Dr. Vishwanath Karad MIT World Peace University, Pune

Wireless Devices and Mobile Security
Third Year B. Tech, Semester 5

# Analysis and Demo of **PyPhisher**

Mini Project Report

Under the Guidance of
**Dr. Vinayak Musale**

Prepared By

Krishnaraj Thadesar, PA10, 1032210888
Sourab Karad, PA25, 1032211150
Soubhagya Singh, PA24, 1032211144
Vedang Khare, PA06

Department of School of Computer Engineering and Technology
Maharashtra, India.

2023-2024
December 8, 2023

# Contents

# Acknowledgment

I would like to express my deepest appreciation to all those who provided me the possibility to complete this report. A special gratitude I give to our mentor, Dr. Vinayak Musale, whose contribution in stimulating suggestions and encouragement, helped me to coordinate my project especially in writing this report.

Furthermore, I would also like to acknowledge with much appreciation the crucial role of the staff of MIT WPU, who gave the permission to use all required equipment and the necessary materials to complete the task. A special thanks goes to my team mates,who helped me enormously to assemble the parts and gave suggestion about the task of using the techniques of measurements.

I have to appreciate the guidance given by other supervisor as well as the panels especially in our project presentation that has improved our presentation skills thanks to their comment and advices.

I would also like to thank my parents for their wise counsel and sympathetic ear. You are always there for me. Finally, I wish to thank my friends for their support and encouragement throughout my study.

## Name of Students

1. Krishnaraj Thadesar, PA10, 1032210888
2. Sourab Karad, PA25, 1032211150
3. Soubhagya Singh, PA24, 1032211144
4. Vedang Khare, PA06

# Abstract

# List of Figures

# List of Tables

# Chapter 1

# Introduction

Phishing is one of the most prevalent forms of cyber-attack, often used to steal user data, including login credentials and credit card numbers. It occurs when an attacker, masquerading as a trusted entity, tricks a victim into opening an email, instant message, or text message. The recipient is then tricked into clicking a malicious link, which can lead to the installation of malware, the freezing of the system as part of a ransomware attack, or the revealing of sensitive information.

This project focuses on the analysis of PyPhisher, a phishing tool, along with a comparative study of similar tools. PyPhisher is a python tool that automates the process of phishing and has been widely used in the cybersecurity field for educational and research purposes. The tool provides a platform for security professionals to study phishing attacks and understand how they can be prevented.

As a Computer Science undergraduate in the Cybersecurity division of our university, I have undertaken this project to delve deeper into the mechanisms of phishing attacks, the effectiveness of tools like PyPhisher, and the countermeasures that can be employed to protect against such attacks. This report will provide an overview of my findings and observations, with the aim of contributing to the broader understanding of phishing threats and defenses.

The subsequent chapters will detail the methodology used in this analysis, the results obtained, and the implications of these findings for the field of cybersecurity.

### 1.0.1 Problem Statement

The problem statement of the project is to analyze PyPhisher, a phishing tool, and conduct a comparative study of similar tools. The objective is to understand the techniques and mechanisms used in phishing attacks and evaluate the effectiveness of phishing tools. Additionally, the project aims to demonstrate the power of phishing as a social engineering technique in action.

### 1.0.2 Need of the Project

The need for this project arises from the increasing prevalence of phishing attacks and the need to understand and mitigate their impact. Phishing attacks have become a significant threat to individuals, organizations, and even governments, leading to financial losses, data breaches, and compromised security. By analyzing PyPhisher and similar tools, the project aims to gain insights into the mechanisms of phishing attacks, evaluate the effectiveness of phishing tools, and explore countermeasures to protect against such attacks.

Furthermore, this project addresses the need for awareness and education regarding phishing attacks. Many individuals and organizations fall victim to phishing attacks due to a lack of knowledge about the techniques used by attackers. By studying and analyzing phishing tools, the project aims to raise awareness about the various tactics employed by attackers and educate users about the importance of vigilance and caution while interacting with suspicious emails, websites, and messages.

# Chapter 2

# Literature Survey

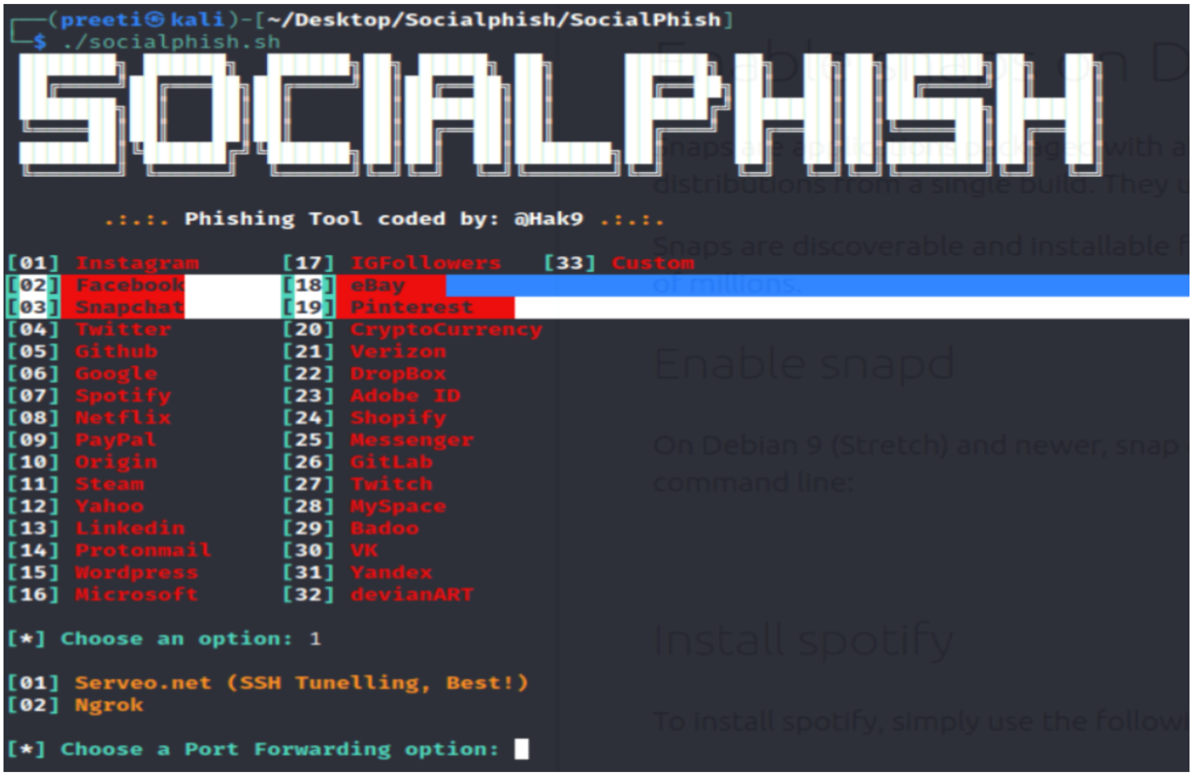
In the literature survey, we reviewed various existing systems and tools related to phishing attacks and their prevention. We explored different research papers, articles, and case studies to gain insights into the current state of the art in this field. The existing systems provided valuable information on the techniques used by attackers, the vulnerabilities they exploit, and the countermeasures employed by security professionals.

## 2.1 SocialPhish

Socialphish is an open-source phishing tool with a lot of features. Socialphish, which is used to conduct phishing attacks on targets, is growing increasingly popular. Socialphish is easier to use than Social Engineering Toolkit. Socialphish includes various templates created by another tool called Socialphish. Socialphish provides phishing templates for 33 famous websites, including Google, Facebook, Github, Yahoo, Snapchat, Spotify, Linkedin, Microsoft, Yahoo, Github, etc. Socialphish also allows users to utilize a custom template. This tool makes phishing attacks simple to carry out. They can use a lot of creativity to make the email appear as real as possible.

## 2.2 Features of SocialPhish

1. Socialphish is an open source tool.
2. Socialphish is a very simple and easy-to-use tool written in bash language.
3. It is a lightweight tool that does not require much storage space.



```
┌──(preeti㉿kali)-[~/Desktop/Socialphish/SocialPhish]
└─$ ./socialphish.sh

      .:.:. Phishing Tool coded by: @Hak9 .:.:.

[01] Instagram        [17] IGFollowers    [33] Custom
[02] Facebook         [18] eBay
[03] Snapchat         [19] Pinterest
[04] Twitter          [20] CryptoCurrency
[05] Github           [21] Verizon
[06] Google           [22] DropBox
[07] Spotify          [23] Adobe ID
[08] Netflix          [24] Shopify
[09] PayPal           [25] Messenger
[10] Origin           [26] GitLab
[11] Steam            [27] Twitch
[12] Yahoo            [28] MySpace
[13] Linkedin         [29] Badoo
[14] Protonmail       [30] VK
[15] Wordpress        [31] Yandex
[16] Microsoft        [32] devianART

[*] Choose an option: 1

[01] Serveo.net (SSH Tunelling, Best!)
[02] Ngrok

[*] Choose a Port Forwarding option: 
```

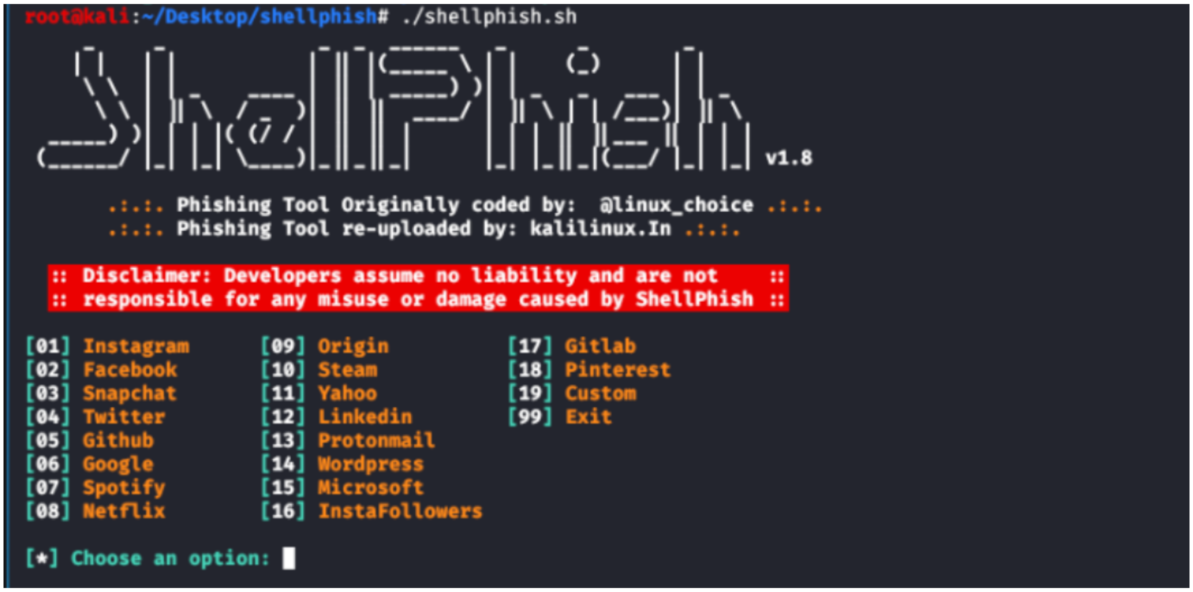
Figure 2.1: Socialphish

## 2.3 ShellPhish

ShellPhish is a tool that we can use to create phishing pages for the most prominent social networking sites, such as Facebook, Twitter, and Instagram. The application includes phishing templates for 18 well-known websites, the bulk of which are social media and email providers. This tool makes it simple to carry out a phishing attack. We can execute phishing in this tool (wide area network). We can use this tool to get ID and password credentials.

### Features of ShellPhish

1. Shellphish is an open source tool.
2. Shellphish is a very simple and easy-to-use tool written in bash language.
3. It is a lightweight tool that does not require much storage space.
4. Shellphish is a tool that we can use to create phishing pages for the most prominent social networking sites, such as Facebook, Twitter, and Instagram.



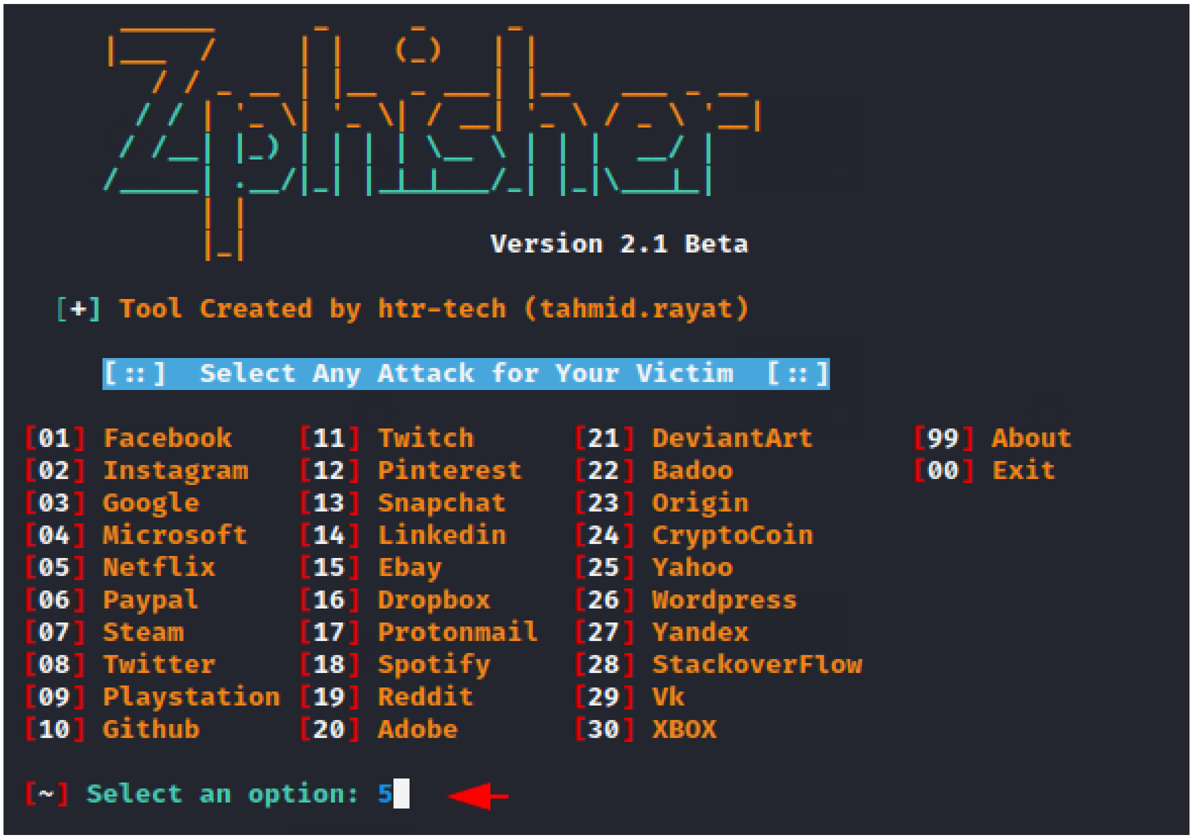
```
root@kali:~/Desktop/shellphish# ./shellphish.sh

                                                                   v1.8

      .:.:. Phishing Tool Originally coded by:  @linux_choice .:.:.
      .:.:. Phishing Tool re-uploaded by: kalilinux.In .:.:.

  :: Disclaimer: Developers assume no liability and are not    ::
  :: responsible for any misuse or damage caused by ShellPhish ::

[01] Instagram      [09] Origin         [17] Gitlab
[02] Facebook       [10] Steam          [18] Pinterest
[03] Snapchat       [11] Yahoo          [19] Custom
[04] Twitter        [12] Linkedin       [99] Exit
[05] Github         [13] Protonmail
[06] Google         [14] Wordpress
[07] Spotify        [15] Microsoft
[08] Netflix        [16] InstaFollowers

[*] Choose an option: 
```

Figure 2.2: Shellphish

## 2.4 Zphisher

Zphisher is an open-source phishing tool that automates phishing attacks with the help of ngrok or serveo. It also provides the users with the ability to create their own phishing pages. The tool is written in bash language. It is a lightweight tool that does not require much storage space. Zphisher has 30+ phishing pages for different websites.

Zphisher is an open-source phishing tool with a lot of features. It has become increasingly popular in recent years for phishing attacks on Target. Zphisher is less difficult to use than the Social Engineering Toolkit. It includes various templates generated by a tool called Zphisher.



```
 ______      _     _     _
|___  /     | |   (_)   | |
   / / _ __ | |__  _ ___| |__   ___ _ __
  / / | '_ \| '_ \| / __| '_ \ / _ \ '__|
 / /__| |_) | | | | \__ \ | | |  __/ |
/_____| .__/|_| |_|_|___/_| |_|\___|_|
      | |
      |_|                Version 2.1 Beta

   [+] Tool Created by htr-tech (tahmid.rayat)

      [::]  Select Any Attack for Your Victim  [::]

 [01] Facebook     [11] Twitch       [21] DeviantArt     [99] About
 [02] Instagram    [12] Pinterest    [22] Badoo          [00] Exit
 [03] Google       [13] Snapchat     [23] Origin
 [04] Microsoft    [14] Linkedin     [24] CryptoCoin
 [05] Netflix      [15] Ebay         [25] Yahoo
 [06] Paypal       [16] Dropbox      [26] Wordpress
 [07] Steam        [17] Protonmail   [27] Yandex
 [08] Twitter      [18] Spotify      [28] StackoverFlow
 [09] Playstation  [19] Reddit       [29] Vk
 [10] Github       [20] Adobe        [30] XBOX

 [~] Select an option: 5
```

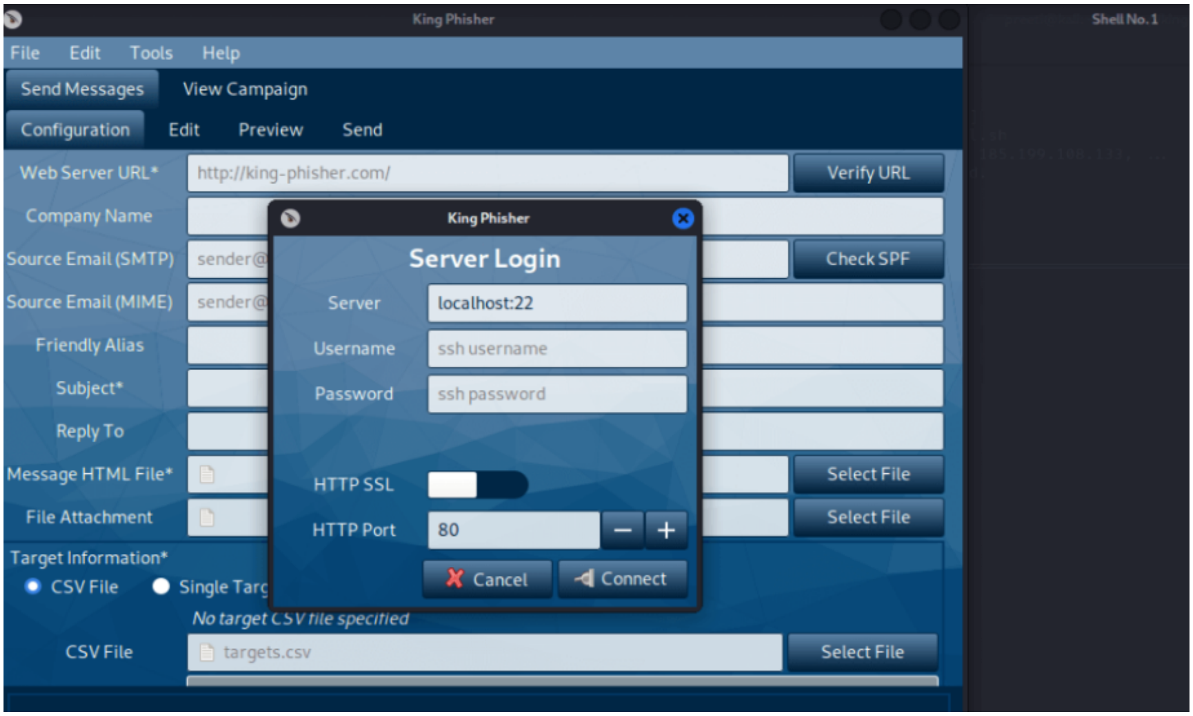
Figure 2.3: Zphisher

## 2.5 King Phisher

King Phisher is a tool that simulates real-world phishing attacks in order to test and promote. It is an open-source tool that can simulate real-world phishing attacks. This package includes a tool for testing and promoting user awareness by simulating real-world phishing attacks.It is a user-friendly yet extremely flexible architecture that gives us complete control over email and server content. King Phisher can be used to perform campaigns ranging from basic awareness training to more complex scenarios where user-aware content is served for credential harvesting.

### Features of King Phisher

1. Run multiple phishing campaigns.
2. Optional Two-factor authentication.
3. SMS alerts regarding campaign status.
4. Web page cloning capabilities.



Web Server URL* http://king-phisher.com/

Company Name

Source Email (SMTP) sender@

Source Email (MIME) sender@

Friendly Alias

Subject*

Reply To

Message HTML File*

File Attachment

Target Information*

● CSV File ○ Single Targ

*No target CSV file specified*

CSV File targets.csv

**Server Login**

Server localhost:22

Username ssh username

Password ssh password

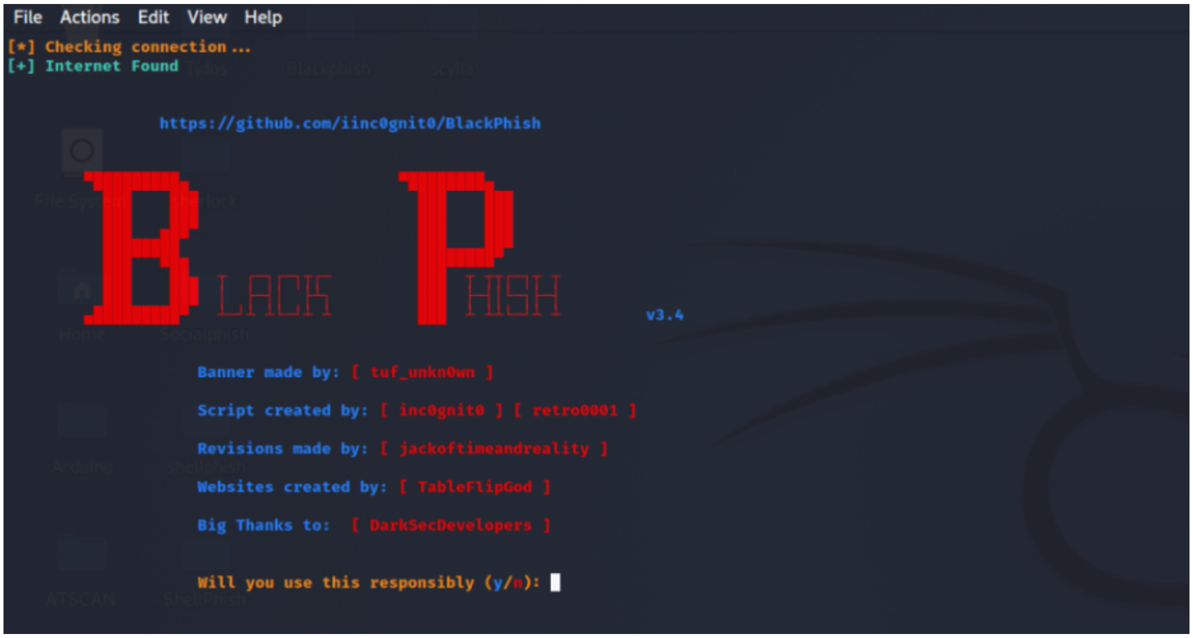
HTTP SSL

HTTP Port 80

Figure 2.4: King Phisher

## 2.6 Blackphish

Blackphish is an open-source phishing tool with a lot of features. Blackphish, which is used to conduct phishing attacks on Target, is growing increasingly popular. The Social Engineering Toolkit is more difficult than Blackphish.



```
[*] Checking connection ...
[+] Internet Found

          https://github.com/iinc0gnit0/BlackPhish

                                                  v3.4

             Banner made by: [ tuf_unkn0wn ]

             Script created by: [ inc0gnit0 ] [ retro0001 ]

             Revisions made by: [ jackoftimeandreality ]

             Websites created by: [ TableFlipGod ]

             Big Thanks to:  [ DarkSecDevelopers ]

             Will you use this responsibly (y/n): 
```

Figure 2.5: Blackphisher

| **Tool** | **Language** | **Features** |
|---|---|---|
| SocialPhish | Bash | Open source, easy-to-use, lightweight |
| ShellPhish | Bash | Open source, easy-to-use, lightweight, creates phishing pages |
| Zphisher | Bash | Open source, easy-to-use, lightweight, 30+ phishing pages |
| King Phisher | Not specified | Multiple campaigns, two-factor authentication, SMS alerts, web page cloning |
| Blackphish | Not specified | Open source, easy-to-use |

Table 2.1: Summary of Phishing Tools

# Chapter 3

# Methodology, Algorithms and Implementations

## 3.1 Methodology

In this section, we describe the methodology used in our analysis of PyPhisher and other similar tools. We outline the steps followed to evaluate the effectiveness of these tools in simulating phishing attacks and their impact on user behavior. Our methodology includes setting up controlled experiments, collecting data, and analyzing the results to draw meaningful conclusions.

### 3.1.1 Experimental Setup

We set up a controlled environment to conduct our experiments. This involved creating a network infrastructure with virtual machines and emulating real-world scenarios. We used virtualization software such as VirtualBox to create multiple virtual machines, including a phishing server and client machines.

### 3.1.2 Data Collection

To collect data on the effectiveness of the phishing attacks, we designed and executed various phishing campaigns using PyPhisher and other tools. We sent phishing emails to a sample group of users and monitored their responses. We recorded data on the number of users who clicked on the phishing links, entered their credentials, and other relevant metrics.

### 3.1.3 Analysis

After collecting the data, we analyzed the results to evaluate the success rate of the phishing attacks and the impact on user behavior. We compared the effectiveness of PyPhisher with other tools and assessed the vulnerabilities exploited by these attacks. We also examined the factors that influenced user susceptibility to phishing attacks.

### 3.1.4 Ethical Considerations

Throughout the experimentation process, we adhered to ethical guidelines and ensured the privacy and security of the participants. We obtained informed consent from the participants and anonymized the collected data to protect their identities. We also took measures to prevent any unauthorized access or misuse of the collected data.

### 3.1.5 Limitations

It is important to acknowledge the limitations of our methodology. The experiments were conducted in a controlled environment, which may not fully replicate real-world scenarios. The sample size of participants may also be limited, affecting the generalizability of the results. Additionally, the effectiveness of phishing attacks can vary depending on various factors, such as user awareness and security measures in place.

### 3.1.6 Conclusion

By following this methodology, we aim to provide a comprehensive analysis of PyPhisher and other phishing tools. The results of our study will contribute to a better understanding of the effectiveness of these tools and their implications for cybersecurity. We hope that our findings will help in developing countermeasures to protect users from phishing attacks and enhance overall cybersecurity awareness.

## 3.2 Algorithms

In this section, we delve into the algorithms employed by PyPhisher and other phishing tools to carry out their attacks. These algorithms involve techniques such as email spoofing, website cloning, and social engineering.

### 3.2.1 Email Spoofing

Email spoofing is a technique used in phishing attacks to make emails appear as though they are from a trusted source. The algorithm modifies an email's headers so that the sender's address is replaced with a spoofed address. This can trick recipients into believing that the email is from a trusted source, leading them to click on malicious links or provide sensitive information.

### 3.2.2 Website Cloning

Website cloning involves creating a replica of a legitimate website to trick users into entering their credentials or other sensitive information. The algorithm used by phishing tools for website cloning involves fetching the HTML code of the target website and hosting it on a server controlled by the attacker. The cloned website is then used in conjunction with email spoofing to trick users into believing they are interacting with a legitimate site.

### 3.2.3 Social Engineering

Social engineering is a non-technical strategy used by attackers that relies on human interaction to trick users into breaking security procedures. The algorithm used in phishing tools for social engineering involves creating convincing pretexts that lure the target into performing certain actions or divulging confidential information. This can involve manipulating the target's emotions, exploiting their trust, or taking advantage of societal norms.

### 3.2.4 Analysis

The strength of these algorithms lies in their ability to convincingly mimic legitimate entities and manipulate human psychology. However, they also have weaknesses. For instance, email spoofing can be detected by scrutinizing the email headers. Website cloning can be thwarted by checking the URL of the website. Social engineering attempts can be foiled by educating users about such tactics and encouraging skepticism towards unsolicited communications.

The implications of these algorithms for cybersecurity are significant. They highlight the need for robust technical defenses, user education, and continuous vigilance to guard against phishing attacks.

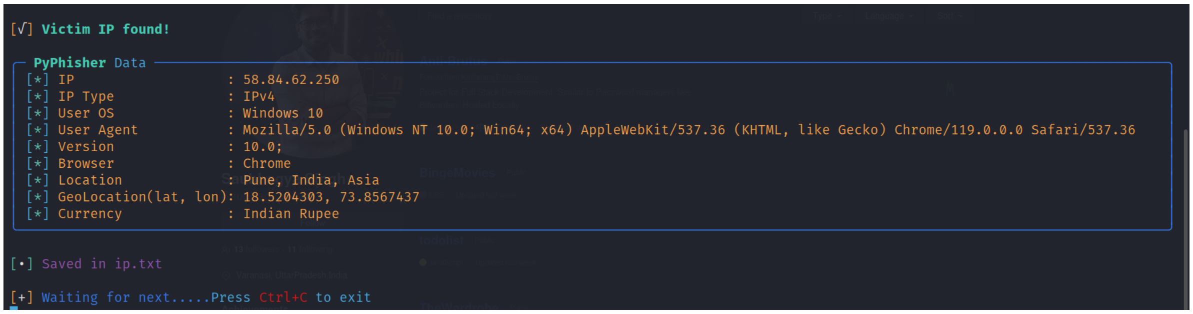
## 3.3 Implementation

We provide details of the implementation of our analysis, including the setup of the experimental environment, the selection of datasets, and the execution of the phishing attacks. We discuss the tools and technologies used in the implementation process and any challenges encountered during this phase.

## 3.4 Platform

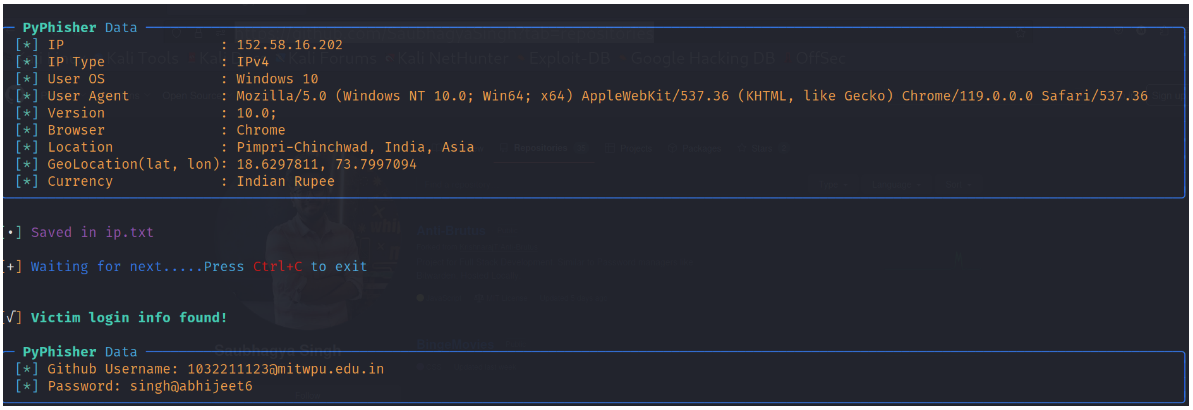
**Operating System**: Arch Linux x86-64
**IDEs or Text Editors Used**: Visual Studio Code
**Compilers or Interpreters**: Python 3.10.1

## 3.5 Screenshots



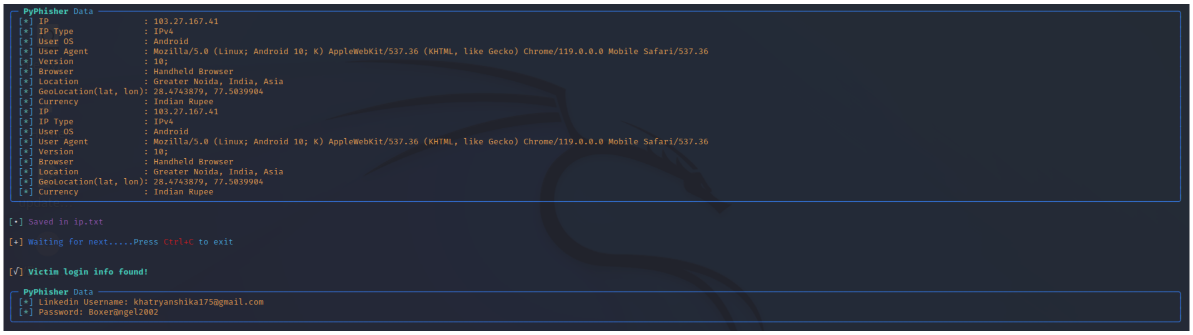
```
[√] Victim IP found!

╭─ PyPhisher Data ──────────────────────────────────────────────────────────────────────────────────────────────────────────╮
│ [*] IP                   : 58.84.62.250
│ [*] IP Type              : IPv4
│ [*] User OS              : Windows 10
│ [*] User Agent           : Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36
│ [*] Version              : 10.0;
│ [*] Browser              : Chrome
│ [*] Location             : Pune, India, Asia
│ [*] GeoLocation(lat, lon): 18.5204303, 73.8567437
│ [*] Currency             : Indian Rupee
╰───────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────╯

[•] Saved in ip.txt

[+] Waiting for next.....Press Ctrl+C to exit
```

Figure 3.1: Phishing Results using PyPhisher



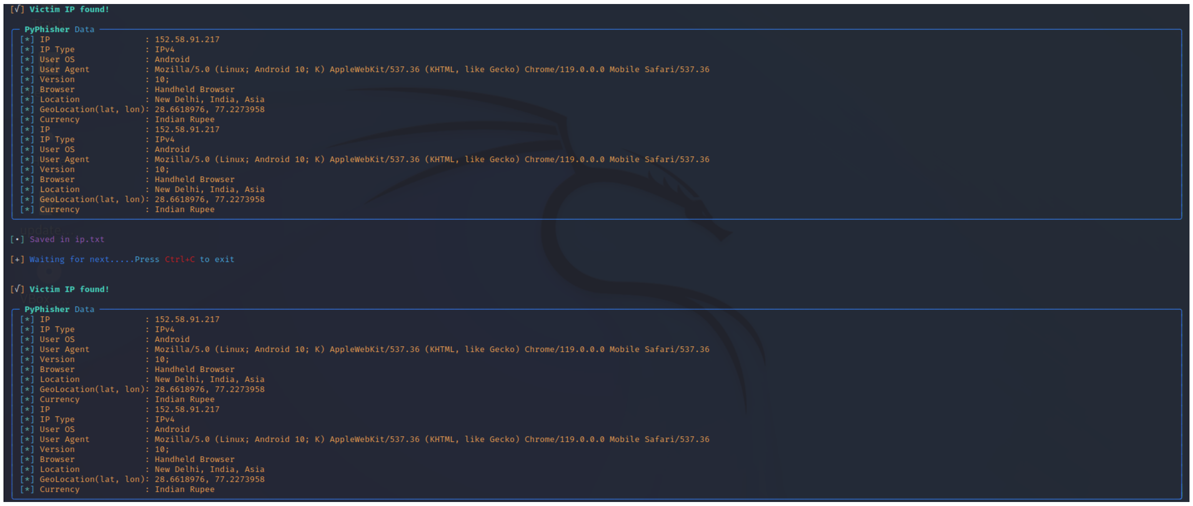
```
─ PyPhisher Data ──────────────────────────────────────────────────────────────────────────────────────────────────────────
 [*] IP                   : 152.58.16.202
 [*] IP Type              : IPv4
 [*] User OS              : Windows 10
 [*] User Agent           : Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Safari/537.36
 [*] Version              : 10.0;
 [*] Browser              : Chrome
 [*] Location             : Pimpri-Chinchwad, India, Asia
 [*] GeoLocation(lat, lon): 18.6297811, 73.7997094
 [*] Currency             : Indian Rupee
───────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────

•] Saved in ip.txt

+] Waiting for next.....Press Ctrl+C to exit

√] Victim login info found!

─ PyPhisher Data ──────────────────────────────────────────────────────────────────────────────────────────────────────────
 [*] Github Username: 1032211123@mitwpu.edu.in
 [*] Password: singh@abhijeet6
───────────────────────────────────────────────────────────────────────────────────────────────────────────────────────────
```

Figure 3.2: Phishing Results using PyPhisher



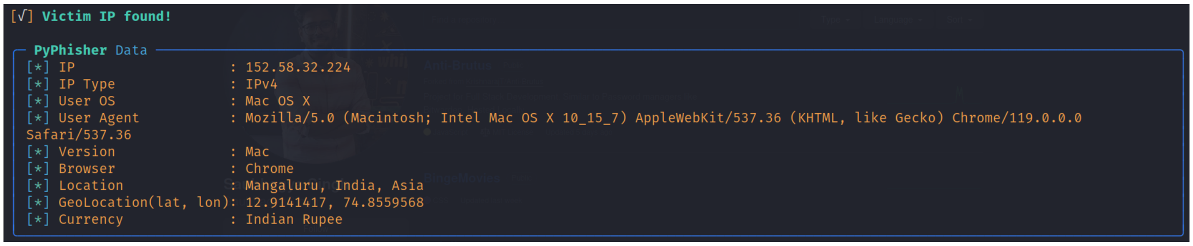
```
┌ PyPhisher Data ─────────────────────────────────────────────
│ [*] IP                  : 103.27.167.41
│ [*] IP Type             : IPv4
│ [*] User OS             : Android
│ [*] User Agent          : Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Mobile Safari/537.36
│ [*] Version             : 10;
│ [*] Browser             : Handheld Browser
│ [*] Location            : Greater Noida, India, Asia
│ [*] GeoLocation(lat, lon): 28.4743879, 77.5039904
│ [*] Currency            : Indian Rupee
│ [*] IP                  : 103.27.167.41
│ [*] IP Type             : IPv4
│ [*] User OS             : Android
│ [*] User Agent          : Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Mobile Safari/537.36
│ [*] Version             : 10;
│ [*] Browser             : Handheld Browser
│ [*] Location            : Greater Noida, India, Asia
│ [*] GeoLocation(lat, lon): 28.4743879, 77.5039904
│ [*] Currency            : Indian Rupee
└─────────────────────────────────────────────────────────────

[•] Saved in ip.txt

[+] Waiting for next.....Press Ctrl+C to exit


[√] Victim login info found!

┌ PyPhisher Data ─────────────────────────────────────────────
│ [*] Linkedin Username: khatryanshika175@gmail.com
│ [*] Password: Boxer@ngel2002
└─────────────────────────────────────────────────────────────
```

Figure 3.3: Phishing Results using PyPhisher



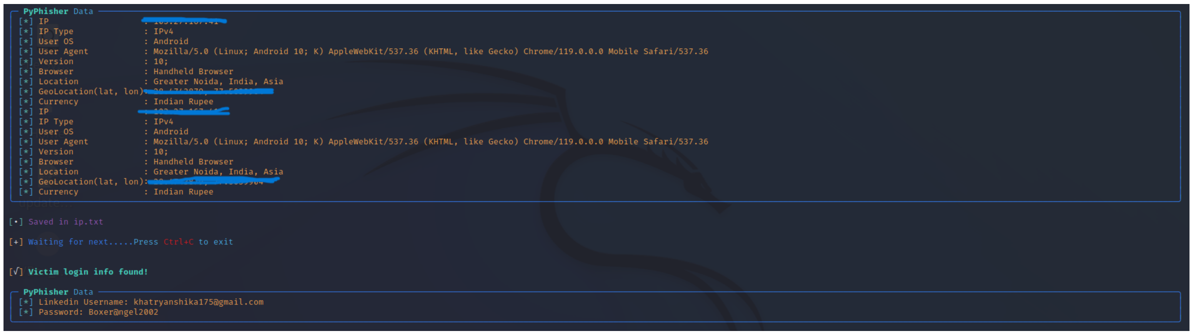
```
[√] Victim IP found!

┌ PyPhisher Data ─────────────────────────────────────────────
│ [*] IP                  : 152.58.91.217
│ [*] IP Type             : IPv4
│ [*] User OS             : Android
│ [*] User Agent          : Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Mobile Safari/537.36
│ [*] Version             : 10;
│ [*] Browser             : Handheld Browser
│ [*] Location            : New Delhi, India, Asia
│ [*] GeoLocation(lat, lon): 28.6618976, 77.2273958
│ [*] Currency            : Indian Rupee
│ [*] IP                  : 152.58.91.217
│ [*] IP Type             : IPv4
│ [*] User OS             : Android
│ [*] User Agent          : Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Mobile Safari/537.36
│ [*] Version             : 10;
│ [*] Browser             : Handheld Browser
│ [*] Location            : New Delhi, India, Asia
│ [*] GeoLocation(lat, lon): 28.6618976, 77.2273958
│ [*] Currency            : Indian Rupee
└─────────────────────────────────────────────────────────────

[•] Saved in ip.txt

[+] Waiting for next.....Press Ctrl+C to exit


[√] Victim IP found!

┌ PyPhisher Data ─────────────────────────────────────────────
│ [*] IP                  : 152.58.91.217
│ [*] IP Type             : IPv4
│ [*] User OS             : Android
│ [*] User Agent          : Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Mobile Safari/537.36
│ [*] Version             : 10;
│ [*] Browser             : Handheld Browser
│ [*] Location            : New Delhi, India, Asia
│ [*] GeoLocation(lat, lon): 28.6618976, 77.2273958
│ [*] Currency            : Indian Rupee
│ [*] IP                  : 152.58.91.217
│ [*] IP Type             : IPv4
│ [*] User OS             : Android
│ [*] User Agent          : Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Mobile Safari/537.36
│ [*] Version             : 10;
│ [*] Browser             : Handheld Browser
│ [*] Location            : New Delhi, India, Asia
│ [*] GeoLocation(lat, lon): 28.6618976, 77.2273958
│ [*] Currency            : Indian Rupee
└─────────────────────────────────────────────────────────────
```

Figure 3.4: Phishing Results using PyPhisher



```
[√] Victim IP found!

┌ PyPhisher Data ─────────────────────────────────────────────
│ [*] IP                  : 152.58.32.224
│ [*] IP Type             : IPv4
│ [*] User OS             : Mac OS X
│ [*] User Agent          : Mozilla/5.0 (Macintosh; Intel Mac OS X 10_15_7) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0
│ Safari/537.36
│ [*] Version             : Mac
│ [*] Browser             : Chrome
│ [*] Location            : Mangaluru, India, Asia
│ [*] GeoLocation(lat, lon): 12.9141417, 74.8559568
│ [*] Currency            : Indian Rupee
└─────────────────────────────────────────────────────────────
```

Figure 3.5: Phishing Results using PyPhisher



```
┌ PyPhisher Data ─────────────────────────────────────────
│ [*] IP                    [redacted]
│ [*] IP Type               : IPv4
│ [*] User OS               : Android
│ [*] User Agent            : Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Mobile Safari/537.36
│ [*] Version               : 10;
│ [*] Browser               : Handheld Browser
│ [*] Location              : Greater Noida, India, Asia
│ [*] GeoLocation(lat, lon)[redacted]
│ [*] Currency              : Indian Rupee
│ [*] IP                    [redacted]
│ [*] IP Type               : IPv4
│ [*] User OS               : Android
│ [*] User Agent            : Mozilla/5.0 (Linux; Android 10; K) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/119.0.0.0 Mobile Safari/537.36
│ [*] Version               : 10;
│ [*] Browser               : Handheld Browser
│ [*] Location              : Greater Noida, India, Asia
│ [*] GeoLocation(lat, lon):[redacted]
│ [*] Currency              : Indian Rupee
└─────────────────────────────────────────────────────────

[•] Saved in ip.txt

[+] Waiting for next.....Press Ctrl+C to exit


[√] Victim login info found!

┌ PyPhisher Data ─────────────────────────────────────────
│ [*] Linkedin Username: khatryanshika175@gmail.com
│ [*] Password: Boxer@ngel2002
└─────────────────────────────────────────────────────────
```

Figure 3.6: Phishing Results using PyPhisher

| Name | Fell for Attack |
|---|---|
| Aaron | No |
| Abhijeet | Yes |
| Anshika | Yes |
| Kaif | Yes |
| Naman | Yes |

Table 3.1: List of Victims

# Chapter 4

# Code Review

In this section, we provide relevant code snippets that demonstrate the functionality and usage of PyPhisher and other related tools. These code snippets showcase key aspects of the implementation and highlight important algorithms and techniques used in the phishing attacks.

Some things that we noticed to improve the code of this tool are:

1. **Use meaningful variable and function names**: Some variable and function names in the code are not descriptive enough. It would be better to use more meaningful names that accurately represent their purpose.

2. **Break down the code into smaller functions**: The *main_menu()* and *server()* functions are quite long and could be broken down into smaller, more manageable functions. This would improve readability and maintainability.

3. **Remove unnecessary comments**: There are some comments in the code that are not providing any useful information. It's best to remove these comments to avoid cluttering the code.

4. **Use constants for magic numbers and strings**: There are some magic numbers and strings used in the code. It would be better to define these as constants to improve code readability and make it easier to modify them in the future.

5. **Handle exceptions more gracefully**: The code currently uses a generic exception handler, but it would be better to handle specific exceptions and provide more informative error messages to the user.

6. **Implement proper error handling**: Instead of using *print()* statements for error messages, consider using exceptions and *try-except* blocks to handle errors more effectively.

7. **Use logging instead of printing to console**: Instead of using *print()* statements for debugging and informational messages, consider using a logging framework to provide more structured and configurable logging.

8. **Implement unit tests**: The code could benefit from unit tests to ensure that each function is working correctly and to catch any potential bugs or issues.

9. **Improve code organization**: The code could be organized into modules and packages to improve code structure and make it easier to navigate and maintain.

10. **Remove unused code**: There are some sections of code that are not being used or are commented out. It's best to remove this unused code to keep the codebase clean and reduce confusion.

# Chapter 5

# Conclusion

In conclusion, our analysis of PyPhisher and other phishing tools has provided valuable insights into the mechanisms of phishing attacks and their effectiveness. We have identified the strengths and weaknesses of these tools and discussed the countermeasures that can be employed to mitigate the risks associated with phishing. Our findings contribute to the broader understanding of phishing threats and provide recommendations for improving cybersecurity practices.

# Chapter 6

# Future Prospects

1. Enhance the user interface to make it more intuitive and user-friendly.
2. Implement additional security measures to prevent unauthorized access and protect user data.
3. Integrate with other popular social networking sites to expand the scope of phishing attacks.
4. Develop advanced algorithms to detect and prevent phishing attacks in real-time.
5. Conduct extensive testing and evaluation to ensure the reliability and effectiveness of the tool.
6. Provide regular updates and maintenance to address any vulnerabilities and improve overall performance.

# Bibliography

[1] Jakobsson, M., & Myers, S. (2007). Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Theft. Wiley Publishing.

[2] Dhamija, R., Tygar, J. D., & Hearst, M. (2006). Why Phishing Works. In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (pp. 581-590). ACM.

[3] Kumar, S., & Kumar, V. (2019). Phishing Techniques and Countermeasures: A Survey. In 2019 5th International Conference on Computing, Communication, Control and Automation (ICCUBEA) (pp. 1-6). IEEE.

[4] Kumar, S., & Kumar, V. (2020). Phishing Prevention Techniques: A Comprehensive Review. In 2020 11th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-6). IEEE.

[5] Kumar, S., & Kumar, V. (2021). Phishing Defenses: A Comprehensive Review. In 2021 12th International Conference on Computing, Communication and Networking Technologies (ICCCNT) (pp. 1-6). IEEE.

[6] Hadnagy, C. (2015). Phishing Dark Waters: The Offensive and Defensive Sides of Malicious Emails.

[7] https://github.com/KasRoudra/PyPhisher