN ○ NULL ○ Xmas ○ TCP Ping ○ UDP Ping ☑ Fragment

Host Discovery Options

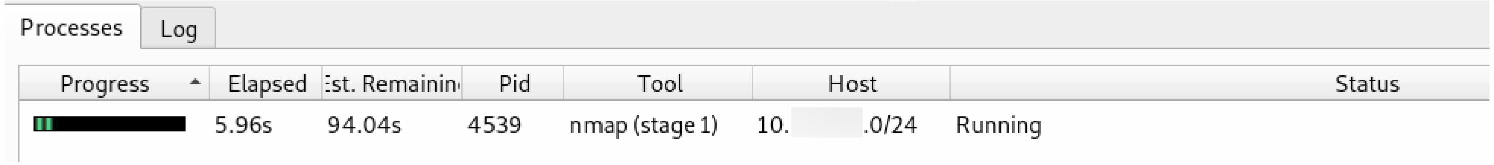
○ Disable ○ Default ○ ICMP ● TCP SYN ○ TCP ACK ○ Timestamp ○ Netmask

Custom Options

Additional arguments -sV -O

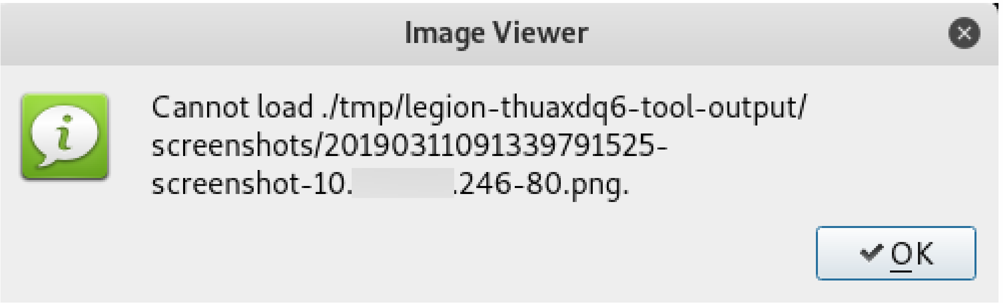
Selecting "Hard" mode allows you to fine tune the port scan, host discovery, and custom options. Once you're satisfied with the scope, select Submit.

At the bottom of the application, in the processes tab, you will see that the scan has already begun:



Progress	Elapsed	Est. Remainin	Pid	Tool	Host	Status
	5.96s	94.04s	4539	nmap (stage 1)	10.[redacted].0/24	Running

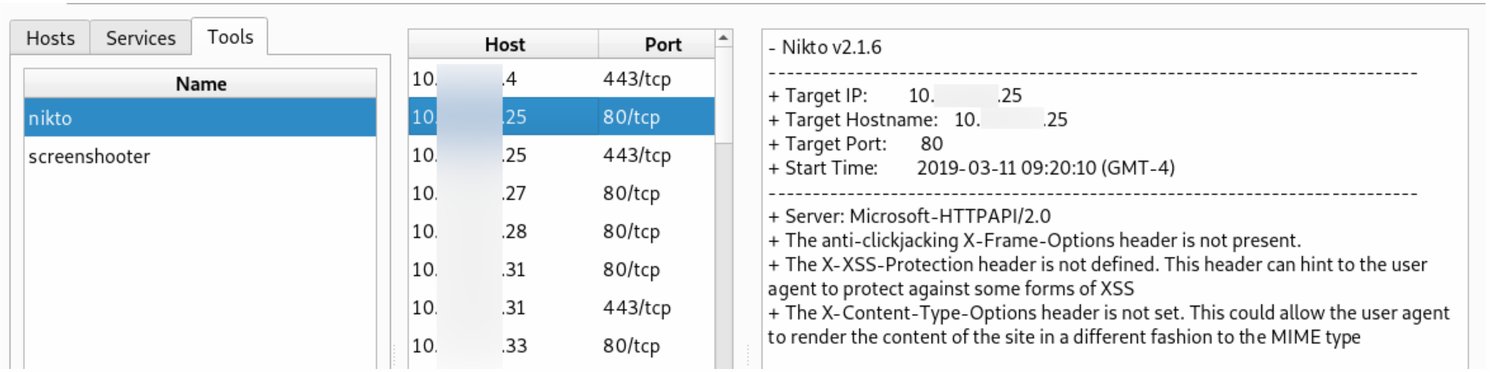
As the scan ran, it opened an "Image Viewer" window. However, I received an error message stating the the image could not be loaded. This eventually caused the application to crash.



Cannot load ./tmp/legion-thuaxdq6-tool-output/screenshots/20190311091339791525-screenshot-10.[redacted].246-80.png.

As the process runs, we can navigate through the Hosts, Services, and Tools tabs. There is even a search tool so that you can narrow down to specific hosts.

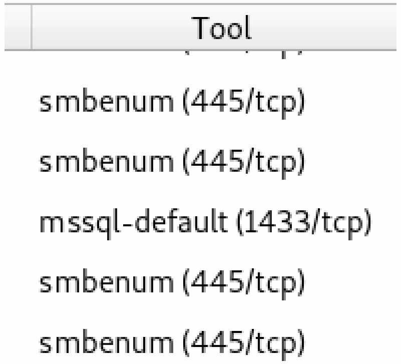
The hosts and services tabs display exactly what you would expect them to. The tools tab displays the different tools used against the hosts. For example, here Nikto was used and you can see the full output. Over time, the tool will continue to discover new information about your scoped hosts. For example, the hostname and OS. It will also run relevant tools such as `smbenum` for hosts that have port 445 open.



Name
nikto
screenshooter

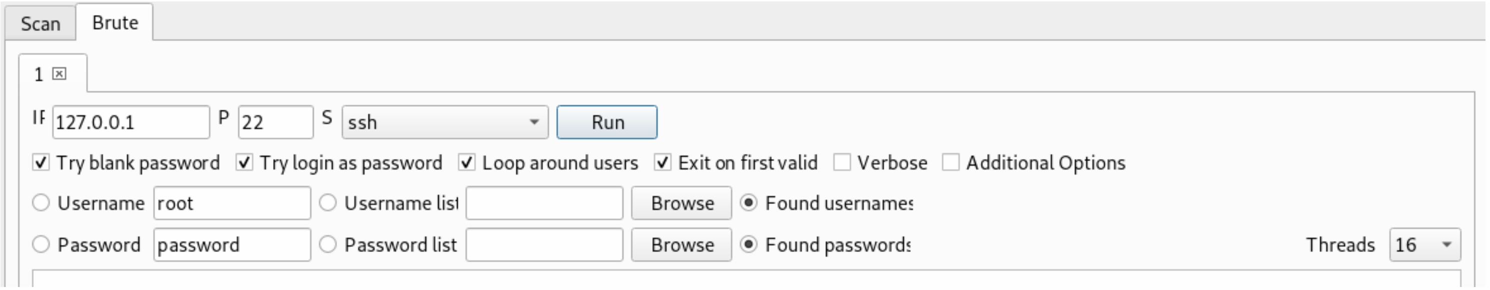
Host	Port
10.[redacted].4	443/tcp
10.[redacted].25	80/tcp
10.[redacted].25	443/tcp
10.[redacted].27	80/tcp
10.[redacted].28	80/tcp
10.[redacted].31	80/tcp
10.[redacted].31	443/tcp
10.[redacted].33	80/tcp

```
- Nikto v2.1.6
---------------------------------------------------------------------------
+ Target IP:          10.[redacted].25
+ Target Hostname:    10.[redacted].25
+ Target Port:        80
+ Start Time:         2019-03-11 09:20:10 (GMT-4)
---------------------------------------------------------------------------
+ Server: Microsoft-HTTPAPI/2.0
+ The anti-clickjacking X-Frame-Options header is not present.
+ The X-XSS-Protection header is not defined. This header can hint to the user
agent to protect against some forms of XSS
+ The X-Content-Type-Options header is not set. This could allow the user agent
to render the content of the site in a different fashion to the MIME type
```



Tool
smbenum (445/tcp)
smbenum (445/tcp)
mssql-default (1433/tcp)
smbenum (445/tcp)
smbenum (445/tcp)

There is also the "Brute" tab at the top. This allows you to run brute force attacks. For example, by default it fills in SSH with `root` and `password`. You can import a list of usernames or passwords to streamline your brute force attack. There are also several other options such as the number of threads, exiting on first valid, and verbose.



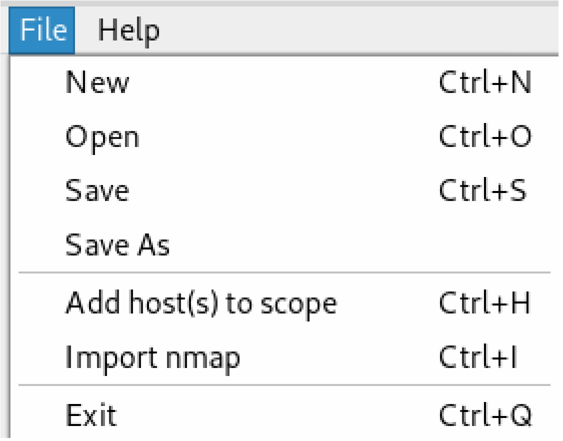
IP 127.0.0.1 P 22 S ssh

☑ Try blank password ☑ Try login as password ☑ Loop around users ☑ Exit on first valid □ Verbose □ Additional Options

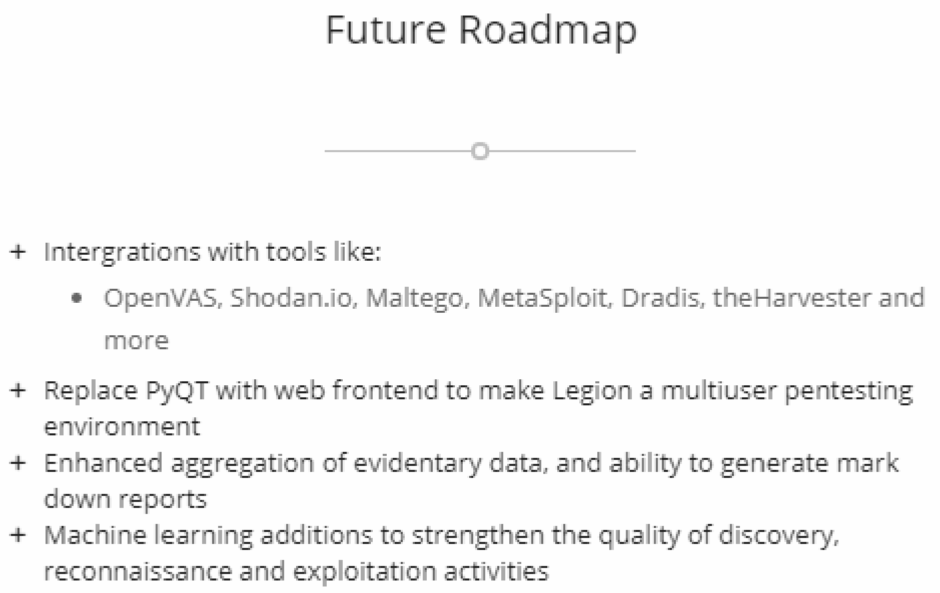
○ Username root ○ Username list ● Found usernames

○ Password password ○ Password list ● Found passwords Threads 16

Another feature worth pointing out is the ability to import an existing nmap scan. This could be useful if you've started a test already and want to let Legion do some more digging.



Overall, Legion does exactly what it claims to do. It even goes beyond my original expectations with the sensible exploitation of the hosts. It manages to automate a good portion of the early testing phases. At the time of writing, the current limitation that stands out to me is the lack of ability to export the data. Thankfully, that feature is on their roadmap.



Future Roadmap

- Intergrations with tools like:
  - OpenVAS, Shodan.io, Maltego, MetaSploit, Dradis, theHarvester and more
- Replace PyQT with web frontend to make Legion a multiuser pentesting environment
- Enhanced aggregation of evidentary data, and ability to generate mark down reports
- Machine learning additions to strengthen the quality of discovery, reconnaissance and exploitation activities

I'm really looking forward to deploying this tool on my next assessment!

SHARE

TAGS: RECON INFORMATION GATHERING PENETRATION TEST AUTOMATION

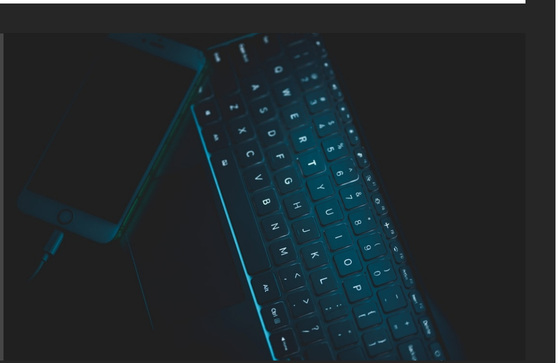
— ABOUT RYAN SMITH

TWITTER

NEXT

Bow Before the All Powerful CrackMapExec!!

MARCH 13, 2019



PREVIOUS

Digging into DNS

MARCH 07, 2019

— ABOUT —

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

```
<!DOCTYPE html>
<html <?php 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']);

$logo_pos_class = fruitful_get_class_pos($logo_pos);
$menu_pos_class = fruitful_get_class_pos($menu_pos);
```

— LATEST POSTS —

Information Gathering With Cobalt Strike
AUGUST 16, 2019

Navigating To A Web Site Step By Step
AUGUST 01, 2019

Atomic Red Team
JULY 30, 2019

— AUTHORS —

- Ryan Smith
- Bestest RedTeam
- Ryan Villarreal

— TAGS —

802.11 802.1X ACTIVE DIRECTORY ANTI-CSRF AUTOMATE AUTOMATION AWS BETA BETTERCAP BGP BITCOIN BLOODHOUND BLUE TEAM BURPSUITE BYPASS BYT3BL33D3R C2 CA CAPTURE THE FLAG CERTIFICATES CLOUD CLUSTER CME COBALT STRIKE COMMAND AND CONTROL

COMMAND LINE CONTAINER CORS CRACKMAPEXEC CSRF CTF CYBERSECURITY DETECTION DOCKER DOMAIN ADMIN DOMAIN CONTROLLER

DVWA ELEARNSECURITY ELK ELKSTACK ENUMERATION EWPT EXECUTIONPOLICY FREERADIUS GHOST 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 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.