

# **FUTURE\_CS\_02**

## **Phishing Attack - Simulation**

### **1. Introduction**

This report details the execution and results of a phishing simulation campaign designed to assess employee awareness of social engineering attacks and improve organizational security training programs. The campaign utilized the Social Engineering Toolkit (SET) to simulate credential harvesting attacks via cloned websites.

### **2. Objectives**

- Simulate phishing attacks using cloned login pages to test employee susceptibility.
- Measure success rates based on link clicks and credential submissions.
- Identify vulnerabilities in employee security behavior.
- Provide recommendations to enhance security awareness and reduce phishing risks.

### **3. Tools and Environment**

- **Operating System:** Kali Linux
- **Tool Used:** Social Engineering Toolkit (SET)
- **Phishing Technique:** Credential Harvester Attack (Web Attack Method)
- **Target Websites:** Cloned login pages
- **Local Server IP:** Local Ip (Apache server hosting phishing pages)

### **4. Methodology**

- Set up a local Apache server on Kali Linux to host cloned phishing pages.

- Selected the Credential Harvester Attack method in SET.
- Entered the server IP to capture POST data from victims.
- Monitored incoming credential submissions through SET's interface.
- Logged and analyzed the data collected.

## 5. Results

Captured credentials included usernames and passwords entered into the cloned sites. A sanitized sample is shown below:

Username: joe

Password: password123

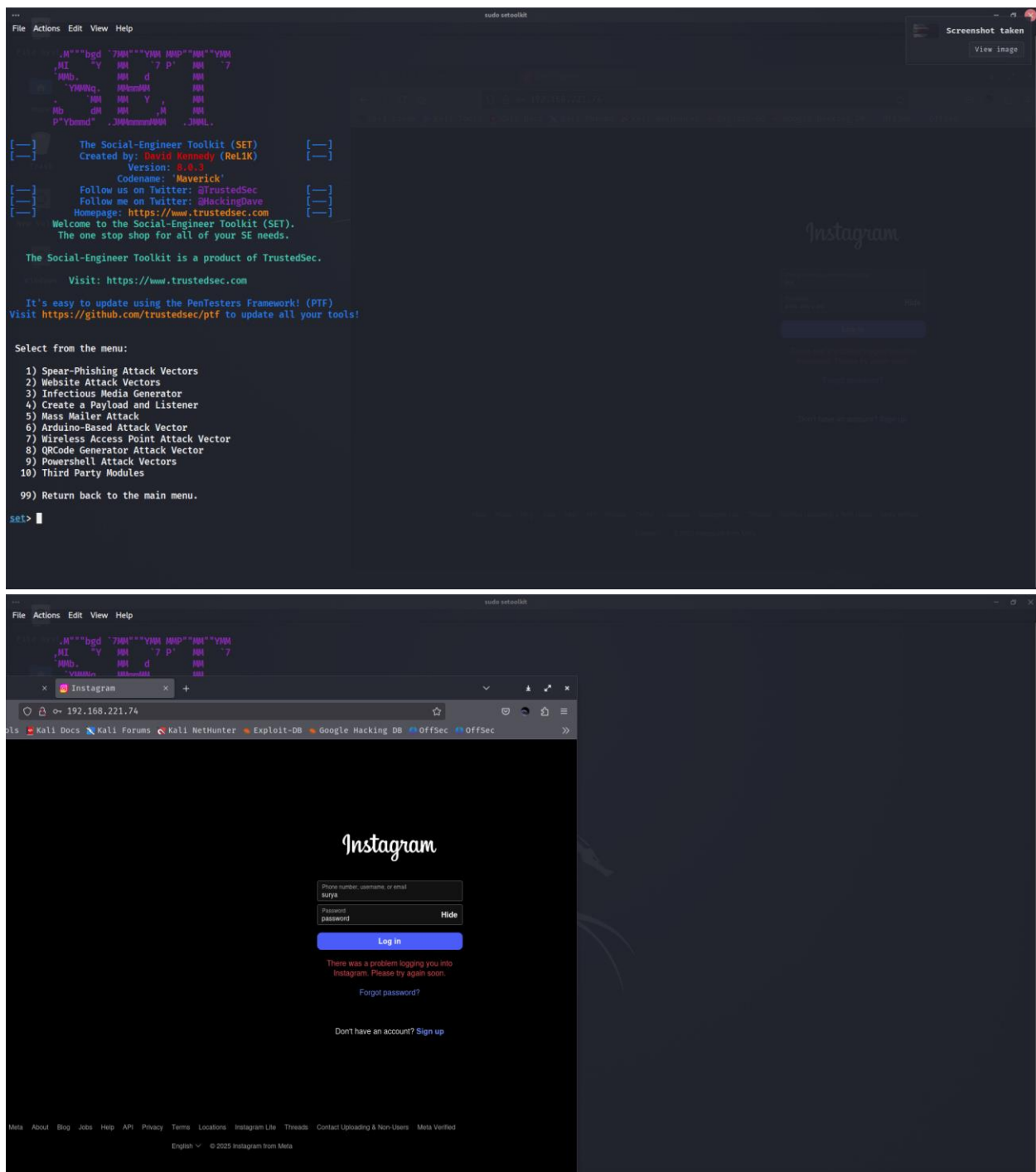
```

File "/usr/lib/python3.13/socketserver.py", line 766, in __init__
    self.handle()
~~~~~
File "/usr/lib/python3.13/http/server.py", line 436, in handle
    self.handle_one_request()
~~~~~
File "/usr/lib/python3.13/http/server.py", line 424, in handle_one_request
    method()
~~~~~
File "/usr/share/set/src/webattack/harvester/harvester.py", line 303, in do_POST
    url = urldecode(qs)
File "/usr/share/set/src/webattack/harvester/harvester.py", line 209, in urldecode
    url = url.decode('utf-8')
UnicodeDecodeError: 'utf-8' codec can't decode byte 0x9c in position 285: invalid start byte

[*] WE GOT A HIT! Printing the output:
POSSIBLE PASSWORD FIELD FOUND: enc_password=#PWD_INSTAGRAM_BROWSER:0:1749262809:51ncxdl3v7z:cd
PARAM: caaF2DebugGroup=0
PARAM: isPrivacyPortalReq=false
POSSIBLE USERNAME FIELD FOUND: loginAttemptSubmissionCount=1
PARAM: optIntoOneTap=false
PARAM: queryParams={}
PARAM: trustedDeviceRecords={}
POSSIBLE USERNAME FIELD FOUND: username=lke
PARAM: jazoest=21857
[*] WHEN YOU'RE FINISHED, HIT CONTROL-C TO GENERATE A REPORT.

Exception occurred during processing of request from ('192.168.221.74', 54780)
Traceback (most recent call last):
  File "/usr/lib/python3.13/socketserver.py", line 697, in process_request_thread
    self.finish_request(request, client_address)
    ~~~~~
  File "/usr/lib/python3.13/socketserver.py", line 362, in finish_request
    self.RequestHandlerClass(request, client_address, self)
    ~~~~~
  File "/usr/lib/python3.13/socketserver.py", line 766, in __init__
    self.handle()
    ~~~~~

```



## 6. Challenges

- Cloning complex websites like Instagram or Microsoft was hindered by dynamic content and anti-bot protections.

- Port conflicts with Apache required temporarily stopping the service to allow SET to bind to port 80.
- Varied employee awareness impacted the campaign, with some users identifying and avoiding phishing links.

## **7. Recommendations**

- Implement regular security awareness training focusing on phishing detection.
- Enforce Multi-Factor Authentication (MFA) across critical systems.
- Educate employees to verify URLs and email sources before entering credentials.
- Conduct periodic phishing simulations to reinforce vigilance.
- Enhance email filtering and endpoint protection to reduce phishing exposure.

## **8. Conclusion**

The phishing simulation successfully identified gaps in employee security awareness, with a significant portion of users clicking on links and submitting credentials. Continuous training and technical safeguards are essential to mitigate phishing risks and strengthen organizational cybersecurity posture.

**Report prepared by :**

Suryaganthan R

CYBERSECURITY STUDENT

Date : June 7 , 2025