Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/cose

Computers & Security

Utilisation of website logo for phishing detection

Kang Leng Chiew *, *Ee Hung Chang*, *San Nah Sze*, *Wei King Tiong*

Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak, 94300 Kota Samarahan, Sarawak, Malaysia

ARTICLE INFO

Article history:
Received 2 February 2015
Received in revised form 4 July 2015
Accepted 31 July 2015
Available online

Keywords:
Anti-phishing
Website logo
Website identity
Google image search
Identity consistency
Logo extraction

ABSTRACT

Phishing is a security threat which combines social engineering and website spoofing techniques to deceive users into revealing confidential information. In this paper, we propose a phishing detection method to protect Internet users from the phishing attacks. In particular, given a website, our proposed method will be able to detect if it is a phishing website. We use a logo image to determine the identity consistency between the real and the portrayed identity of a website. Consistent identity indicates a legitimate website and inconsistent identity indicates a phishing website. The proposed method consists of two processes, namely logo extraction and identity verification. The first process will detect and extract the logo image from all the downloaded image resources of a webpage. In order to detect the right logo image, we utilise a machine learning technique. Based on the extracted logo image, the second process will employ the Google image search to retrieve the portrayed identity. Since the relationship between the logo and domain name is exclusive, it is reasonable to treat the domain name as the identity. Hence, a comparison between the domain name returned by Google with the one from the query website will enable us to differentiate a phishing from a legitimate website. The conducted experiments show reliable and promising results. This proves the effectiveness and feasibility of using a graphical element such as a logo to detect a phishing website.

1. Introduction

The emergence of information technology has made our life easier. For example, we no longer need to be available in front of the bank counter to make a transaction. We can do this by using a personal computer through the Internet infrastructure. The online application platform is huge and encompasses a vast variety, ranging from casual information sharing (e.g., social networking) to a more monetary intensive related application (e.g., Internet banking, bill payment, E-commerce, etc.).

The popularity of this infrastructure has attracted immoral parties to gain profit illegally. This illegal profit oriented threat is considered an online crime. One of the simple yet effective threats is a phishing attack. Usually in the phishing attack, the phisher will send a huge number of emails that impersonates as it was sent from a genuine party. Typically, the email content is crafted to create a sense of urgency, worry, or offer some great incentive and asks the victims to take action. For example, the email will urge victims to update their confidential information (e.g., login password) before his or her account is suspended. Once the victim innocently updates the confidential information, the phisher will gain all necessary details. They will utilise them for illegal access purposes, as the information the user transmits is sent to the phisher's counterfeit website rather than the genuine one.

* *Corresponding author.* Tel.: +60 82 583762/3758.
E-mail address: klchiew@fit.unimas.my (K.L. Chiew).

http://dx.doi.org/10.1016/j.cose.2015.07.006
0167-4048/© 2015 Elsevier Ltd. All rights reserved.

Phishing is a very serious security threat, and it causes a multitude of pitfalls which include identity theft, stolen money, unauthorised account access, and credit card fraud. The impact of this threat is dreadful, and causes tremendous financial losses every year. Besides the tangible losses, phishing also causes long term damage (i.e., reputation, credibility and confidence losses) to the customer–company relationship.

The main reason that makes phishing attacks possible to perpetrate is the lack of computer knowledge among Internet users. Many Internet users do not know how the web applications work. For example, they do not understand the syntax or the meaning of the Uniform Resource Locators (URLs), and cannot differentiate a fraudulent website from a legitimate one. Lack of computer knowledge also blindfolds the Internet users from utilising the security indicators, which are normally available in the Internet browser. With the advancement of web technology, it is possible for the phishing attacks to exploit visual deception. This technique ranges from manipulating textual to graphical form. For example, the phishers may choose the number one to substitute the alphabet one in http://www.paypal.com (Dhamija et al., 2006), hence creating a fraudulent website that looks similar to the legitimate one. Whereas for the graphical form, the phishers may use Javascript to load the secure padlock icon in the address bar to indicate that the website is secure and deceive the users to believe that it is a legitimate website. In addition, security is often treated unconsciously as a secondary goal by most Internet users when focusing on their primary tasks (i.e., performing online banking, transactions, etc.) (Dhamija et al., 2006).

Clearly, we can see that the attackers exploit the human factors (e.g., computer illiteracy and carelessness) as the loophole. While improving public awareness is important in fighting against phishing attacks, it is even more crucial and necessary to equip the users with a more automated security mechanism. Currently, the security mechanisms can be broadly divided into list-based and heuristic-based approaches. A list-based approach is assessing the existence of a query website (e.g., the URL of a suspicious website) to the set of entries stored in the predefined list. The list can be a blacklist, a whitelist, or both. On the other hand, a heuristic-based approach is based on the mechanism of extracting some distinctive features or characteristics from the query website to facilitate the detection and identification of a phishing website. The list-based approach is fast and produces low false positives, but its effectiveness is only as good as its up-to-date list. While the heuristic-based approach comparatively takes more computational power, it is more preferable due to its flexibility to detect a new phishing website.

In this paper, we propose a method that belongs to the heuristic-based approach, and it is an extension of our work proposed in Chang et al. (2013). We claim that in order to detect a phishing website, we must first be able to determine the consistency of the website identity. With the consistency, we will be able to assess the legitimacy of a website. Among the many elements within a website, a logo is the most suitable candidate because it is the official trademark and representative of a website. As we all know, the relationship between a logo and the domain name of a website is exclusive; any mismatch is an indication of a phishing attack. In Chang et al. (2013), we had illustrated and proved that it is possible to identify the associated legitimate website from querying the logo image through a Google image search. We applied fixed segmentation with manual best-fit cropping to extract the logo. We obtained four different sizes of segmentation from a rendered website screenshot, and utilised the Google image database to identify the website identity. We performed Google image searches using the segmented images, and used the returned keywords to perform a second search by using Google text Search. The top 30 URLs from the second search results were recorded and the legitimacy of the query websites were determined based on the comparison between the domain name of a query website with the URLs from the search results.

Differing from Chang et al. (2013), in this paper, we employ image processing and machine learning techniques to locate the logo. From the logo, we utilise Google Images as a source of a knowledge database to determine the website identity. Immediate comparison between the domain name of the determined identity with the one from the query website will enable us to differentiate a phishing from a legitimate website.

While there are many heuristic-based methods in the literature, they are different from the one proposed in this paper. They are either determining the identity of a website based on textual elements or direct evaluation based on some form of phishing characteristic without knowing the identity. The former is similar to ours, but different in the context, and the main weakness is its textual semantic gap. Whereas the latter is like finding some evidence from the wild without any baseline, and it has many uncertainties. While it might be effective to detect existing phishing (i.e., with a known phishing characteristic), it is certainly not effective for an unknown phishing (zero-day phishing) attack. For example, evaluating the URL for domain name obfuscation (a phishing characteristic) will fail when a legitimate website is injected with a phishing webpage. For another example, by assessing the structure of an HTML page (e.g., DOM) for abnormality, it is difficult to judge the legitimacy of a website, simply because the phisher can have the exact clone of the website. Further discussion of the existing techniques will be given in Section 2.

The remainder of the paper is structured as follow. In the next section, we will discuss some of the related works. We will then discuss in detail the proposed method in Section 3. In Section 4, we present the experimental results. We will later give the analysis in Section 5. Section 6 concludes the paper.

2. Related work

While there exists numerous different techniques in phishing detection, they can be divided into list-based and heuristic-based approaches. According to Huh and Kim (2012), one of the popular techniques is blacklisting. Many popular web browsers are using this approach to detect phishing website (Abrams et al., 2013; Schneider et al., 2008). In this technique, a query website is checked with a list (i.e., a list of known phishing URLs), which is compiled and maintained by some consortium or organisation. If the checking returns a match, then the website will be labeled as phishing. On the contrary, instead of maintaining the blacklist, one can compile a list of legitimate URLs. This technique is known as whitelisting, and it is also a type of list-based approach. An example of a whitelisting technique is the research proposed by Cao et al. (2008). The authors developed an automated technique that maintains and stores a whitelist at the client side. A more dynamic and flexible list-based approach is called PhishNet (Prakash et al., 2010). This method uses several URL variation heuristics to process the existing blacklisted URLs and generate multiple variations URLs. The generated URLs will form a predictive blacklist. The results showed that it can effectively detect new and old phishing websites. Although a list-based approach provides simplicity in design and is easy to implement, keeping the list complete and up-to-date requires great effort, and always suffers from incompleteness.

A more prominent approach which receives great attention is the heuristic-based approach. This approach overcomes the issue of over-reliance on a predefined list to detect a phishing website. The heuristic-based approach relies on the analysis of some discriminative properties extracted from the query website to decide if it is a phishing website. Usually, the discriminative properties are extracted from the HTML content (e.g., Document Object Model (DOM) structure, textual and graphical content), the structure of the URL, and third party information (e.g., Whois lookup and Alexa). Due to its flexibility and ability to detect new phishing websites, there exists a variety of techniques proposed for this approach.

In general, there are two facets to approach a heuristic-based method, namely identity-mediated and unmediated. The former relates to a method which will first determine the portrayed identity of the website. If the method can prove the portrayed identity is different than the query website's identity, then it is a phishing website. For example, a query website A claims it is PayPal, but an evaluation on the URL of the query website shows that it does not belong to the PayPal domain. Hence, website A is impersonating PayPal and it is a phishing website. For the second facet, the method will start with the feature extraction process, and then follow it by the detection process. Some of the common ways used in the detection process include machine learning and similarity measurement. Note that this method does not require determining the portrayed identity prior to the detection, and the portrayed identity is only used indirectly.

A notable example of the first facet is the one proposed by Xiang and Hong (2009). The method consists of two main components, namely identity recognition and keyword-retrieval detection components. In its first component, the method will use two textual objects from the DOM (i.e., the title and copyright field) to determine the identity. If the title and copyright field is missing, the method will employ a named entity recognition (NER) technique to find the identity. Whereas for the second component, the method will use the TF-IDF technique to identify the keywords. It is invoked only when the first component failed. Later, both of the components will utilise the search capability from Google and Yahoo to decide if a query website is a phishing website.

A more recent research study on the identity-based heuristic approach is the one proposed by Liu et al. (2010). The main objective of the method is to determine the identity of the phishing target when a phishing webpage is detected. The method is based on the idea of a self-organised semantic data model, called the Semantic Link Network which is usually used in organising web resources. Although this method is using different detection mechanisms, its groundwork starts from the textual elements (i.e., the extraction of hyperlinks, keywords and textual contents for the operation of link relations, search relations and text relations, respectively).

Another method related to the identity-based heuristic approach is GoldPhish (Dunlop et al., 2010). The main idea is to leverage the Google search engine to determine the identity of a webpage through the textual content. The authors will capture a screenshot of the webpage, and perform optical character recognition (OCR) to extract all textual contents (including the text within a logo). Then the authors will feed the extracted text into the Google search engine and evaluate the search results. The authors believed that through the extracted text, the Google search engine will be able to return the most relevant website. The evaluation of the search results involves checking for matching between the domain name of the relevant website to the query website. A mismatch means that the portrayed identity (relevant website) is different than the real identity (query website) which concludes that it is a phishing webpage.

It is worth mentioning that the above identity-based heuristic approaches are all based on the analysis of textual elements. The main limitation of using textual elements to determine the website's identity is the constraint of language dependency. In other words, the detection method will only work with its optimum efficacy for websites written in a specific language. For example, an English-based detection method is only effective for website written in the English language, but not for the Spanish language. To the best of our knowledge, there are very few research studies from the identity-mediated heuristic approach in the literature which use graphical elements to determine the identity of a website. In particular, there is no research employing the logo to determine the website's identity as the one proposed in this paper. The closest method to ours is GoldPhish (Dunlop et al., 2010). However, GoldPhish is only using the text within the logo and not the logo as a whole.

The majority of the proposed techniques in the literature belong to the second facet, the unmediated heuristic approach. One of the popular methods is CANTINA (Zhang et al., 2007). This method will calculate the TF-IDF from the content of a webpage, and generate a lexical signature. The method will use the generated lexical signature to perform a web search using the Google search engine. The returned result will be used to determine the legitimacy of a website. Although the method can perform reasonably well in the detection of phishing, it is unable to identify the phishing target.

Another popular method is to employ a machine learning technique. For example, Garera et al. analysed the structure of URLs to determine various patterns which are usually exploited by phishers (Garera et al., 2007). From the analysis, the authors proposed several features which include page, domain, type, and word based features. With these features, the authors used logistic regression to distinguish a phishing from a legitimate webpage. Similar methods which used a machine learning technique can be found in Sorio et al. (2013) and Chu et al. (2013).

Obfuscating the URL to deceive unsuspecting users is one of the popular deceptions used by phishers. The deceptions are simple, yet effective. This has attracted considerably high attention from the anti-phishing researchers to focus on a URL-based approach (Maurer and Höfer, 2012). Nguyen et al. (2013) and Verma and Dyer (2015) are among the works that contribute to the URL-based approach. An interesting one is

Please cite this article in press as: Kang Leng Chiew, Ee Hung Chang, San Nah Sze, Wei King Tiong, Utilisation of website logo for phishing detection, computers & security (2015), doi: 10.1016/j.cose.2015.07.006

Maurer and Höfer (2012) which uses spelling recommendations from a search engine and a string similarity algorithm. Since each URL is unique, phishers can only use small spelling mistakes that might go overlooked by unsuspecting users to impersonate a legitimate URL. The spelling recommendation is used to predict the targeted legitimate URL, and the string similarity algorithm is used to find the level of similarity between the legitimate URL and the suspected URL. If the similarity is high, it is a sign of a phishing website. As acknowledged by the authors, the proposed method may fail if the phishing URL does not contain any spelling mistake.

Other interesting unmediated heuristic approaches include using visual similarity measurement to detect phishing webpages. In general, the method consists of two components: visual feature extraction and similarity measurement components. The visual feature extraction component will extract a set of features (e.g., wavelet, scale-invariant feature transform (SIFT), etc.) from the query webpage. Based on the feature set, the similarity measurement component will compute the similarity score between the query webpage and all the webpages in the database. If the similarity score is exceeded by a certain threshold (highly similar) and the query webpage is not from the domain name recorded in the database, it will be detected as a phishing webpage. Note that this method needs a great effort to keep the database complete and up-to-date in order to be effective. Research in Hara et al. (2009), Mao et al. (2013), and Zhang et al. (2013) belonged to this type of method.

In a rule-based approach, the classification between phishing and legitimate webpage is governed by a set of rules. Mohammad et al. (2014) proposed a method that uses a combination of 16 rules and a classifier to perform the classification. The rules are used as the transformation agents that will transform different features into three states, namely, legitimate, suspicious, and phishy. The features are extracted from the webpages, and can be categorised as address bar-based, abnormal-based, HTML-Javascript-based, and domain-based features. The states produced by each rule will be fed to a classifier. The authors claimed that a C4.5 classifier gave the best classification results. Clearly, this method is superior in term of speed, since it has less dimensionality (i.e., only the three states as opposed to the raw features values). However, the effectiveness towards more complicated phishing webpages such as a multiple identities phishing webpage is not guaranteed, since the transformation will definitely cause information loss. Similar rule-based approaches include Cook et al. (2009) and Abdelhamid (2015).

3. Proposed methodology

It is noteworthy to mention that the main objective of a phishing website is to deceive users into believing that the phishing website they are visiting is the legitimate website. In order to do this, phishers will clone a phishing website that visually resembles the legitimate website. Usually phishers will rip off the visual components (i.e., logo, emblem, or trademark) from the legitimate website and use them in their phishing website.

In order to detect a phishing website, the first question to ask is: how to differentiate a phishing website from a legitimate website, given the fact that they look identical? If we can somehow determine the portrayed identity of a query website (if the query website is a phishing website, the portrayed identity will be the identity of the targeted legitimate website), we can then differentiate the phishing website from the legitimate website. Knowing that the phishers will use the visual components ripped off from the legitimate website, especially the logo, in their phishing websites, this motivates us to propose an anti-phishing method based on the identification of website identity through the logo. This is rational, as the logo usually represents the identity of a legitimate website.

Even though anti-phishing methods based on textual elements receive more attention, it has some limitations. For example, choosing the right terms using term frequency-inverse document frequency (TF-IDF) from the webpage content is challenging. In other words, extracting the right keyword which is representative of the website's identity is difficult. Phishers can easily evade and jeopardise the detection mechanism by introducing a few unrelated but statistically significant terms in the content. Furthermore, identity detection based on textual information is sometimes indistinguishable and noisy. Hence, we believe that using a graphical element, especially the logo, is important and complemental. This will compensate for the limitations faced in textual-based methods, and will make the detection more robust.

To ease the discussion, we use the following definition:

- *Query website*: The website which is under scrutiny.
- *Portrayed identity*: The brand or entity for which a legitimate website is asserted to. For example, the portrayed identity for a legitimate website with the domain http://www.paypal.com is PayPal. Likewise, for a phishing website (e.g., domain http://www.www1-paypai.com) which is impersonating the PayPal website, the portrayed identity is PayPal.
- *Real identity*: The actual identity of a query website. For example, PayPal is the real identity of the website with domain http://www.paypal.com. Whereas for a phishing website with the domain http://www.www1-paypai.com which is impersonating the PayPal website, its real identity is www1-paypai.
- *Phishing target*: The legitimate website which is targeted by a phisher. In other words, it is the website which a phishing website is trying to impersonate.

The proposed method involves two main processes: logo extraction and identity verification. A logo extraction process will extract the logo from the query website. Based on the extracted logo, the identity verification process will evaluate the consistency between the real identity and portrayed identity of the query website. If the identity is consistent, the query website is legitimate, and vice versa. Fig. 1 shows the framework of the proposed method.

3.1. *Logo extraction*

The task of identifying and extracting a logo from the webpage, however, may be very difficult to achieve. We may start with a more realistic and modest goal, such as retrieving all images from the webpage regardless of whether they are a logo or non-logo. Thus, the first subprocess of logo extraction is to download

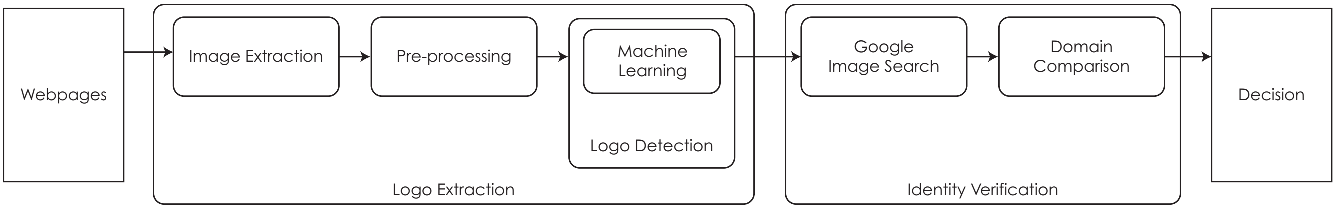


Fig. 1 – Framework of the proposed method.

all images from a query webpage. We apply pre-processing to discard images with a property which is unlikely to be a logo image. Namely, we exclude any image with a width or height which is less than 10 pixels, or an image with one colour (pixel intensity of one).

According to Baratis et al. (2008), a logo image usually has characteristics which are different than a non-logo image. In their research, they employed a machine learning technique to classify logo and non-logo images. In order to represent the images statistically, they used a features set of 23 dimensions, and fed into a decision tree for the classification.

Motivated by Baratis et al.'s (2008) work, we employed a similar machine learning technique in the logo detection subprocess to find the right logo image. However, we only selected eight features from their 23 features. We only selected the most sensitive features and not all because of the response time constraint. Using all the 23 features would be very time consuming and impractical for real time phishing detection. In addition, we notice that the height and width ratio for a logo image is around one. Unlike a logo image, the ratio for most non-logo images has a high deviation (i.e., very much higher or lower than one). Besides, discrete cosine transform (DCT) is widely used in image processing. One of its properties is the ability to pack input data into as few coefficients as possible, which is known as energy compactness. Different image content will certainly have a different compactness level. Hence, it is useful to use the energy compactness as an additional feature. We add these two features to the eight features selected from Baratis et al.'s work and to form the final features set. Note that the majority of the proposed features are derived from a pixels intensity distribution. Given an image, we first compute the pixel intensity histogram $H = [h_0, h_1, h_2, \ldots, h_{255}]$, where h_i indicates the histogram bin at i-th pixel intensity. After that, we normalise the histogram to get the pixels intensity distribution as $h_i' = \frac{h_i}{\sum_{j=0}^{255} h_j}$. The proposed features set is listed as follows:

- *Mean*: Mean of pixels intensity distribution, $\frac{\sum_{i=0}^{255} h_i'}{256}$.
- *Standard deviation*: Standard deviation of pixels intensity distribution, $\sqrt{\sum_{i=0}^{255} (i-\mu)^2 h_i'}$. Where μ is the mean of pixels intensity distribution.
- *Skewness*: Skewness of pixels intensity distribution, $\sum_{i=0}^{255} \left(\frac{i-\mu}{\sigma}\right)^3 h_i'$. Where σ is the Standard deviation of pixels intensity distribution.
- *Kurtosis*: Kurtosis of pixels intensity distribution, $\sum_{i=0}^{255} \left(\frac{i-\mu}{\sigma}\right)^4 h_i' - 3$.
- *Energy*: Energy measures the homogeneousness of image content, $\sum_{i=0}^{255} (h_i')^2$.
- *Significant pixel intensity*: The number of significant pixel intensity. We count the number of bins of pixels intensity distribution with a value greater than 0.02 (0.02 is determined empirically). Namely, the number of bin $h_i' > 0.02$ for $i \in [0 \ldots 255]$.
- *Entropy*: Entropy measures the average bits per pixel, $\sum_{i=0}^{255} h_i' \log_2 h_i'$.
- *Otsu threshold*: A threshold that separates pixels into dark and light regions (Otsu, 1979).
- *Image resolution ratio*: Image height and width ratio.
- *Energy compactness*: Discrete cosine transform image and compute the entropy of the DCT coefficients, $\sum h_{dct} \log_2 h_{dct}$. Where h_{dct} is the histogram of DCT coefficients with 256 bins.

Note that a webpage can have many different images, namely logo, scenery, clipart (e.g., button and icon), etc. Identifying a logo from these images without fault is very difficult because the characteristic of some images is very similar to the logo. For example, some clipart highly resembles a logo. We acknowledge that current implementation of the logo extraction process may sometimes return more than one image (i.e., logo and non-logo images). However, this limitation is uncritical. As mentioned above, these images are usually obtained from the legitimate website, and these images are somehow associated with the portrayed identity and will be captured in the next subprocess. This turns out to be sometimes useful, especially in detecting a phishing website. We will discuss this issue in Section 5.

Like any other machine learning approaches, the proposed method will perform classification based on the extracted feature sets. There are many classifiers available, namely, neural network, support vector machine, Fisher linear discriminant etc. Since our focus is not on the classification, we have deliberately selected support vector machine (SVM) as the classifier due to its promising performance. We acknowledge that the use of SVM may be suboptimal; however, changing to an optimal classifier later is effortless. We use the SVM library implemented in Chang and Lin (2001) with the default setting (i.e., radial basis function is used as the kernel function, and the values for parameter γ and C is set to $\frac{1}{n}$[1] and 1.0, respectively).

[1] n is the number of features used during the classification and in our case, it is 10.

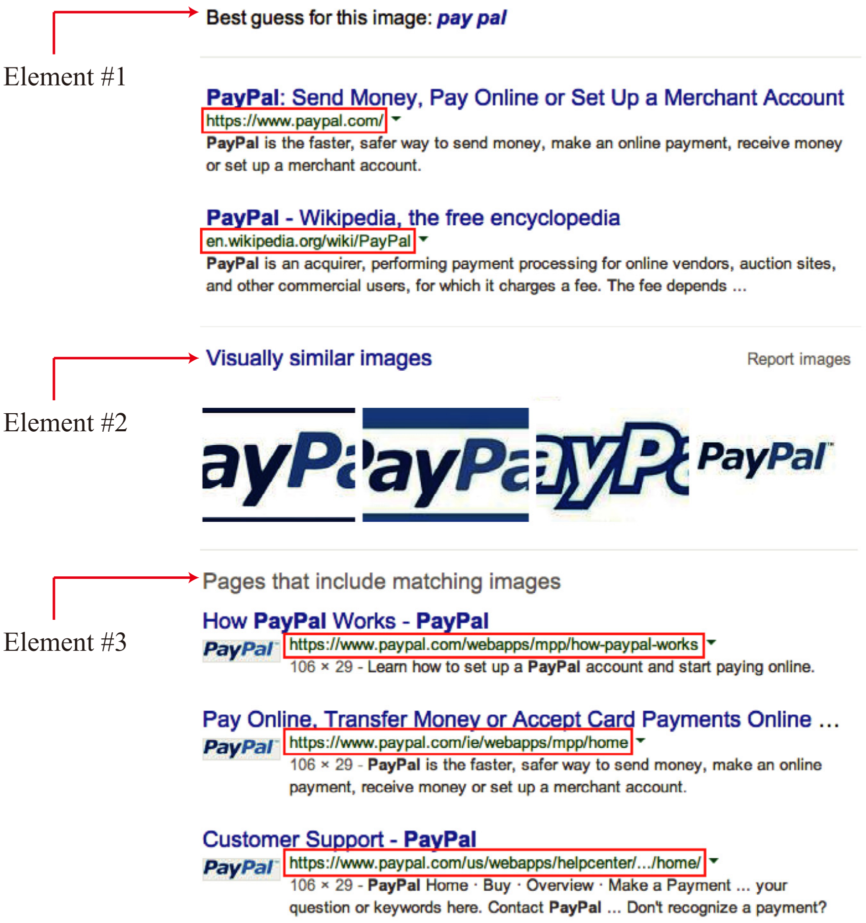
3.2. Identity verification

Consistent identity means that the real identity and the portrayed identity is identical. The real identity can be obtained from the domain name of the query website. Whereas the portrayed identity can be retrieved from a database which entry has logo matches to the extracted logo. Since the relationship between the logo and domain name of a website is exclusive, any mismatch is an indication to the phishing attack. Clearly, a complete and up-to-date database of different website logos with the corresponding domain names is needed. Maintaining this database effectively alone is impossible. Hence, we utilise Google Images as a source of the knowledge database. To fully utilise the Google Images database, we employed the content-based image retrieval feature from the Google image search facility. It allowed us to retrieve the portrayed identity of a query website from the vast image database. This is depicted as a Google image search subprocess in Fig. 1.

The output from the Google image search subprocess is the search result which includes elements such as: i) the best guess for the query image, ii) a list of visually similar images, and iii) pages that include matching images. Figure 2 shows an example of a Google image search result using the PayPal logo image as the search query. The current setting is to use only the first page of the search result. Next, a domain comparison subprocess will parse the URLs (as marked with the red outlines shown in Fig. 2) from the first and third elements of the search result. After that, this subprocess will extract only the domain name from each of the URLs and compare them to the domain name of the query website. We have taken the liberty to refer to the name which excludes the TLD (top level domain) and any sub domain as the domain name. For example, the domain name for http://www.mydomain.com is mydomain. If the comparisons return at least one match, our method will classify the query website as legitimate. Otherwise, it is classified as a phishing website. As for the limitation in the logo detection subprocess mentioned above (i.e., when multiple images of logo and non-logo are returned), we will repeat the identity verification process for each image, and aggregate the comparison results. Similarly, our method will classify the query website as legitimate if the aggregated comparisons return at least one match.

3.3. Experimental results

We have constructed two non-overlapping datasets from a total of 1140 webpages. The webpages consist of phishing and legitimate webpages. Phishing webpages are downloaded from



Best guess for this image: **pay pal**

PayPal: Send Money, Pay Online or Set Up a Merchant Account
https://www.paypal.com/
PayPal is the faster, safer way to send money, make an online payment, receive money or set up a merchant account.

PayPal - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/PayPal
PayPal is an acquirer, performing payment processing for online vendors, auction sites, and other commercial users, for which it charges a fee. The fee depends ...

Visually similar images

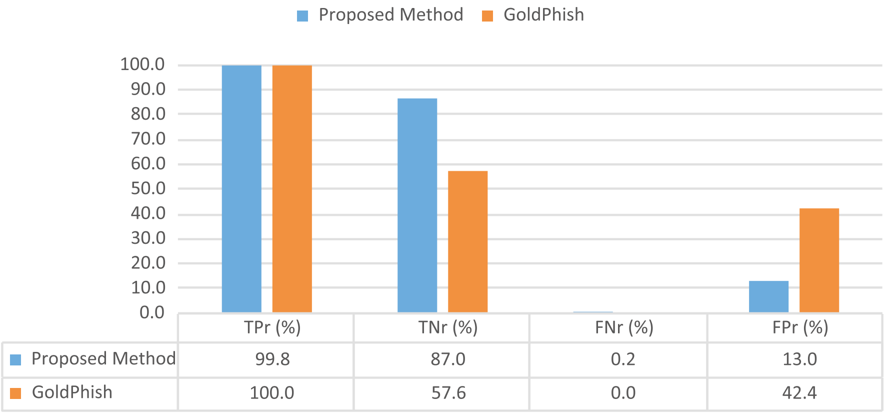
Pages that include matching images

How **PayPal** Works - **PayPal**
https://www.paypal.com/webapps/mpp/how-paypal-works
106 × 29 - Learn how to set up a **PayPal** account and start paying online.

Pay Online, Transfer Money or Accept Card Payments Online ...
https://www.paypal.com/ie/webapps/mpp/home
106 × 29 - **PayPal** is the faster, safer way to send money, make an online payment, receive money or set up a merchant account.

Customer Support - **PayPal**
https://www.paypal.com/us/webapps/helpcenter/.../home/
106 × 29 - **PayPal** Home · Buy · Overview · Make a Payment ... your question or keywords here. Contact **PayPal** ... Don't recognize a payment?

Fig. 2 – Search results returned by Google image search when the PayPal logo is used as the search query.



	TPr (%)	TNr (%)	FNr (%)	FPr (%)
🟦 Proposed Method	99.8	87.0	0.2	13.0
🟧 GoldPhish	100.0	57.6	0.0	42.4

Fig. 3 – Comparison of detection performance between the proposed method and GoldPhish.

PhishTank[2] and legitimate webpages are from Alexa[3] within different categories (i.e., banking, social networking, news, e-commerce, forums and blogging). We manually downloaded these webpages and organised them into folder by folder. Each folder consists of one webpage with all the image resources. We have divided them into 140 webpages for dataset 1 and the remaining 1000 webpages for dataset 2. We also constructed an additional dataset 3, which consists of 500 other unpopular legitimate webpages. In addition to http://www.alexa.com, we also obtained the webpages from http://www.dmoz.org and http://botw.org.

Dataset 1 is used for the logo extraction process as discussed in Section 3.1. It consists of a total of 3894 images (i.e., 1947 logo images and 1947 non-logo images). To ensure that the proposed method is able to detect all sorts of logo images, we use the logo and non-logo images from both phishing and legitimate websites image resources. This is important, as sometimes the phisher will try to evade the detection by slightly altering the logo image (e.g., scaling down a logo image). For dataset 2, it is used for the identity verification process as discussed in Section 3.2. Dataset 2 contains 500 phishing and 500 legitimate webpages. The final detection performance of the whole proposed method is measured using dataset 2. The purpose of dataset 3 is to evaluate the effectiveness of the proposed method in some very challenging situations, namely the very unpopular legitimate webpages. They are ranked between 1001 to 5150 in Alexa and 16.4 percent of them (82 webpages) are not ranked in the Alexa top one million websites. These webpages usually have one or more of the following characteristics: newly launched, low in quality (not W3C compliant), not optimised for SEO, and excessive use of frames. Note that we do not include additional phishing webpages to dataset 3 because the lifespan for phishing webpages is usually short and does not fall into popular and unpopular categories. Hence, it does not complicate the detection mechanism (this is verified by the high true positive rate for both the proposed method and GoldPhish in Fig. 3).

The experiment is designed to evaluate the effectiveness of our proposed method, as well as comparing our performance with another anti-phishing method, GoldPhish (Dunlop et al., 2010). We selected GoldPhish as a comparison because it has a similarity to our proposed method, where the main idea is to leverage the Google search engine to determine the identity of a webpage through the extracted contents. However, GoldPhish is using a different approach on the feature extraction: it is only focused on extracting the textual contents. The method starts by capturing a screenshot of the webpage, and performs OCR to extract the textual contents (including the text within a logo). Then the authors will feed the extracted text to the Google search engine and evaluate the search results. Unlike our proposed method, GoldPhish uses Google textual search. The authors believed that through the extracted text, the Google search engine would be able to return the most relevant website. The evaluation on the search results involves checking for matching between the domain name of the relevant website to the query website. A mismatch means the portrayed identity (relevant website) is different than the real identity (query website), which concludes that it is a phishing webpage.

We implemented GoldPhish in C#, and evaluated our proposed method with GoldPhish on the same datasets. Figure 3 shows the detection results for the experiment. We abbreviate the performance metrics as TP_r, TN_r, FP_r and FN_r for the rate of true positive, true negative, false positive, and false negative, respectively.

From Fig. 3, both methods achieve a high true positive detection rate, and GoldPhish performs the best with 100.0% of a true positive rate. Although our proposed method has a lower true positive rate at 99.8%, it still can be concluded as a highly significant detection rate. For the true negative detection, GoldPhish and our proposed method scored the rate of 57.6% and 87.0%, respectively. The results show no significant difference between our proposed method and GoldPhish in true positive detection, but ours has a notable leading of 29.4% higher than GoldPhish in true negative detection.

In order to get a clearer and better understanding of the detection performance, we consider *Accuracy* as an additional performance metric. It measures the percentage of test set

[2] http://www.phishtank.com
[3] http://www.alexa.com

Please cite this article in press as: Kang Leng Chiew, Ee Hung Chang, San Nah Sze, Wei King Tiong, Utilisation of website logo for phishing detection, computers & security (2015), doi: 10.1016/j.cose.2015.07.006

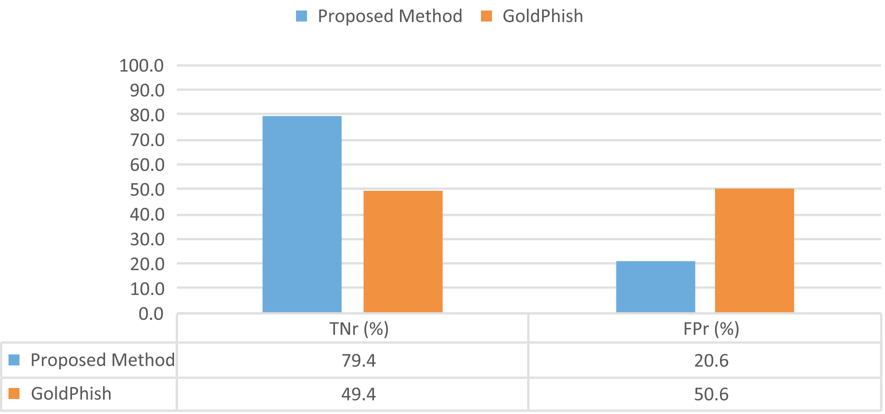


Fig. 4 – Results comparison using Dataset 3.

tuples that are correctly classified. *Accuracy* is defined as $\frac{TP+TN}{Pw+Lw}$, where Pw and Lw represent the total number of phishing and legitimate webpages, respectively. TP and TN denote the number of true positive and true negative webpage samples, respectively. The proposed method achieved *Accuracy* of 93.4%, while GoldPhish achieved 78.8%.

As for the experiment carried out on dataset 3, the result shown in Fig. 4 indicates a performance drop for both the proposed method and GoldPhish. This is not surprising, as dataset 3 is purposely constructed with some uncommon characteristics (more detail is discussed in the next section). Comparatively, the proposed method only dropped 7.6%, whereas GoldPhish dropped more at 8.2%. The proposed method achieves 79.4% of a true negative rate and 20.6% of a false positive rate. These rates are still considered acceptable, given the nature of the dataset. While GoldPhish only manages to achieve 49.4% and 50.6% for the true negative and false positive rate, respectively.

4. Analysis and discussion

GoldPhish, an image contents-based phishing detection method, has done very well in detecting the phishing websites, where it scored 100.0% of a true positive detection rate in this experiment, but incorrectly labelled nearly half of the legitimate websites as phishing. The low true negative detection is likely due to the fact that GoldPhish relies on the extracted textual contents. Thus, its performance is greatly dependent on the accuracy and the significance of the OCR extracted texts. Due to the English-based OCR tool used in GoldPhish, its detection ability is only limited to English written websites (Dunlop et al., 2010). Therefore, it failed to extract the significant texts from our dataset which consists of a variety of foreign language websites, such as Arabic, Chinese, and others. On the contrary, our proposed method does not suffer this limitation because we are using the logo image as the search query.

From Fig. 3, we can clearly see that the proposed method gives promising results. This justified the effectiveness and feasibility of our proposed method to detect phishing websites using logos. However, the effectiveness of our proposed method will be affected by the input which is fed to the Google image search subprocess. There exists two complications: i) the logo extraction process fails to extract and return the right logo image (note that the logo image actually existed), and ii) the webpage does not contain any logo image. These complications are the reasons for the relatively lower true negative rate (TN_r = 87.0%). Further analysis reveals that 56 out of the 500 legitimate webpage samples (these are the false positive samples) belonged to the first complication. From the experiment, we observed that the first complication contributes to two situations. The first situation happened when our proposed method ineffectively extracted a non-logo image as the logo and removed the actual logo image. Whereas the second situation removes all images (logo and non-logo). In other words, these two situations leave no logo image to be used for the Google image search subprocess. From the 56 webpage samples, half of the samples belong to the first situation and the other half belongs to the second situation. To remedy some of these problems, we crop the top 200 pixels from the full-width screenshot of query webpage. The reason is to obtain the logo within this region. From the experiments, we found that most legitimate websites have their logo located on the left or right part of this region. We acknowledge that this remedy is rather an ad hoc solution, and that the long term solution is to find a better logo extraction approach. The proposed method will apply this remedy when the logo extraction process fails to extract any image. Hence, we only apply this remedy for the second situation (recall that in the second situation, the logo extraction process fails to extract any image). With this remedy, the proposed method managed to reduce the false positive from 56 to 42 webpage samples. Whereas for the second complication, there exists only one out of the 500 webpage samples. This webpage contains only textual content.

Besides these complications, 22 false positive samples came from the identity verification process (Fig. 1). From our observation, we noticed that the logo extraction process had extracted the right logo images, but the Google image search subprocess returned the wrong results. Further analysis showed that the logo images of the 22 false positive samples had at least one of the following properties:

Please cite this article in press as: Kang Leng Chiew, Ee Hung Chang, San Nah Sze, Wei King Tiong, Utilisation of website logo for phishing detection, computers & security (2015), doi: 10.1016/j.cose.2015.07.006

- The image file has less foreground colour and a transparent background.
- The logo has only a textual element (i.e., the brand name). For example, *HAGERTY*.

The lack of visual properties has made the logo become too general and similar to other images (i.e., cliparts). This scenario has caused a Google image search to return unrelated results. In total, the proposed method wrongly classified 65 out of 500 legitimate webpages as phishing webpages ($FP_r = 13.0\%$). Namely, 43 and 22 false positive samples contributed by logo extraction and identity verification processes, respectively. Besides the two properties of the logo image mentioned above, there are other factors which have confused our proposed method. They are:

- The logo existed within a banner image.
- The logo existed within a sprite type of image, which is usually used to optimise webpage loading speed.
- The logo image is in a vector graphic file format (e.g., SVG format).

Besides, there are some logos which may highly resemble other logos as illustrated in Fig. 5. This will cause a Google image search to return an undesired result as well. This is not a surprise as even human cannot effectively differentiate them.

Comparatively, the proposed method performs better in detecting a phishing website ($TP_r = 99.8\%$) than a legitimate website ($TN_r = 87.0\%$). Although the above mentioned complications and issues also happened in detecting phishing websites, the impact is less. The reason is because the proposed identity verification process is based on an identity consistency mechanism. Recall that in Section 3.2, consistent identity means that the real identity (i.e., domain name of the query webpage) is identical to the portrayed identity (i.e., domain name extracted from the Google search results). If the identity is found to be inconsistent, the proposed method will classify the webpage as a phishing webpage. Proving an inconsistent identity is easier than otherwise. Thus, the above mentioned complications and issues which have increased the false positive rate does not affect the same way as in detecting a phishing webpage.

As for the experiment on dataset 3, Table 1 shows the breakdown of the 103 false positive webpages (i.e., $FP_r = 20.6\%$). To simplify the discussion, we used the following abbreviations for different complications (note that the complications are those discussed above for dataset 2):

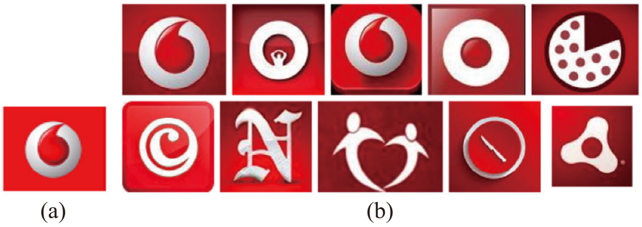


Fig. 5 – Highly similar images. (a) Query image. (b) Similar images returned by Google image search.

Table 1 – Breakdown of different complications for dataset 3.

Complication	Number of webpages	Percentage (%) over 500 webpages
C1	49	9.8
C2	23	4.6
C3	31	6.2
Total	103	20.6

Table 2 – Comparison between dataset 2 and dataset 3 for each complication.

Complication	Dataset 2: Number of webpages	Dataset 3: Number of webpages
C1	42	49
C2	1	23
C3	22	31
Total	65	103

C1: proposed logo extraction process fails to extract and return the right logo image;
C2: the original webpage does not contain any logo image;
C3: proposed logo extraction process had extracted the right logo image, but Google image search subprocess returned the wrong results.

Table 2 shows the comparison between dataset 2 and dataset 3 for each complication. Clearly, we notice that complication C2 has increased the most, while the increments for C1 and C3 are relatively marginal. These results reflect that the performance drop is significantly caused by the complication C2 (i.e., the original webpage which does not contain any logo image). While C1 and C3 also contributed a small portion of the performance drop, it is not critical. The results show that the proposed method can achieve consistent performance even for the very unpopular websites. This can be seen from the low increment for C1 and C3 (C1 increases from 42 to 49 webpages and C3 increases from 22 to 31 webpages). Although at first glance, C2 has a serious increment (increased from 1 to 23), it is expected and non-dangerous. This is because it is very unlikely that phishers will put their efforts onto a less visited website while it will be more lucrative to focus on a popular website. In addition, 17 out of the 23 webpages from C2 are webpages which are very unlikely to be phished. Because those are the webpages that have no login feature, non-financial websites, personal blogs or company informative websites.

With such challenging webpage characteristics constructed for dataset 3, the performance drop is inevitable, and a drop of only 7.6%[4] in the true negative rate is considered moderately good. Low quality logo design usually happened in the unpopular websites, and it is the main reason for the complications C1 and C3. This type of logo is not unique, very small, and highly resembles a text. Besides, when an unpopular website detail is listed in other popular websites, the unpopular website will be listed much later than the popular website

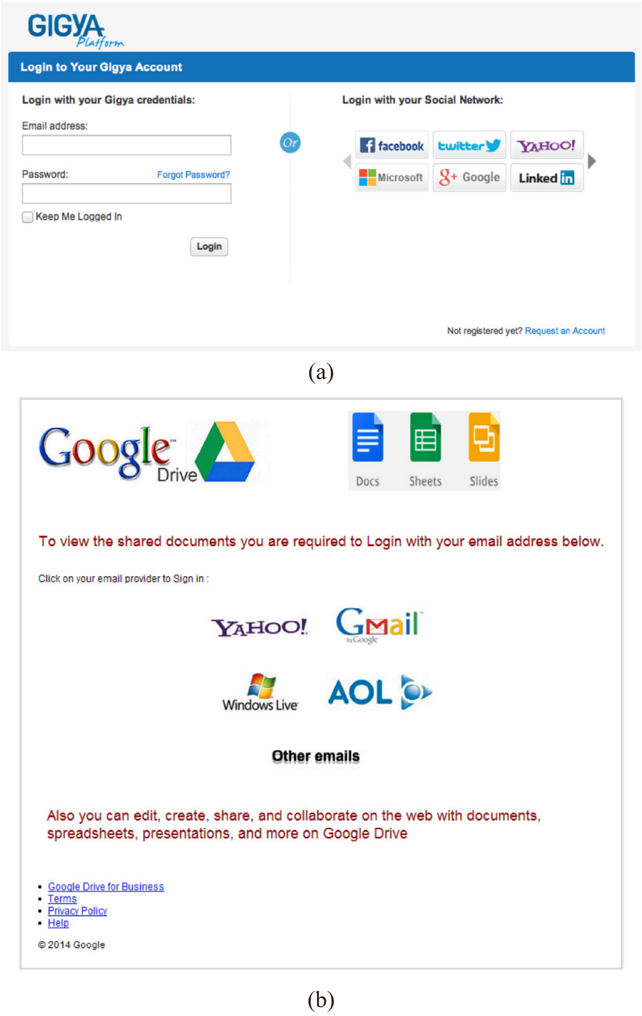
[4] 87.0% – 79.4% = 7.6%, namely the difference of true negative rate between dataset 2 and dataset 3.

in the search result. For example, when the proposed method is examining http://www.arte.tv website, the returned result is the Wikipedia page describing http://www.arte.tv. Obviously, Wikipedia is more popular and ranked higher than http://www.arte.tv, hence the returned result will be Wikipedia. However, we noticed that Google actually managed to return the right keyword in the search results, which is arte. This has motivated us to look into the returned keywords, instead of only using the URLs for future work.

A notable observation is that the proposed method does not certainly fail when a logo image is absent; using non-logo images (images which were ripped off from legitimate website by phisher) still makes it possible to detect the phishing webpage. This is not a surprise, as very often there are images which are unique and representative of the legitimate website. Therefore, a search on these images will lead to the legitimate website.

It is also noteworthy to mention that our proposed method is capable of handling a webpage with multiple logos, which appears to be a difficult problem for an anti-phishing technique using the textual element. For instance, some websites allow users to login with multiple social network IDs as shown in Fig. 6. In this case, the website contains multiple logos on the webpage (i.e., logo of the website and logos of the social networks). The proposed method works because the identity verification process will aggregate the comparison results based on each logo. As discussed in Section 3.2, our method will classify the query website as legitimate if the aggregated comparisons return at least one match.

In order to make the proposed method more effective, we believe that the first complication discussed above can be improved by enhancing the logo extraction process with a more effective logo detection algorithm. For example, using an object segmentation technique to extract logo image from the screenshot of webpage is a suitable approach. As for the second complication, we can integrate with other heuristic approaches such as a TF-IDF technique.



Login to Your Gigya Account

Login with your Gigya credentials:

Email address:

Password: Forgot Password?

□ Keep Me Logged In

Login

Login with your Social Network:

Not registered yet? Request an Account

(a)

To view the shared documents you are required to Login with your email address below.

Click on your email provider to Sign in :

Other emails

Also you can edit, create, share, and collaborate on the web with documents, spreadsheets, presentations, and more on Google Drive

- Google Drive for Business
- Terms
- Privacy Policy
- Help

© 2014 Google

(b)

Fig. 6 – Example of websites which allow users to login with multiple social networks IDs. (a) Legitimate website. (b) Phishing website.

5. Conclusions and future work

In this paper, we propose a heuristic-based approach to detect phishing webpages. We use logo images to determine the identity consistent between the real identity and the portrayed identity of a website. Consistent identity indicates a legitimate website and inconsistent identity indicates a phishing website. The proposed method consists of logo extraction and identity verification processes. The logo extraction process uses a machine learning technique to detect and extract the right logo image. Whereas the identity verification process uses a Google image search to retrieve the portrayed identity, which will be used for the verification. The experimental results show a promising outcome. This justifies the effectiveness and feasibility of using logos to detect phishing websites. We observe that using a graphical element like a logo is more advantageous compared to a textual element when it comes to identity determination.

Clearly, an effective logo extraction process will improve the overall phishing detection accuracy. We will explore an object segmentation approach as our future work. For example, instead of locating the logo image from a pool of downloaded images (image resources of a query webpage), we will capture the screenshot and perform object segmentation directly to extract the logo. This approach has a few advantages. Namely, the captured screenshot is the actual rendered web content, which means there is no other hidden image. Furthermore, this approach can avoid getting a logo within a sprite type of image which is usually used to optimise website loading speed. Using a sprite image as a query will cause a Google image search to return an undesired result even through the logo existed within the sprite image. Another advantage is the logo extraction from the banner image of a website will be more precise. In other words, the extracted logo image will not contain other non-logo images.

Acknowledgments

The funding for this project is made possible through the research grant obtained from UNIMAS and the Ministry of Higher

Please cite this article in press as: Kang Leng Chiew, Ee Hung Chang, San Nah Sze, Wei King Tiong, Utilisation of website logo for phishing detection, computers & security (2015), doi: 10.1016/j.cose.2015.07.006

Education Malaysia under the Fundamental Research Grant Scheme 2/2013 [Grant No: FRGS/ICT07(01)/1057/2013(03)].

We would like to thank Jeffrey Choo Soon Fatt and Colin Tan Choon Lin for their help in the coding of a few modules.

REFERENCES

Abdelhamid N. Multi-label rules for phishing classification. Appl Comput Inform 2015;11(1):29–46.

Abrams R, Barrera O, Pathak J. Browser security comparative analysis – phishing protection. *NSS LABS*, <https://www.nsslabs.com/reports/2013-browser-security-comparative-analysis-phishing-protection>; 2013 [accessed 13.07.14].

Baratis E, Petrakis EGM, Milios EE. Automatic website summarization by image content: a case study with logo and trademark images. IEEE Trans Knowl Data Engin 2008;9(20):1195–204.

Cao Y, Han W, Le Y. Anti-phishing based on automated individual white-list. Proceedings of the 4th Workshop on Digital Identity Management, 51–60, 2008.

Chang C-C, Lin C-J. LIBSVM: a library for support vector machines, Software. <http://www.csie.ntu.edu.tw/~cjlin/libsvm>; 2001 [accessed 05.03.14].

Chang EH, Chiew KL, Sze SN, Tiong WK. Phishing detection via identification of website identity. 2013 International Conference on IT Convergence and Security, 1–4, 2013.

Chu W, Zhu BB, Xue F, Guan X, Cai Z. Protect sensitive sites from phishing attacks using features extractable from inaccessible phishing URLs. Proceedings of IEEE International Conference on Communications, 1990–4, 2013.

Cook DL, Gurbani VK, Daniluk M. Phishwish: a simple and stateless phishing filter. Secur Commun Netw 2009;2(1):29–43.

Dhamija R, Tygar JD, Hearst M. Why phishing works. Proceedings of the 2006 Conference on Human Factors in Computing Systems, 581–90, 2006.

Dunlop M, Groat S, Shelly D. GoldPhish: using images for content-based phishing analysis. International Conference on Internet Monitoring and Protection, 123–8, 2010.

Garera S, Provos N, Chiew M, Rubin AD. A framework for detection and measurement of phishing attacks. Proceedings of the 2007 ACM Workshop on Recurring Malcode, 1–8, 2007.

Hara M, Yamada A, Miyake Y. Visual similarity-based phishing detection without victim site information. IEEE Symposium on Computational Intelligence in Cyber Security, 30–6, 2009.

Huh JH, Kim H. Phishing detection with popular search engines: simple and effective. Foundations and Practice of Security – 4th Canada-France MITACS Workshop, 6888, 194–207, 2012.

Liu W, Ning F, Quan X, Qiu B, Liu G. Discovering phishing target based on semantic link network. Future Gener Comput Syst 2010;26(3):381–8.

Mao J, Li P, Li K, Wei T, Liang Z. BaitAlarm: detecting phishing sites using similarity in fundamental visual features. 5th International Conference on Intelligent Networking and Collaborative Systems, 790–5, 2013.

Maurer M-E, Höfer L. Sophisticated phishers make more spelling mistakes: using URL similarity against phishing. Cyberspace Safety and Security Lecture Notes in Computer Science, 7672, 414–26, 2012.

Mohammad RM, Thabtah F, McCluskey L. Intelligent rule-based phishing websites classification. IET Inform Secur 2014;8(3):153–60.

Nguyen LAT, To BL, Nguyen HK, Nguyen MH. Detecting phishing web sites: a heuristic URL-based approach. 2013 International Conference on Advanced Technologies for Communications, 597–602, 2013.

Otsu N. A threshold selection method from gray-level histograms. IEEE Trans Syst Man Cybern 1979;9(1):62–6.

Prakash P, Kumar M, Kompella RR, Gupta M. PhishNet: predictive blacklisting to detect phishing attacks. INFOCOM 2010 29th IEEE International Conference on Computer Communications, 346–50, 2010.

Schneider F, Provos N, Moll R, Chew M, Rakowski B. Phishing protection design documentation, <https://wiki.mozilla.org/Phishing_Protection:_Design_Documentation>; 2008 [accessed 13.07.14].

Sorio E, Bartoli A, Medvet E. Detection of hidden fraudulent URLs within trusted sites using lexical features. Eighth International Conference on Availability, Reliability and Security, 242–7, 2013.

Verma R, Dyer K. On the character of phishing URLs: accurate and robust statistical learning classifiers. Proceedings of the 5th ACM Conference on Data and Application Security and Privacy, 111–22, 2015.

Xiang G, Hong JI. A hybrid phish detection approach by identity discovery and keywords retrieval. Proceedings of the 18th International Conference on World Wide Web, 571–80, 2009.

Zhang W, Lu H, Xu B, Yang H. Web phishing detection based on page spatial layout similarity. Informatica (Slovenia) 2013;37(3):231–44.

Zhang Y, Hong J, Cranor L. CANTINA: a content-based approach to detecting phishing web sites. Proceedings of the 16th International Conference on World Wide Web, 639–48, 2007.

Kang Leng Chiew is currently a senior lecturer at the Faculty of Computer Science and Information Technology, Universiti Malaysia Sarawak (UNIMAS). He received his PhD in Computer Science specialised in Information Hiding from Macquarie University, Sydney, Australia. His research interest is in information security. He is currently working in anti-phishing research. Past researches include steganalysis on digital images. Previously, he also worked in the area of image processing.

Ee Hung Chang received his BSc degree in Mathematics with Computer Graphics from Universiti Malaysia Sabah. His research interests include cryptography, data compression, anti-phishing and image similarity.

San Nah Sze received her BSc and MSc from University Technology Malaysia, and PhD from Sydney University. She is now a senior lecturer at University Malaysia Sarawak. Her research interests include vehicle routing, manpower planning, shift scheduling, heuristic solution and timetabling.

Wei King Tiong received his BSc and MSc from Universiti Teknologi Malaysia, Malaysia and the PhD degree from Loughborough University, UK. He is a senior lecturer in the Department of Computational Science and Mathematics at Universiti Malaysia Sarawak. His research interests include nonlinear dispersive wave and the Whitham modulation theory.

Please cite this article in press as: Kang Leng Chiew, Ee Hung Chang, San Nah Sze, Wei King Tiong, Utilisation of website logo for phishing detection, computers & security (2015), doi: 10.1016/j.cose.2015.07.006