PhishFingerprint: A Practical Approach for Phishing Web Page Identity Retrieval Based on Visual Cues

Abhishek Gangwar[1,2,3], E. Fidalgo[1,2], E. Alegre[1,2], V. González-Castro[1,2]
[1]Universidad de León, Spain
[2]INCIBE (Spanish National Institute of Cybersecurity), León, Spain
[3]CDAC, Mumbai, India
{agan,eduardo.fidalgo, enrique.alegre,victor.gonzalez}@unileon.es

Abstract. Phishing is an example of social engineering crime conducted to steal user data through impersonating a trusted third party. When a phishing site is detected by some user, like a bank or other financial organization, they normally report it to Law and Enforcement Agency (LEA) to take it down. In this paper, we propose a framework to maintain a repository of phishing web pages and to retrieve the identity of new reported phishing web pages. Specifically, the major contributions of this paper are twofold: first, we propose a semi-automated approach to create a non-redundant phishing web pages database. Second, we present robust and efficient two-stage approach to retrieve the identity of a reported phishing web page based on visual similarity between the suspect page and pages registered in the database. The presented framework is based on perceptual hash fingerprinting of the web pages. The evaluation of the proposed methodology is conducted on a dataset containing 15000 web screenshots from real phishing cases and 700 web pages which do not belong to phishing sites. The proposed solution reports an accuracy of 98.05% on the test dataset.

Keywords. Perceptual Hash, Phishing attack, Image fingerprinting, Phishing web

1. Introduction

The major focus of this work is to propose a solution oriented approach for phishing image identity/domain retrieval based on perceptual hashing methods, which are also known as fingerprinting or content-based media identification approaches. We consider a phishing page as a web page that, without permission, acts on behalf of a third party to confuse a visitor into doing some activity that he/she would only have done with a true representative of the third party, e.g., submit personal information. First phishing attack was reported by America online network systems (AOL) in the early 1990s [1] in which some fraudulent users registered on AOL with fake credit card details. When an entity like a financial institution has been alerted to the fact that they are under phishing attack, as a standard practice, they report it to local Law and Enforcement Agency (LEA), which normally take such sites down. Unfortunately, for LEAs, in the absence of some automated process, phishing analysis is a time-consuming manual process of identifying the information about phishing site and maintaining its records for future use.

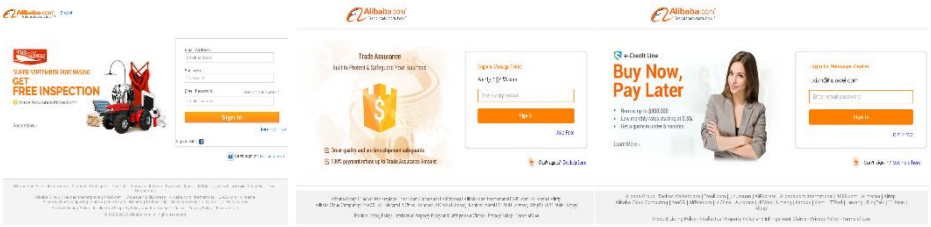

Figure 1. Different snapshots of same website created for phishing

Some previous works also report methods based on image similarity for phishing detection using visual appearance [2,5-12]. Zhou et al. [2], proposed a visual similarity based approach for phishing detection, in which they used local and global image features of the web pages to compare them. In the approach, for the local feature, they utilized image logo and to compute global features, they used whole visible pages. Marchal et al. [12], proposed 212 features based approach including HTML page text and learned a model to classify the page as phishing and no phishing. In [11], authors presented a review of various similarity based approaches for phishing detection.

In this work, we evaluated four perceptual hashing based approaches to extract features to measure web page similarity: average hashing (aHash) [5], perception hashing (pHash) [7], difference hashing (dHash) [6], and wavelet hashing (wHash) [8]. We also considered a fast keypoint descriptor to compute visual similarity among web pages: Oriented FAST and rotated BRIEF (ORB) [4].

2. Proposed Methodology

2.1. Background

To compute the visual similarity between web page screenshots, we utilized four perceptual hashing methods to generate a fingerprint extracted from web page content. Normally a perceptual hash code is a shortcode which can be represented as a binary sequence (e.g. 64 bits). Therefore, two hash codes can be compared with Hamming Distance [3]. The perceptual hash codes are also tolerant to various image transformations like scaling, minor coloring adjustment (e.g. contrast, brightness), skew, different aspect ratios or different compression/formats [7]. Since, apart from these transformations, phishing web pages may also have other variations like rotation, we also considered a well-known state-of-the-art keypoint descriptor named ORB in this study.

A. average hashing (aHash): it is a simple version of perceptual hashing and is computed as: Input Image→size reduce to 8x8 pixels→convert to grayscale→compute mean and threshold the 8x8 pixels with mean resulting into 64 bits.
B. difference hashing (dHash): it is similar to aHash, but it computes relative gradient direction by computing difference between adjacent pixels and setting 1 if difference is positive otherwise 0, resulting into 64 bit long code.
C. perception hashing (pHash): Input Image→size reduce to 32x32 pixels→ grayscale →(Discrete Cosine Transform→ comparison of the 64 DCT coefficients (top-left 8x8) with the mean of these 64 DCT coefficients resulting into 64 bits binary code.
D. wavelet hashing (wHash): It computes code in frequency domain similar to pHash, but uses DWT (Discrete Wavelet Transform) instead of DCT.

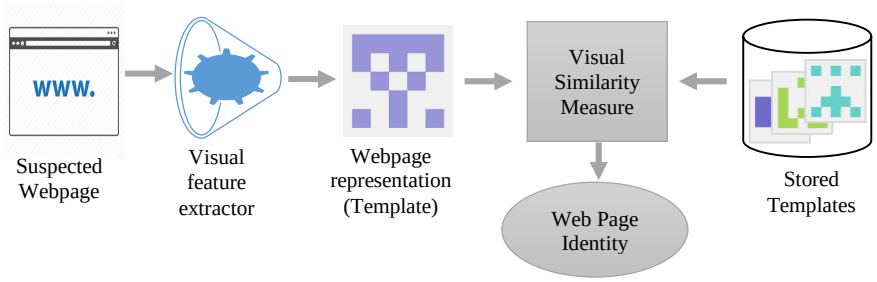

Figure 2: Illustration on Identity Retrieval

2.2. Semi-Automated Non-Redundant Phishing Web Page Database Creation Method

One of the biggest challenges in phishing web page identification is zero-hour phishing attack detection. To increase the identification rate, normally anti-phishing techniques compare the suspicious web page with a large pool of legitimate web pages or web pages belonging to previous phishing cases. Hence maintaining a database of web pages is a crucial step in the phishing detection process.

In this paper, we propose an approach to maintain a phishing web pages' database in a non-redundant way. For each domain, we add phishing web snapshots from the pool which are apart from each other within a range of hash distance or dissimilarity measure $D \in [thres1, thres2]$. In other words, if two templates have hash distance lower than thresh1, they are taken too similar and one of them is redundant. On the other hand, if *D* is higher than thresh2, they are considered to belong to different domains. The process is semi-automated because we found two scenarios when we have to populate pages manually to the repository: i) when adding a new domain which has only 1 page, ii) pages belonging to same domain but having a hash distance higher than thresh2.

2.3. A Two-stage process to retrieve Web Page Identity

In the approach (Fig. 1), the detection of the suspected web page is performed as a two-stage process. During the first stage, the test page is compared against all the reference templates stored in the data repository using the hash distance. If the dissimilarity score *D* for the test page with any of the stored template is lower than a predefined threshold (thresh3), the test page is considered as belonging to some phishing website. Otherwise it is considered as non-phishing page. In the second stage, the identity or domain name of the test page is identified. If, during the first stage, the test page is found as belonging to a phishing site, the identity of the domain with whom *D* is the lowest is assigned to it.

3. Experimental Analysis and Results

3.1. Experimental Setup

The experiments are carried out on a PC with Pentium 3.0 GHz processor and 8 GB RAM. To perform the comparative evaluation, we implemented five approaches (Section 2.1). Our dataset contains total 15000 images from 120 different reported phishing domains and 700 web pages from non-phishing websites. The samples in this dataset are unlabeled and to evaluate our proposed methodology we created two labeled sets; i) data repository with 2850 images which are selected using our proposed approach of section

Table 1. The Performance Analysis of different Approaches

Approach	Stage1: decision on phishing / no-phishing		Stage2-Identity Retrieval Accuracy (%)	Avg. running time per image in Seconds (hash code + matching)
	Threshold	Accuracy (%)		
aHash	3	79.84	88.93	0.0071
dHash	5	92.23	96.35	0.0073
wHash	5	85.10	89.41	0.011
pHash	**9**	**98.05**	**99.38**	0.012
ORB	--	--	86.04	0.050

2.2 (this set is used as gallery set), ii) test set using remaining 12150 phishing web page images + 700 non-phishing images.

The performance of ORB descriptor is evaluated based on a score computed as the number of matched keypoints divided by the total number of keypoints in the test image and repository images. The accuracy of ORB is computed through a one-step process i.e. by verifying the identity of the web page in repository which is found most similar to the test page. For ORB, the number of features parameter was set to 1000 and patch size was taken 31x31 pixels while keeping other parameter as default. In case of perceptual hashing approaches, the matching is performed using the Hamming Distance [3]. All the reported thresholds were found empirically, and the distribution of hash distances for inter-class, intra-class and among non-phishing web pages are shown in Fig. 3.

3.2. Phishing Page Identity Retrieval Accuracy

The results of our analysis are presented in Table 1. It is seen that pHash outperforms all other approaches evaluated. It also provides quite an impressive identity retrieval accuracy, which justifies the proposed methodology.

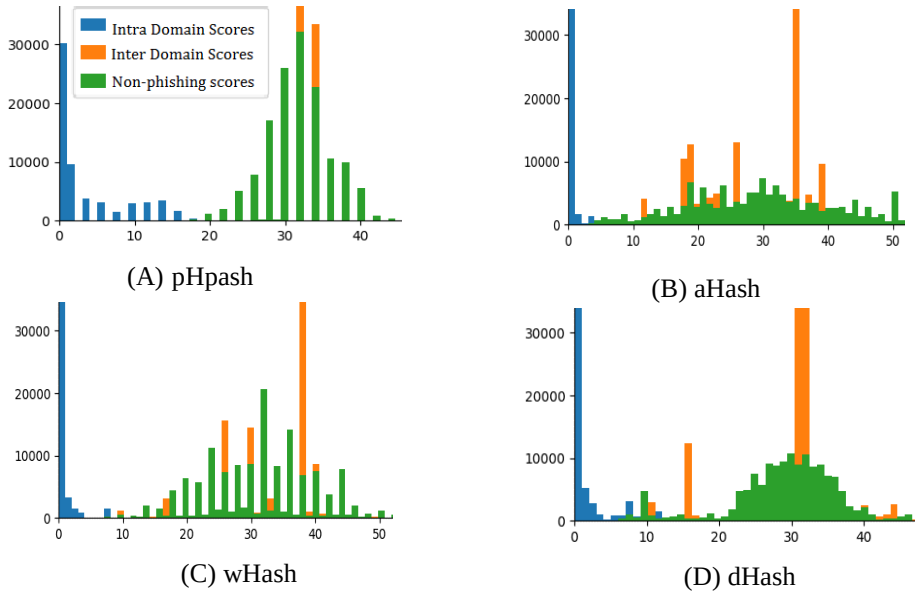

Figure 3: Hash Distance Distribution Graphs

4. Conclusion and Future Work

In this paper, we presented a new methodology to retrieve the identity or domain name of a phishing web page from a locally maintained repository. The approach is based on phishing page fingerprinting using visual characteristics and using visual similarity the identity of a new page is retrieved using a proposed two-stage process. Further, we introduced a semi-automated approach to create a non-redundant phishing web page repository. For evaluation, we created a labeled dataset from real reported phishing cases. Based on the overall very high accuracy obtained through the proposed methodology it can be deduced that process of phishing web page identity retrieval can be automated based on visual features. Another finding is that by following proposed semi-automated repository creation process, the number of templates in the data repository can be reduced up to 80% while getting good accuracy. It will reduce the search time by huge margin especially in case of large datasets.

As future work, we plan to enlarge our dataset with more templates, collected for example from http://www.phishtank.com/, and improve the detection rate at stage1.

References

[1] M. Jakobsson and S. Myers, Phishing and Countermeasures: Understanding the Increasing Problem of Electronic Identity Tef, Wiley-Interscience, 2006.

[2] Y. Zhou, Y. Zhang, J. Xiao, Y. Wang and W. Lin, "Visual Similarity Based Anti-phishing with the Combination of Local and Global Features," *2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications*, Beijing, 2014

[3] M. Norouzi, D. J. Fleet, and R. R. Salakhutdinov, "Hamming distance metric learning," in Advances in neural information processing systems, 2012, pp. 1061–1069

[4] Rublee, Ethan; Rabaud, Vincent; Konolige, Kurt; Bradski, Gary, "ORB: an efficient alternative to SIFT or SURF" (PDF). IEEE International Conference on Computer Vision (ICCV) 2011.

[5] S. F. C. Haviana and D. Kurniadi, "Average hashing for perceptual image similarity in mobile phone application," Journal of Telematics and Informatics, vol. 4, no. 1, pp. 12–18, 2016.

[6] V. Bajaja, S. Keluskar, R. Jaisawala, and R. Sawant, "Plagiarism detection of images," International Journal of Innovative and Emerging Research in Engineering, vol. 2, pp. 140–144, 2015.

[7] C. Zauner, M. Steinebach, and E. Hermann, "Rihamark: perceptual image hash benchmarking," in IS&T/SPIE Electronic Imaging. International Society for Optics and Photonics, 2011.

[8] D. Petrov. Wavelethash. [Online]. Available: https://fullstackml.com/wavelet-image-hash-in-python-3504fdd282b5 (accessed September 20, 2017).

[9] A. Swaminathan, Yinian Mao and Min Wu, "Robust and secure image hashing," in IEEE Transactions on Information Forensics and Security, vol. 1, no. 2, pp. 215-230, June 2006

[10] C. Zauner. Implementation and benchmarking of perceptual image hash functions. Master's thesis, Austria, July 2010

[11] Ankit Kumar Jain and B. B. Gupta, "Phishing Detection: Analysis of Visual Similarity Based Approaches," Security and Communication Networks, vol. 2017, Article ID 5421046, 20 pages, 2017.

[12] S. Marchal, K. Saari, N. Singh, and N. Asokan, "Know Your Phish: Novel Techniques for Detecting Phishing Sites and their Targets," oct 2015