

One Less Phish To Bite: Phishing For Phish

Jonah Grier
Software Vulnerabilities, and Security
Northeastern University Seattle
Seattle, USA
grier.jo@northeastern.edu

Golbahar Mohseni
Software Vulnerabilities, and Security
Northeastern University Seattle
Seattle, USA
mohseni.g@northeastern.edu

Abstract—PhishingForPhish is the task of providing a safer and less fraudulent experience interacting with government websites. Originally, the World Wide Web served as an exchange of academic information through documents using hypertext markup language, but, as it expanded, the usages have significantly grown. Now, the internet serves everyone in a different fashion and is estimated that there are over 1.5 billion websites on the internet today [1]. PhishingForPhish application aims to protect users from phishing attempts on governmental websites within the United States. Though a significantly smaller number of “.gov” websites exist today, it is a starting ground for a new tactic in securing internet browsing.

Keywords—PFP (PhishingForPhish), RDS (Relational Database Server), TLD (Top Level Domain .gov, .com, etc.)

I. INTRODUCTION

In a day and age where applications go to extreme measures to verify the credibility of the individual behind the screen, PFP (PhishingForPhish) takes a different approach. While increasing security of applications reduces risks of negligent behavior, it is difficult to secure platforms against forgery and replication. Instead of implementing a third-party application that ensures protective measures, PFP places trust in the users. PFP application verifies the website users are currently on is, in-fact, a verified government website. This methodology reduces an extra application layer that websites would need to apply and strengthens confidence in internet users.

II. SYSTEM ANALYSIS

A. System

PFP relies on Google Chrome’s extension service. This service is utilized for two reasons: (1) Google Chrome already has a strong userbase with the added feature that installation is brief, and (2) developing a third-party system that tracks user’s internet interactions brings its own concerns. As an extension toolbar application, PFP is able to apply Chrome’s extensive security knowledge, and functionality to bring a simple, yet powerful toolbar app.

PFP is simple. When a user installs the program, they are prompted with a small icon on the top right of their toolbar. This icon allows the user to interact with the extension and check the validity of the current webpage. When a government site is loaded, PFP logs the URL into its system and processes whether it matches a whitelisted URL. If the current URL is a match nothing happens. However, if the user selects a webpage that is a government website and PFP cannot find the domain listed in its database, PFP flags the user from the icon and reports the link to be suspicious. Additionally, if the URL does not match the application performs an assessment of the current webpage and reports back links to alternate websites that are knowingly verified and of similarity.

PFP does not store any URL, or track anything about the user. Its sole responsibility is to keep a lookout for any suspicious government websites that may pose a threat.

B. Assets

PFP’s main concern is that of the user’s safety. According to the Anti-Phishing Working Group (APWG), the number of phishing attacks rose in Q3, 2019, and the use of SSL/HTTPS is no longer a good indication for safety [3]. PhishingForPhish aims to protect our userbase from the hidden cyberthreats designed to steal information and falsify identities. It is our goal to ensure all U.S. government domains have not been compromised and if so to redirect people to safety. PFP is certain that each step in protecting people against phishing attempts is a step in the right direction.

C. Adversaries

The contemporary internet system is littered with cyber criminals looking for a leg up on unsuspecting victims. Mimicking websites and copying design patterns are easily accessible by novice programmers. Nefarious actors can quickly breach privacy by building traps and stealing content in large quantities through the use of the internet. While initial thoughts might conclude a single thief in the scheme, large corporations could be to blame. If phishing scams become large enough, there could be underground sales through companies that provide website anonymity, URL boosts, or protection in exchange for monetary value. Large tech companies who dominate the internet footprint could easily provide plugins and solutions to these daily phishing problems. It begs the question as to why they continue to allow it. It is possible that anti-virus companies who provide functions for a cleaner web, purposefully resist building applications to increase sales by providing a “unique” anti-phishing feature.

D. Vulnerabilities

Computers are sought for two reasons: (1) to make their lives easier, and (2) to connect to the internet. The internet provides a virtual playground for people to store information, communicate, and compute. In 2016, Thomas Barnett Jr., Cisco director of thought Leadership, explained the idea of measuring internet traffic in terms of bytes. One zettabyte is equivalent to one sextillion bytes [4]. This is equivalent of “watching 36,000 hours of high-definition video, which, in turn, is the equivalent of streaming Netflix’s entire catalog 3177 times” (S. Pappa). In essence, the internet is just a simple network of nodes, all linked to one-another through a network of networks. The world wide web, on the other hand, is a way of accessing and sharing information through the network of nodes. The web platform was built on top of the internet. As such, the web is decentralized. While this is good in that no one-party has full control, it poses security risks.

Over the years different parties have been established to aid in the shortcoming of privacy and vulnerabilities. Companies such as Certificate Authorities, allow websites to apply for different verification levels to prove that their track record is clean, and they are who they say they are. These authorities, who are often regarded as CA's, provide strong crypto connections that internet-surfers utilize when connecting to websites. Additionally, they grant different certifications to websites to help improve legitimacy: DV (Domain Validated), OV (Organization Validated) and EV (Extended Validated). These certifications can be a good indication of website validity. However, as mentioned earlier, websites with strong certifications do not always prove to be safe. Other identifiers of phishing websites can be that of recognizing poor English grammar throughout the page, too many advertisements, or a lack of visual cues of security features.

This is only the tip of the iceberg, however. The web has always been a hotspot for creative hacks. Vulnerabilities such as injection flaws, broken authentications, XSS (Cross Site Scripting), sensitive data exposures as well as others can be fatal to websites survival.

E. Threats & Risks

PFP is designed for simplicity and ease of use. It is not a third-party system storing information, or a tracking device used to secretly follow user-based interactions. PFP maintains and utilizes a whitelisting database to type check government websites. For vulnerabilities to occur, PFP would need to undergo an SQL injection or XSS attack involving changing web URLs. Even then, the attack would need to be specialized. PFP works in the background so there are no input forms or user interaction with the extension. In layman's terms, the application does not allow a user to enter information to query what they want. Removing that user interaction allows PFP to keep the simplicity it desires and eradicates the ability for attackers to probe at its whitelisted websites.

XSS attacks, on the other hand, are when users try to execute malicious code within the website. For PFP, a user would need to spoof the current URL loaded in order to break the system. Given there are no input queries, and the plugin runs in the background, the likelihood of XSS attacks are slim.

PFPs largest asset is the whitelist, however, fear of exposure is not a large concern.

III. SYSTEM ADD-ON

By system of choice, PhishingForPhish can be regarded as a system add-on. While the application does not redesign the way users query information, it does change the probability of stumbling upon an illegitimate website. PFP provides Chrome an internal layering counter-phishing measure.

Expected benefits include increased user safety, reduction in the replication of government domains, and providing a steppingstone for strategies against phishing.

Drawbacks to this solution occur when thinking about the internet at large. Keeping a whitelist of verified internet domains would not only be incredibly large, but URLs may

also expire. It is not uncommon for a legitimate website to sink out of existence, being bought by another party. Domain redistribution could wreak havoc to users if the only countermeasure were tracking URLs. To mitigate this, future versions of PFP would need to perform regular inspection of current whitelisted links.

IV. ACCOMPLISHMENTS

A. Overview

PFP is in beta stage. All diagrams, depictions, and forward thinking have been accepted and are being implemented. As of today, the application has established a whitelist of 348 unique government websites, along with "tags" (words) that describe each link. These tags act as helpful indicators when trying to recommend other websites. To summarize, PFP uses a three-stage system. First, PFP registers the URL from the users current tab. It performs a simple SQL query to an Amazon relational database server checking for matches. If there is a match, step two is as easy as returning no response at all. However, if the URL does not match, PFP utilizes an internal web-scraping function to retain text on the unmatched site, cleans the text scraped, and performs a frequency analysis on its set of words. In phase three, PFP takes the top ten words in the frequency analysis and compares them to the tags corresponding to verified government domains in the whitelist. This check allows PFP to accurately return websites that have a similarity in tags and are known to be safe.

B. Backend

The backend system is reliant on an Amazon RDS located in Oregon. Having a nearby storage center allows PFP to test speed in a low latency environment. Inside the RDS, information is kept simple. There are two tables that represent a domain: (1) Website, and (2) Tag. Each *Website* table consists of a websiteID, protocol, domain, path, parameters and fragments. This allows URLs of all sizes to be inserted without the need to condense them.

TABLE I. AWS RDS "WEBSITE" TABLE

websiteID	protocol	domainName	pathName	parameters	fragments
1	http	www.abilityone.gov	/	None	None
2	http	www.access-board.gov	/	None	None
3	http	www.grants.gov	/	None	None
4	http	www.acl.gov	/	None	None
5	http	www.acf.hhs.gov	/	None	None

Fig. 1. Mysql Backend Representation

The *Tag* table looks very similar in that it contains a websiteID. Each Tag has a one-to-many relationship with the *Website* table. Using this relationship allows PFP to chain link tags to form an array of words per website. Additionally, to maintain storage size, there is a column for frequency. This column reduces the volume of references for terms to a descriptor and prevalence number. This way, instead of having seventy-six occurrences of the same word, there is only one record with the frequency associated with it.

TABLE II. AWS RDS “TAG” TABLE

tagID	websiteID	tag_text	freq_count
47047	199	cancer	139
4862	21	arc	76
54753	228	nirb	66
14458	59	helsinki	64
35828	151	fsis	64

Fig. 2. Mysql Backend Representation

C. Natural Language Processing

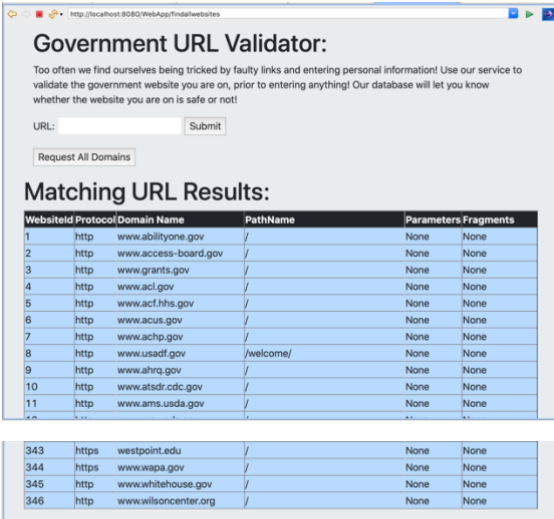
During PFP’s infancy stage, PFP relied heavily on natural language processing. NLP allows the application to “suggest” users a verified website from its whitelist as an alternative to the unverified URL currently used. Through the use of BeautifulSoup (Python 3.8 import), PFP is able to quickly parse through texts on a webpage. After collection of all texts, PFP then utilizes numerous functions to clean and analyze frequencies. This process, however, is very tedious. Using BeautifulSoup to scrape sites requires changing VPN’s often to mask sending too many requests engendering suspicious activity.

Unfortunately, to remain as a dynamic web application and keep scraping speeds constant, PFP has employed the use of Luminati.io webscraping. This allows PFP to apply Luminati’s proxy and easily integrate their API.

D. Java Database Connectivity (JDBC)

Currently, PFP uses the java database connectivity interface to allow clients access into the RDS. The JDBC manages sound connections asynchronously and can quickly perform execution of complex queries. Along with utilizing a JDBC, PFP uses Apache Tomcat server 9.0 to test its functionalities on localhost. By using Tomcat, PFP can sort out kinks in a controlled environment prior to releasing publicly.

TABLE III. JDBC “REQUEST ALL DOMAINS”



WebsiteID	Protocol	Domain Name	PathName	Parameters	Fragments
1	http	www.abilityone.gov	/	None	None
2	http	www.access-board.gov	/	None	None
3	http	www.grants.gov	/	None	None
4	http	www.acl.gov	/	None	None
5	http	www.acl.hhs.gov	/	None	None
6	http	www.acus.gov	/	None	None
7	http	www.achp.gov	/	None	None
8	http	www.usa11.gov	/welcome/	None	None
9	http	www.ahrq.gov	/	None	None
10	http	www.atsdr.cdc.gov	/	None	None
11	http	www.ams.usda.gov	/	None	None
343	https	westpoint.edu	/	None	None
344	https	www.wapa.gov	/	None	None
345	http	www.whitehouse.gov	/	None	None
346	http	www.wilsoncenter.org	/	None	None

Fig. 3. Testing all domains

E. Chrome Extension & CORS

Unfortunately, PFP will have to hold off the release of its Google Chrome extension. To protect against web vulnerabilities developers enacted a policy called CORS: *Cross Origin Resource Sharing*. CORS blocks HTTP requests and responses unless they are of the same origin [5]. In other words, if a website sends a GET request to an entirely different website, it will be blocked due to the two not sharing the same origin. Unless a website sends a GET request to something within their domain, it will end up failing. This policy breaks the original rationale of using PFP to query a tabs URL and send requests to the RDS. Due to CORS, the application is unable to be a stable Chrome extension.

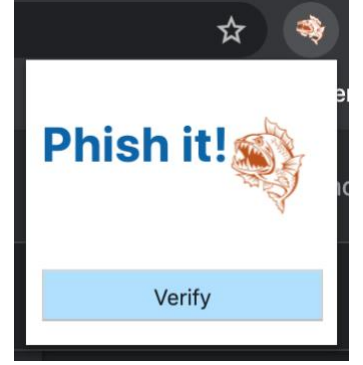


Fig. 4. Chrome extension

Remedying this issue, PFP will need to discover an alternate solution that avoids breaking CORS policy. It was discovered that AWS may have an RDS setting to allow certain domains access. PFP will need to correctly figure out how to allow not only a domain, but a full web-application access to the RDS.

Alternatively, steps that can be taken to achieve this may include trying another database provider such as Google Cloud SQL. Chrome extensions may detect Google cloud database software as in-house, and not fault to breaking CORS. If this is the case, PFP will switch indefinitely.

V. FUTURE IMPROVEMENTS

A. User Specific Whitelists

PFP would like to provide users the ability to manage and store their own personal whitelists. This allows users to feel a stronger sense of ownership over their personal information.

B. Cryptography

To provide users with the ability to manage personal whitelists, PFP would need to build an accompanying secure login feature. Maintaining user confidence is a high priority therefore a one-way encryption algorithm would be used.

C. Stronger WebScraping Technologies

Apart from all the database connectivity and querying, the belly of the beast comes from webscraping. Currently, PFP uses BeautifulSoup to parse through texts combing for domain specific descriptor words. As stated in the section: *Overview of Accomplishments*, this allows PFP to provide cognizant recommendations when a user’s tab does not match one found in the database. If the application could also process images through computer vision and pattern recognition it might allow PFP to generate stronger descriptor words about websites. Doing so could improve PFPs recommendation performance.

D. Additional Relational Database Servers

Currently, the cloud storage PFP utilizes is located conveniently in Oregon. If the server was located elsewhere, such as the East Coast, performance times for querying information may experience slower speeds. This is due to the extra distance the connection must travel before sending a response back. To improve this, purchasing additional web servers located around the country would allow users from different areas to connect to the closest one. This would not only help reduce the stress on one database but would also remove any concerns regarding Amazon experiencing outages. If the current webserver went down for any reason, PFP would fall prey until it was brought online.

E. Top Level Domain Expansion

Eventually, PFP would like to expand to include all major TLD's (Top Level Domain) in its fight against phishing.

VI. RELATED FIELD WORK

A. MetaCert: Internet Security

Most closely related to PFP is MetaCert. MetaCert is a Chrome browser extension that serves to protect against phishing and malware scams. Through URL classification and a threat intelligence database, MetaCert detects, secures, and proactively feeds users information.

B. Netcraft

Part lookup table, part protection service, Netcraft provides a broad overview of website information. This includes giving advanced metrics such as the IP address, DNS lookups, and other hard to find site specific details. Netcraft's mission is to build an anti-phishing community through self-reports to ultimately cover the entire internet.

C. Other

There are numerous other smaller anti-phishing plugins that cater to specific features.

VII. CONCLUSION

Currently, through the use of captcha's, SMS text messages, email verifications and other methods, computer systems are designed to verify individuals but the corollary of individuals querying website legitimacy is not available. PhishingForPhish resolves this issue by confirming site validity improving user's confidence.

REFERENCES

- [1] InternetLiveStats, "InternetLiveStats," 2020. [Online].
- [2] M. Lee, "Phishing Scams Cost American Businesses Half A Billion Dollars A Year," 2017. [Online].
- [3] S. Cook, "Phishing statistics and facts for 2019-2020," 2020. [Online].
- [4] S. Pappas, "How Big Is the Internet, Really?," 2016. [Online]
- [5] Codecademy, "What is CORS?,"[Online].