Detecting Phishing in a Heuristic Way (Abstract)

Lucas Torrealba A. and Javier Bustos-Jiménez
NIC Chile Research Labs, University of Chile

lucas@niclabs.cl - jbustos@niclabs.cl

Phishing is one of the most prevalent forms of cybercrime. In addition, due to COVID-19 and how it has impacted both the economy and the lifestyle of people, the cybercrime risks have increased even more [1].

This has led several researchers to investigate how to prevent and detect it. One of the techniques to detect phishing websites is use blacklists to filter them. Blacklist is efficient and simple but three problems of use it is: (1) the null defense against newly created sites (zero-day), (2) the effort to keep them updated (add domains, delete domains, etc.) properly, (3) the accuracy with which information is saved (if a single character is changed it becomes a totally different domain).

Also heuristics-based solutions have been developed as the intent to identify phishing through prediction. The features of the website are used to classify a page as phishing. There are 3 main features that have been study: (1) Non-content-based (analysis of lexical patterns in the domain, whitelist), (2) Content-based (CSS formating, HTML-JavaScript code, misspellings, etc.) and (3) Visual similarity-based. In addition, other features such as the DNS information (A, MX, NS, PTR) of the domain, ASN associated with the domain, register used to register the site, connection speed, etc. are identified. Our main goal is to predict those domains or subdomains that are phishing websites using heuristics over such features.

As an idea of a general solution we propose: Given a domain the first filter is to use a blacklist. Second, the general idea is to use a whitelist that will store information on the most queried domains (Alexa ranking) and the domains with the highest probability of carrying out some type of phishing attack (banks, retails, etc.). Using n-grams with techniques similar to those used in these [2, 3], we will generate metrics and rankings to determine if a website could be characterized as phishing. Third, to identify typos we will also use hash tables and Karp-Rabin algorithm to improve the execution time.

As mentioned, the features associated with non-content-based can be faster to process but not necessarily the most triggering. Then various features that can be extracted without increasing the execution time of our algorithm and improving its predictive capacity will also be studied. Finally use the features selected in the previous steps to find rules (cosine similarity, Jaccard, Sorensen-Dice, etc.) or create rules to predict if a domain corresponds to phishing.

References

[1] Chi Tran. “Recommendations for Ordinary Users from Mitigating Phishing and Cybercrime Risks During COVID-19 Pandemic”. In: (May 2020).

[2] Hong Zhao et al. “Malicious Domain Names Detection Algorithm Based on Lexical Analysis and Feature Quantification”. In: *IEEE Access* 7 (2019), pp. 128990–128999. DOI: 10.1109/ACCESS.2019.2940554.

[3] Hong Zhao et al. “Malicious Domain Names Detection Algorithm Based on N-Gram”. In: *J. Comput. Netw. Commun.* 2019 (Jan. 2019). ISSN: 2090-7141. DOI: 10.1155/2019/4612474. URL: https://doi.org/10.1155/2019/4612474.