Brycen Fulscher

Kaleb Lingard

Group 5

IT 359 002

**GoPhish Project Code Documentation**

[**IT359 GoPhish Project Github Link**](https://github.com/bfulscher/IT359Project)

**GoPhish Install**

wget https://github.com/gophish/gophish/releases/latest/download/gophish-linux-64bit.zip

sudo apt update && sudo apt install unzip -y

unzip gophish-linux-64.zip -d gophish

cd gophish

chmod +x gophish

sudo ./gophish

From here, the gophish command will give you a temporary password and a URL (127.0.0.1:3333) to access the Gophish admin tab from

The admin tab is where we can build out our campaign (landing pages, email templates, mail accounts, etc.)

**MailHog Install**

wget https://github.com/mailhog/MailHog/releases/latest/download/v1.0.1/MailHog\_linux\_amd64

chmod +x MailHog\_linux\_amd64

sudo mv MailHog\_linux\_amd64 /usr/local/bin/mailhog

mailhog

Mailhog will run indefinitely from the command line while you want the service open

When running the command, it will give you a URL (0.0.0.0:8025) and password to access MailHog from.

**login.html**

The index.html file is a simulated login page created for our GoPhish campaign. It mimics a legitimate login interface and is designed to collect user-entered credentials (username and password). When submitted, the form sends the data via a POST request to the GoPhish listener at http://10.10.10.11/submit, allowing the campaign to track and log responses for training and analysis purposes.

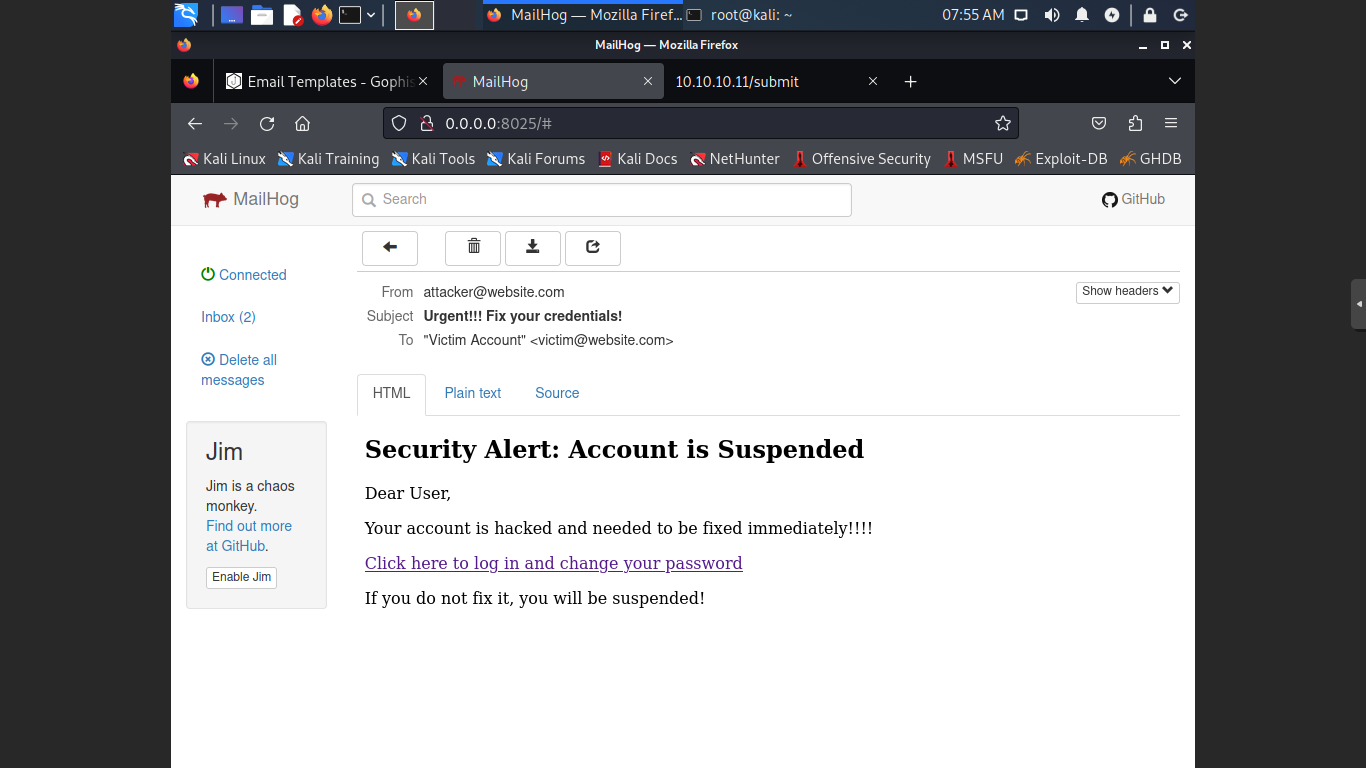
**login.php**

This PHP script processes login submissions by connecting to the login\_db MySQL database. When a user submits their username and password, the script checks for a matching record in the users table. If a match is found, it displays a success message. If it does not, it notifies the user of an invalid login. This script is intended to have very simple functionality and lacks security features like input sanitization, making it unsuitable for real-world deployment, but usable for our testing.

**emailtemplate.html**

This HTML file is a phishing email template used in the project to simulate a security alert. It contains a message urging the recipient to click a fake login link that leads to a phishing landing page (login.html). The template also includes a tracking pixel ({{.TrackingURL}}) to monitor when the email is opened, and an optional tracking tag ({{.Tracker}}) for compatibility with different GoPhish configurations. The goal of the email is to simulate a real attack, but not be too convincing as to commit an actual phishing attack.

Screenshot of the template from the view of email receiver:

****

**login\_db\_backup.sql**

This SQL file defines and populates a MySQL database table named users for the login\_db database. It first drops the table if it exists, then creates the users table with three fields: id (primary key), username, and password. It inserts two example user records with identical credentials (admin, password123). The dump includes setup and teardown commands for consistent import/export across environments. This setup is used to validate logins submitted through the phishing form.