Title: **CREDENTIAL HARVESTING IN ETHICAL HACKING**

AUTHOR: W. BRIAN NUKAM SMITH

COURSE: CYBER SECURITY

MATRICULE: TTSET25G024

INSTITUTE: PHIBMAT

DATE: JULY 30, 2025

1.1 OBJECTIVE

- Main Objective:

The primary objective of this ethical hacking lab is to demonstrate and understand the methods of credential harvesting, particularly through phishing techniques, within a controlled and legal environment. The goal is to simulate a real-world attack to study how attackers gather sensitive user information, like usernames, emails, and passwords, and how security teams can detect and defend against such attacks.

- Specific Objectives:
- To understand how phishing attacks are used to get user credentials.
- To create a phishing page that replicates a legitimate login interface like FACEBOOK, GOOGLE.
- To simulate a controlled phishing attempt and collect test credentials.
- To identify the tools used in credential harvesting (KALI LINUX “Setoolkit”).
- To recognize the associated risks and potential damages of phishing.
- To explore and recommend various defense mechanisms against phishing attacks.

1.2 LAB SETUP

To simulate a credential harvesting attack ethically, the following lab environment was created:

Environment Configuration:

- Operating System: Kali Linux (preferred ethical hacking OS)
- Target Platform: A cloned login page (e.g., Gmail, Facebook, or custom web app)
- Web Server: Apache (to host the fake login page)
- Internet Connection: Required for testing over a LAN or remote network
- Victim Simulation: A second machine or virtual machine to access the phishing site
- **Procedures:**

1. Kali Linux was installed and updated.

2. A social engineering tool (e.g., SET “Setoolkit”) was launched.

3. A phishing website was generated and hosted locally.

4. The phishing link was sent to a simulated user (second VM or test device).

5. The victim entered credentials, and data was captured in a log file.

1.3 Tools Used

Here are the tools used during the credential harvesting simulation:

- Kali Linux: Main OS for penetration testing
- SET (Social Engineering Toolkit): Used to craft phishing pages
- Apache Server: Hosting the phishing page
- HTML (Google.html): A basic HTML file created to redirect users to the fake page:

```
<html>
<body>
<a href="http://10.118.101.85">GOOGLE<a/>
<body/>
<html/>
```

- Web Browser (Chrome/Edge):   Used to test the phishing page as a victim

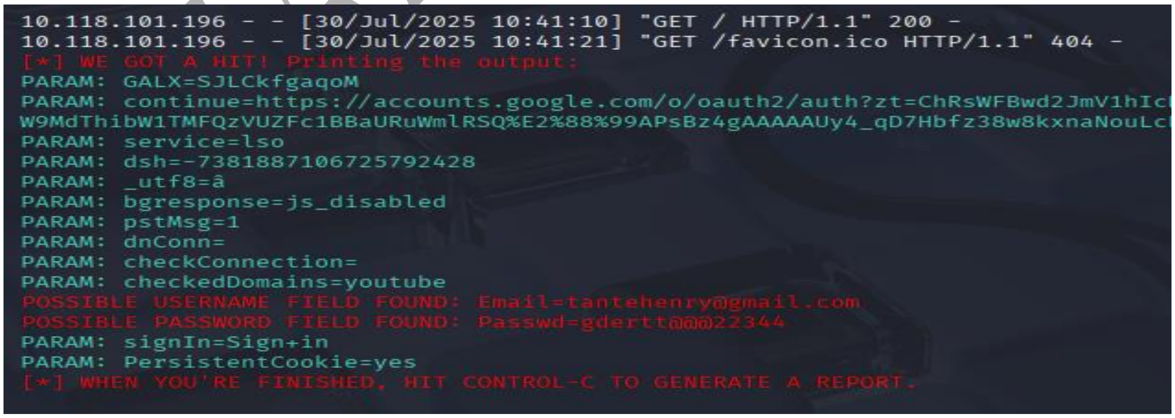
1.4 Execution Steps

Step-by-step Execution:

1. Launch the Terminal on Kali Linux.
2. Select Social Engineering Tool
3. Select a Target Platform
4. Choose a platform like Gmail, Facebook, or Twitter from the menu. The script auto-generates a fake login page hosted on localhost. A public URL is generated and shared with the test victim.
5. Hosting the Phishing Page
6. Victim Clicks and Enters Credentials. The phishing page is accessed by a test machine, and dummy credentials are entered.
7. Harvesting Credentials. Credentials are logged into a log.txt or usernames.txt file in the tool's folder.
8. Analyze and Report The contents of the captured credentials are reviewed in a secure and ethical manner.

1.5 Output Example

Captured Credentials (Sample from user’s information.txt):



```
10.118.101.196 - - [30/Jul/2025 10:41:10] "GET / HTTP/1.1" 200 -
10.118.101.196 - - [30/Jul/2025 10:41:21] "GET /favicon.ico HTTP/1.1" 404 -
[*] WE GOT A HIT! Printing the output:
PARAM: GALX=SJLCkfgaqoM
PARAM: continue=https://accounts.google.com/o/oauth2/auth?zt=ChRsWFBwd2JmV1hIc
W9MdThibW1TMFQzVUZFc1BBaURuWmlRSQ%E2%88%99APsBz4gAAAAAUy4_qD7Hbfz38w8kxnaNouLc
PARAM: service=lso
PARAM: dsh=-7381887106725792428
PARAM: _utf8=â
PARAM: bgresponse=js_disabled
PARAM: pstMsg=1
PARAM: dnConn=
PARAM: checkConnection=
PARAM: checkedDomains=youtube
POSSIBLE USERNAME FIELD FOUND: Email=tantehenry@gmail.com
POSSIBLE PASSWORD FIELD FOUND: Passwd=gdertt@@@22344
PARAM: signIn=Sign+in
PARAM: PersistentCookie=yes
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.
```

This output shows how credential data is harvested when a user enters information into the fake login page. No real accounts were compromised in this test.

1.6 Risks of Phishing Attacks

Credential harvesting poses serious security threats to individuals and organizations:

I. Data Breach: Stolen credentials can give attackers access to private emails, banking portals, or business systems.
II. Identity Theft: Attackers may impersonate users to commit fraud.
III. Unauthorized Access: Internal systems can be breached, resulting in financial and operational damages.
IV. Reputation Damage: Businesses that suffer from phishing attacks may lose customer trust.
V. Legal and Compliance Issues: Organizations may face fines under GDPR, HIPAA, etc., if data is mishandled.
VI. Botnet Recruitment: Stolen accounts may be used in spam campaigns or automated botnets.

1.7 Defense Mechanisms

To protect against phishing and credential harvesting, the following defense strategies are recommended:

Technical Defenses

Two-Factor Authentication (2FA): Adds an extra layer of security.

Spam Filters: Block suspicious or fake emails.

Web Filters: Detect and block malicious URLs.

SSL Certificates: Secure communications between users and servers.

User Awareness: Education Training and Teach users how to identify phishing attempts.

Don’t Click Unknown Links: Avoid clicking links from untrusted sources.

Check the URL Carefully: Look for small changes like *gma1l.com* instead of *gmail.com*.

Organizational Policies: Regular Security Audits

Email Authentication Protocols like SPF, DKIM, and DMARC.

Incident Response Plans in place for phishing detection and reaction.

In Conclusion, Credential Harvesting is a major cybersecurity threat, and ethical hackers play a vital role in identifying and fixing vulnerabilities before attackers exploit them. Through this lab, we demonstrated how phishing works, how credentials are harvested, and what tools and steps are involved in a controlled ethical scenario. As Ethical Hackers, we must always operate legally, with clear permissions, and with the goal of protecting systems and users, not exploiting them.