



DEPI Project

24.10.2024

—

Name	Email
Nada Essa	nadaessa086@gmail.com
Nehal Sherif	nehalsherif54@gmail.com
Raghad Atef	raghad.atef20@gmail.com
Asmaa Abderahman	asmaaabdelrhman7654@gmail.com
Heba Samir	heba.samir@ejust.edu.eg

Overview

How can we effectively hide a malicious payload within an image, deliver it to unsuspecting users via a phishing email, and subsequently detect and prevent similar attacks using network security tools?

Tasks

Steghide to extract data from an image

1. Introduction

The objective of **Steghide** is to securely embed and conceal data (such as text, files, or messages) within image or audio files in a way that is imperceptible to the human eye or ear. This allows for covert communication or data storage, while protecting the hidden content from detection and unauthorized access.

```
(raghad@raghad)-[~]
$ steghide embed -cf images.jpeg -ef secret.txt
Enter passphrase:
Re-Enter passphrase:
embedding "secret.txt" in "images.jpeg" ... done

(raghad@raghad)-[~]
$
```

2. Practical example

First I uploaded an image and created a text file, then used the above command to hide the text file inside the image and the image didn't change to the user though the text file is hidden inside of it.

```
raghad@raghad: ~
File Actions Edit View Help
(raghad@raghad)-[~]
$ ls
Desktop Documents Downloads Music Pictures Public Templates Videos images.jpeg secret.txt

(raghad@raghad)-[~]
$ steghide extract -sf images.jpeg
Enter passphrase:
wrote extracted data to "secret.txt".
```

After that, I used the above command to extract the data inside the image and successfully got the text file extracted from the image.

Phishing Simulation Task Report using GoPhish

1. Introduction

This report documents the process of conducting a phishing simulation using GoPhish, a widely used open-source phishing framework. The objective of the task was to create and launch a phishing email campaign targeting users with a fake job offer from Google. The email included an embedded image with a malicious link, designed to test the vulnerability of recipients to phishing attacks.

2. Objective

The main goal of this task was to simulate a realistic phishing attack, designed to:

- Trick recipients into clicking on a malicious link by offering a job opportunity at a reputable company (Google).
- Embed the malicious link within an image to bypass suspicion and track user interaction with the email.

The task was carried out as part of a broader cybersecurity awareness campaign aimed at educating users on the dangers of phishing emails.

3. Phishing Email Content

The phishing email was designed to resemble a genuine job offer from Google. The key components of the email included:

- A **subject line** to catch the recipient's attention: "Job Opportunity at Google - Apply Now!"
- **Body text** that described a fake job offer and encouraged the recipient to click an embedded image to get more information.

Job Opportunity at Google - Apply Now!



nehalsherif54@gmail.com
To You

7:33 p.m.



Join Google - Your Career Starts Here!

Dear Candidate,

Congratulations! Based on your impressive professional background, we are pleased to offer you an exciting job opportunity with **Google**. This could be the perfect role for you to advance your career.

Click the image below to view the job details and begin the application process. This is your chance to be part of a global leader in technology and innovation!



We encourage you to apply as soon as possible to take advantage of this unique opportunity.

Best regards,
Google Recruitment Team

© 2024 Google LLC, 1600 Amphitheatre Parkway, Mountain View, CA 94043, USA



Virus-free. www.avast.com

Conclude a malicious activity and Scan the device network

Setup

- Target Machine (Victim): Ubuntu Desktop With SSH, Telnet and Apache2 enabled (IP: 192.168.15.129)
- Attacker Machine: Kali Linux with Nessus installed.
- Network Range: 192.168.15.0/24 (Local Network)

Scenario

- Scanning the network using Nessus to find the current Hosts.

The screenshot shows the 'Host Discovery' scan results in Nessus. The interface includes tabs for 'Hosts', 'Vulnerabilities', and 'History'. The 'Hosts' tab is active, showing a list of discovered hosts. A search bar is present with the text 'Search Hosts'. The results table shows one host: 192.168.15.129. To the right, the 'Scan Details' panel shows: Policy: Host Discovery, Status: Completed, Severity Base: CVSS v3.0, Scanner: Local Scanner, Start: Today at 2:26 AM, End: Today at 2:27 AM, Elapsed: a minute. Below this is a 'Vulnerabilities' section.

Host	Count
192.168.15.129	1

- Applying Network Scanning on the victim's machine.

The screenshot shows the 'Vulnerability Scan / Apache Httpd (Multiple Issues)' results in Nessus. The interface includes tabs for 'Hosts', 'Vulnerabilities', 'Remediations', and 'History'. The 'Vulnerabilities' tab is active, showing a list of vulnerabilities. A search bar is present with the text 'Search Vulnerabilities'. The results table shows several vulnerabilities, including 'Apache Web Servers', 'SSL General', 'HTTP Web Servers', 'SSH General', 'SSH Misc.', 'TLS (Multiple Issues)', and 'TLS Service detection'. To the right, the 'Scan Details' panel shows: Policy: Basic Network Scan, Status: Completed, Severity Base: CVSS v3.0, Scanner: Local Scanner, Start: Today at 2:30 AM, End: Today at 2:35 AM, Elapsed: 4 minutes. Below this is a 'Vulnerabilities' section with a donut chart showing the distribution of severity levels: Critical (red), High (orange), Medium (yellow), Low (green), and Info (blue).

Sev	CVSS	VPR	EPSS	Name	Family	Count
MIXED	ApaWeb Servers		3
MEDIUM	6.5			Une...	Misc.	1
MIXED	SSL General		4
LOW	2.1 *	4.2	0.8808	ICM...	General	1
INFO	HTTP Web Servers		5
INFO	SSH General		2
INFO	SSH Misc.		2
INFO	TLS (Multiple Issues)	tion	2
INFO	TLS Service detection		2

The screenshot shows the 'Vulnerability Scan / Apache Httpd (Multiple Issues)' results in Nessus. The interface includes tabs for 'Hosts', 'Vulnerabilities', 'Remediations', and 'History'. The 'Vulnerabilities' tab is active, showing a list of vulnerabilities. A search bar is present with the text 'Search Vulnerabilities'. The results table shows three vulnerabilities, all of which are 'HIGH' severity. To the right, the 'Scan Details' panel shows: Policy: Basic Network Scan, Status: Completed, Severity Base: CVSS v3.0, Scanner: Local Scanner, Start: Today at 2:30 AM, End: Today at 2:35 AM, Elapsed: 4 minutes. Below this is a 'Vulnerabilities' section with a donut chart showing the distribution of severity levels: Critical (red), High (orange), Medium (yellow), Low (green), and Info (blue).

Sev	CVSS	VPR	EPSS	Name	Family	Count
CRITICAL	9.8	6.0	0.0359	Apa...	Web Servers	1
HIGH	7.5	5.2	0.0013	Apa...	Web Servers	1
HIGH	7.5	5.1	0.0009	Apa...	Web Servers	1

- Applying Web Application Scanning on the victim's machine.

