Expert Systems With Applications 106 (2018) 1–20

Contents lists available at ScienceDirect

Expert Systems With Applications

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

Review

A survey of phishing attacks: Their types, vectors and technical approaches

Kang Leng Chiew, Kelvin Sheng Chek Yong*, Choon Lin Tan

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

ARTICLE INFO

Article history:
Received 29 June 2017
Revised 14 February 2018
Accepted 23 March 2018
Available online 27 March 2018

Keywords:
Information security threats
Advanced phishing techniques
Anti-phishing
Attack vector
Taxonomy
Review

ABSTRACT

Phishing was a threat in the cyber world a couple of decades ago and still is today. It has grown and evolved over the years as phishers are getting creative in planning and executing the attacks. Thus, there is a need for a review of the past and current phishing approaches. A systematic, comprehensive and easy-to-follow review of these approaches is presented here. The relevant mediums and vectors of these approaches are identified for each approach. The medium is the platform which the approaches reside and the vector is the means of propagation utilised by the phisher to deploy the attack. The paper focuses primarily on the detailed discussion of these approaches. The combination of these approaches that the phishers utilised in conducting their phishing attacks is also discussed. This review will give a better understanding of the characteristics of the existing phishing techniques which then acts as a stepping stone to the development of a holistic anti-phishing system. This review creates awareness of these phishing techniques and encourages the practice of phishing prevention among the readers. Furthermore, this review will gear the research direction through the types of phishing, while also allowing the identification of areas where the anti-phishing effort is lacking. This review will benefit not only the developers of anti-phishing techniques but the policy makers as well.

1. Introduction

Phishing has been plaguing the cyber world for over 2 decades, starting in 1995 with America Online (AOL) (James, 2006). The term *phishing* is a variation of the term *fishing* where the act of phishing resembles that of fishing in the following way: the attacker 'lures' the victim using a 'bait' and 'fishes' for personal or confidential information of the victim (James, 2006; Khonji, Iraqi, & Jones, 2013; McFedries, 2006; Purkait, 2012). A comprehensive study on the definition of phishing is conducted by Lastdrager (2014) where he identified a consensual definition of phishing: *'Phishing is a scalable act of deception whereby impersonation is used to obtain information from a target'*. The attacker uses various channels to either deceive the victim directly by a scam or to deliver payload through indirect manner with the goal to obtain personal or confidential information from the victim (Ollmann, 2004).

Phishing attacks have been growing over the years globally with an increase of 65% to the value of 1,220,523 in 2016 as compared to the previous year (APWG, 2017). Also, APWG reported an increase of 5753% of average phishing attacks per month over the period of 12 years, from 2004 to 2016. In 2015, more than half a billion personal records were stolen, an increase compared to the previous year (Symantec, 2016). Kaspersky Lab reported that phishing in the financial sector reached an all-time high in 2016 (Kaspersky, 2017). Between the period of October 2013 to February 2016, the FBI received business e-mail scam reports amounting to total losses of $2.3 billion (McCabe, 2016). This loss is only through business email scams alone and does not include losses through other phishing scams. As the issue of phishing is serious, it is of interest to know in detail the current phishing attack vectors. This information will be valuable in the development of anti-phishing techniques as well as to create public awareness.

In this paper, we present a detailed survey of the phishing techniques and how they work. The interlinks between the medium of phishing, vectors or channels used and the technical approaches applied in the implementation of the phishing operations are discussed. The interlinks are (i) the interlink between the medium of phishing and the vectors and (ii) the interlink between the vectors and the technical approaches. The first interlink shows the elements in a medium which are exploited and used in a phishing attack. This allows the identification of vectors in a communication medium that is currently being exploited. By having the knowledge of the vectors being exploited, the countermeasure that targets the

* Corresponding author.
E-mail addresses: klchiew@unimas.my (K.L. Chiew), kelvin_yong@outlook.com (K.S.C. Yong), colin89lin@gmail.com (C.L. Tan).

https://doi.org/10.1016/j.eswa.2018.03.050
0957-4174/

vector can be developed and coupled with the vector to prevent further exploitation. Furthermore, new and vulnerable vectors can be identified from the medium and preventive measure can be put in place to prevent exploitation of the new vectors. The second interlink gives the knowledge on how the vectors can be used to launch phishing attacks. In a phishing attack, a combination of technical approaches may be used for a better success rate. This combination of technical approaches may utilise the same vector. By knowing such combination of technical approaches, a countermeasure that targets a specific vector to tackle several technical approaches can be developed. Such countermeasures will be more holistic in nature as opposed to ad hoc solutions.

These interlinks are the characteristics of each phishing technique. Thorough understanding of existing phishing techniques is crucial for the development of holistic anti-phishing techniques to counter these phishing operations, as opposed to an ad-hoc solution that is limited and effective only to a specific case. The knowledge of such interlinks is vital to policy makers in introducing policies and guidelines to put a stop to system or infrastructure exploitation for malicious activities. For example, World Wide Web Consortium (W3C)(W3C, 2018) spearheads the development of web standards. One of W3C's vision is the Web of Trust, which is to recognise the need for trusted interaction between people on the Web and support this vision through the development of the protocols and guidelines for the long-term growth of the Web. These protocols and guidelines include the semantic web, XML security, web of services security, and privacy. Furthermore, such review will be useful for readers from every walk of life as well, enabling them to take precautionary and preventive actions against phishing attacks. With such actions by the general public and the availability of the anti-phishing system, the effectiveness of a phishing attack can be reduced significantly.

The remainder of this paper is segmented into four sections. Section 2 discusses the existing literature on the currently available methods of phishing, which motivates the provision of a more comprehensive and systematic review of these phishing methods. Section 3 classifies the phishing technical approaches into their respective mediums and vectors of propagation and provides a detailed explanation of their operations. Section 4 discusses the existing combination of these phishing technical approaches that complement each other, forming a more advanced phishing attack. The final section, Section 5, concludes the paper.

2. Related literature

There are several phishing-related review papers available currently (Almomani, Gupta, Atawneh, Meulenberg, & Almomani, 2013; Mohammad, Thabtah, & McCluskey, 2015; Patil & Patil, 2015; Rader & Rahman, 2013; Sahoo, Liu, & Hoi, 2017; Singh, 2007).

Mohammad et al. (2015) give a brief description of strategies employed by phishers. These strategies are categorised into three groups, namely mimicking attack, forward attack, and pop-up attack. In mimicking attack, the phishers lure their victims to reveal their personal information through email mimicking an official email from a legitimate organisation. Forward attack and pop-up attacks involve the use of man-in-the-middle (MITM) methods where the phisher intercepts and retrieves the victim's personal information through a proxy website (forward attack) or a pop-up window (pop-up attack). The authors also discuss the lifecycle of the phishing websites as shown in Fig. 1. However, the emphasis of (Mohammad et al., 2015) is not on the phishing techniques; rather, it is on the anti-phishing techniques where a detailed review of the methodologies of various anti-phishing papers is given.

Singh (2007) highlights the innovations of phishing techniques in the banking sector. Techniques are classified into four methods, namely dragnet method, rod-and-reel method, lobsterpot method and Gillnet phishing. The dragnet method is the use of email, website, or pop-up windows that contain an identity element of a legitimate organisation such as logos, corporate names, and trademarks, prompting for immediate action. The rod-and-reel method involves identifying prospective victims through initial contact and targeting them using false information in hopes of having the victims disclose their personal information. The lobsterpot method is the use of a spoofed website, mimicking a legitimate one, which tricks the victims to surrender their personal information. Gillnet phishing utilises malicious code residing in emails and websites to infect the machines of the victims. Unauthorised activities such as spying for the victim's personal information or changing the settings in the machine is performed by this malicious code. The author's main approach is through the case study of phishing attacks in various countries, focusing on the impact of the phishing attacks. Finally, the author lists a number of approaches to combat these phishing attacks in the banking sector.

Rader and Rahman (2013) discuss the current and emerging phishing attack vectors. *Modus operandi* and suitable examples are given for each phishing attack vector in the discussion. The vectors are categorised into classical and emerging attack vectors. The authors also include a brief discussion on the risk reduction and mitigation of these attack vectors.

Sahoo et al. (2017) provide a brief description of the popular types of malicious attacks, not limited to phishing only. They also include a brief discussion on types of machine learning, types of feature for machine learning and approaches for detecting malicious Uniform Resource Locators (URLs). The focus of the paper is on the identification of the features used for classifying malicious websites in literature, noting the design and limitations of some of these features. The features are grouped into five categories. Discussion on the algorithms of the machine learning along with their examples is given as well.

Patil and Patil (2015) discuss the attack vectors specifically in malicious web pages. These attack vectors are not limited to phishing attacks only. They also discuss the various malicious web page detection tools from the literature, highlighting the features and machine learning algorithms used in these tools.

Almomani et al. (2013) review the subject of phishing attacks through email. They give a brief overview of phishing emails, highlighting the types of phishing attacks and the lifecycle of phishing emails, as well as phishing email classification and evaluation methods. A detailed discussion of the various features used in phishing email detection is also provided, as well as a classification of these features into three groups, namely basic features, latent topic model features and dynamic Markov chain features. This is followed by further discussion of the protection approaches against phishing attacks, including their advantages and disadvantages.

Ollmann (2004) presents a comprehensive paper on phishing attacks and the defence mechanisms used to counter these attacks. He gives a brief discussion on the history of phishing before detailing the threat of phishing. Three areas of phishing threat are discussed by the author: social engineering factors, delivery mechanisms of phishing messages and phishing attack vectors. He also discusses the defence mechanism in place in three locations, namely client-side, server-side and enterprise level. Although the paper was published more than 10 years ago, the author relays fundamental information that remains relevant to modern-day discussions of phishing attacks.

Phishing attacks started out as a scam through social engineering. The phisher took on the disguise of legitimate personnel to convince the victim to give out their password as in the case of AOL account phishing back in 1995 (James, 2006). Since then, the phishing techniques have evolved with a more advanced approach. From the literature review, it is apparent that researchers are more focused on the discussion of anti-phishing as opposed to phishing.

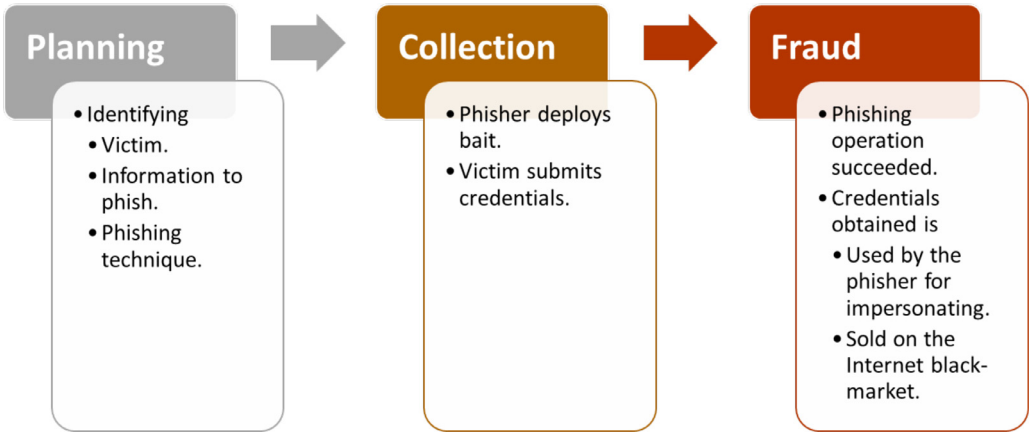

Fig. 1. Lifecycle of a phishing website (Mohammad et al., 2015).

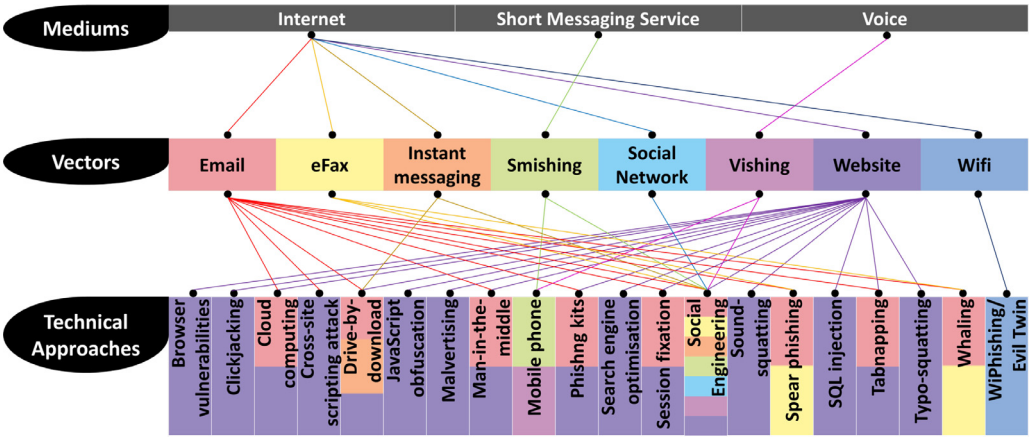

Fig. 2. The interlink between the medium, vector and technical approach of the phishing techniques. The colours of the associated vectors are given as the background colour of the technical approaches for easy identification.

The lack of a detailed and systematic discussion on the phishing techniques has created a gap between anti-phishing and phishing development. Thus, we will focus on classifying the phishing attacks according to their mediums, vectors and technical approaches used. This paper will shed light on the fundamental operation of various phishing techniques. Such knowledge is crucial for the development of effective anti-phishing techniques, hence its importance cannot be overemphasised.

3. Phishing components

The phishing techniques consist of three components which are the medium of phishing, the vector to transmit the attack and technical approaches used during the attack. The interlink between the mediums of phishing, the vectors and the technical approaches used is shown in Fig. 2.

The medium of phishing is the base means of conveying the phishing attacks to the victims. There are three bases commonly used which are internet, voice and short messaging service (SMS). The availability of the internet has opened a huge opportunity for the phishers to reach their victims easily.

The vector of channel deployed by the phishers based on the medium is the vehicle in place for launching the phishing attacks. Email, eFax, websites, instant messaging and social networks are the vectors that are accessible to the phishers via the internet. Vishing is the phishing vector by voice and smishing is the phishing vector by short message service (SMS).

Technical approaches are the technical means that are used on top of the social engineering phishing to further enhance the effectiveness of phishing. Currently, there are many approaches used by the phisher, namely drive-by-download, man-in-the-middle (MITM), cross-site scripting (XSS) attack, tabnapping, spear phishing, whaling, search engine optimisation (SEO), session fixation, malvertising, social engineering, JavaScript obfuscation, browser vulnerabilities, mobile phone, cloud computing and WiPhishing or Evil Twins. There are also some technical approaches that bypass the need of deceiving the victim or social engineering, which are SQL injection, typo-squatting, and sound-squatting. Also, phishing kits fall under technical approaches, but they are a tool to assist in deploying phishing attacks and not as a phishing attack on their own.

3.1. Mediums of phishing

All phishing attacks require interaction from the victim-to-be-phished and medium is required for interaction to happen. Phishers are able to approach their potential victims through these three mediums: internet, voice and short messaging service.

Starting with the first prototype of the Internet called Advanced Research Project Agency Network (ARPANET) (Leiner et al., 2009) and evolving to the current Internet that we have currently, this invention opens up great potential in information dissemination, collaboration, and interaction between people instantly and regardless of place (Leiner et al., 2009). However, this convenience of interaction between people also enables phishers to approach their potential victim easily.

Voice as a medium of communication has been utilised by humans since the beginning of mankind. As the oldest and one of the

most effective means for humans to interact with each other, it is also being used by phishers to trick their victims into giving out their personal information.

Short messaging service (SMS) is a concept developed for telephony by Friedhelm Hillebrand and Bernard Ghillebaert in 1984 (Hillebrand, 2010). Through this service, individuals are able to exchange short text messages on their mobile devices. Again, such convenience allows phishers to interact with their victims in an attempt to steal personal information.

The vectors or channels that utilise these mediums are discussed in the next section.

3.2. Vectors of phishing

There are several vectors associated with the mediums discussed in the previous section. Most of these vectors that will be discussed are associated with the Internet. Thus, the Internet is the popular choice of medium for phishers.

Email, eFax, instant messaging, social networks and websites are vectors affiliated with the Internet. The earliest reported phishing attack on the AOL account is through instant messaging (James, 2006).

Electronic mail or email is a message sent from a computer to one or more recipients electronically via a network. This allows fast communication between people regardless of geographical location compared to the conventional mail which is now known as snail mail. This vector of phishing is popular among the phishers (Ollmann, 2004). A phisher crafts the email to trick the victim into believing that the email came from a legitimate entity and the victim is required perform a certain task, resulting in the victim's personal information being compromised.

Internet fax, eFax or online fax differs from traditional fax in that the fax is sent via the Internet Protocol (IP) instead of through the phone network. Through eFax service providers such as efax.com (j2 Global, 2017), users are able to send and receive faxes by email. The benefits of eFax over traditional fax are that users are able to view their incoming fax online and send a fax to a recipient's fax machine online, without the need for a fax machine. However, with the online capability of this eFax feature, phishers are able to deploy the same tactics as in an email to phish for their victim's personal information.

Instant messaging is the earliest recorded phishing attack on AOL users (James, 2006). The phishers disguised themselves as an AOL administrator and contacted the potential victim via Internet Relay Chat (IRC), the AOL's messaging alert system. A message is sent to the victim informing that there is a problem with the billing and the victim is requested to provide his or her credit card and login details to update the information in the billing system. Instant messaging is an online and real-time chat which allows users to communicate through their computer or mobile devices. The term instant messaging (IM) is used in the 1990s with IM clients such as ICQ, AIM, Yahoo! Messenger, Pingin and MSN Messenger entering the instant messaging arena. Today, IM has been integrated into social media such as Facebook Chat and Twitter and it is no longer restricted to just text-based communication. Users are able to send emoji, photos, GIFs, maps, hyperlinks, files and video clips. Nowadays, users are able to perform video and voice call using their IM client. As IM is much more popular than SMS (Marius, 2016), it is not surprising that phishers will jump into this wagon and use IM as a phishing vector.

In the 2000s, social media started gaining popularity as a means for connectivity with other people (Shah, 2016). This area of social media is often referred to as social networking. Examples of such are the Facebook, Twitter and Google+ platforms which allow people with the same interest, circle or affiliation to connect with each other. In addition, there are other social media platforms like Tumblr, Pinterest, Spotify and Foursquare that cater to specific niches. With the benefit of fast sharing of information in social media, it can also be exploited by phishers to deploy phishing attacks.

Websites are another popular vector for phishers to deploy their baits. It has become commonplace for users to submit their personal information such as login credentials in order to access a particular service. Hence, phishers take advantage of this vector in tricking their victims to submit the personal information as the victims usually do in a legitimate website. Furthermore, a survey shows that the internet users will normally assume that phishing attacks are mostly conducted through email and less through web pages (Jakobsson, 2007). Thus, the internet users will have lower guard against phishing website than phishing email.

Besides the use of the internet as a medium of phishing, there are two other mediums of phishing which are SMS and voice. These two mediums led to phishing vectors of smishing and vishing respectively. Smishing and vishing are the migration from the email phishing vector (Rader & Rahman, 2013). Phishers contact their victim through text message (in smishing) or by call (in vishing) to the victim's phone to lure the victim into giving out his or her personal information.

Detailed discussion on the phishing attacks through these vectors is given in the next section.

3.3. Technical approaches

There are several technical approaches utilising one or more vectors mentioned in the previous section that are deployed by the phishers to obtain personal information from their victims. These technical approaches are given below:

3.3.1. Browser vulnerabilities

Browser, like any commercial software, is subjected to vulnerabilities, which can be exploited by the phisher to launch a phishing attack on the user. With every addition of new features and functionality to the browser, there is a possibility of introducing vulnerability to the software (Ollmann, 2004). Furthermore, the ability to install add-ons and plug-ins from third-party providers has led to more vulnerability in the browser. Such phishing attacks through browser vulnerability are harder to detect and prevent. Examples of a few discovered browser vulnerabilities are given below:

In 2003, a vulnerability in Microsoft Internet Explorer allowed phishers to obfuscate the URL of the phishing website as a legitimate one by including a 0x01 character after the "@" character (Australian Computer Emergency Response Team (AusCERT), 2003; Ollmann, 2004; SecurityFocus, 2003; SecurityTracker, 2003). The rest of the URL after the 0x01 character would not be shown in the address bar, hiding the true domain name of the URL. The sample code below shows the use of the 0x01 character to hide the real location of the page (phishingsite.com). Microsoft released a fix for this vulnerability (SecurityTracker, 2004).

```
<HTML>
  <BODY>
    <A HREF="http://www.microsoft.com%01@
     phishing.com/as001/mypage.htm">Microsoft
    </A>
  </BODY>
</HTML>
```

Another example of vulnerability is the *window.createPopup()* method (Dormann & Manion, 2004; Milletary, 2013) that affected Microsoft Internet Explorer in 2004. This method creates a borderless pop-up window (without chrome and window manager decorations) and can be placed on top of all windows and anywhere

on the screen. The phishers can use this borderless pop-up window to mask the URL of the phishing website or display the HTTPS padlock icon to deceive the victim into thinking that it is a secure website. Microsoft released a fix for this vulnerability through the Windows XP Service Pack 2 (Dormann & Manion, 2004).

Cross-domain vulnerability (Dormann, 2005; Mitnick & Simon, 2002) is another vulnerability that existed in Microsoft Internet Explorer in 2005. The phisher is able to use the DHTML Edit ActiveX control when the victim views a malicious website to gain access to the web control of another domain. Then, the phisher is able to replace and load malicious content on a trusted site. Microsoft released a fix through an update to rectify this vulnerability (Dormann, 2005).

Plug-ins to a browser can cause security loopholes for phishers to exploit. Vulnerability contributed by plug-ins has doubled in 2015 from 2014 with Adobe plug-ins such as Adobe Flash Player being the major contributor (Symantec, 2016). The solution to such plug-in vulnerabilities is through constant updates with patching or discontinuation of support for certain plug-ins, such as in the case of Chrome browser which no longer natively supports Flash (LaForge, 2016).

Phishers may use the auto-fill function available in most browsers to phish for the victim's personal information (Kuosmanen, 2017). Auto-fill is a function that allows a user to quickly fill a form in the webpage automatically based on the stored information about the user. Phishers may design a form that seems to request limited information such as name and email address only. However, the phishers may add in form fields which are not visible to the victim. Upon using the auto-fill function, these hidden form fields are automatically filled. Thus, the victim is unaware of submitting sensitive personal information, such as his or her address, contact number, password, etc. to the phisher.

The vulnerabilities of the browser are related to the website vector where the vulnerabilities may be in the form of URL obfuscation of a phishing website, altering the webpage of a legitimate website or abuse of plug-ins or functions when visiting a website.

3.3.2. *Clickjacking*

Clickjacking (Patil & Patil, 2015), also known as user interface (UI) redressing attack (Akhawe, He, Li, Moazzezi, & Song, 2014), is the manipulation of the UI of a webpage that leads to the user performing an action unknowingly when interacting with the compromised UI. Examples of action that the user may be tricked to perform are liking a page on Facebook (known as Likejacking Yadav & Nagpal, 2016), performing a checkout at PayPal, allowing access to a webcam and microphone, stealing personal information (Huang, Moshchuk, Wang, Schecter, & Jackson, 2012) or posting a tweet (Mahemoff, 2009).

Huang et al. (2012) described that the attacks can be carried out by compromising the integrity of either one of the three areas which are the target display, pointer and temporal.

Compromising the target display (Akhawe et al., 2014; Huang et al., 2012) can be done by fully overlaying the target website with a div container, for example, to hide the target website. Any clicks performed on top of the decoy overlay will actually land on the target website instead of on the decoy overlay as shown in Fig. 3. Instead of the full overlay, a partial overlay can be used too. The partial overlay will hide the actual information and replace it with the false copy while leaving certain elements on the target website visible to the user. For example, the attacker placed an overlay over the PayPal checkout webpage with the recipient and purchase details but leaving the "Pay" button uncovered. The user sees that the details are correct and has no knowledge that the actual information is hidden from him or her. The user then performs the purchase by clicking on the button. The user is unaware that the actual recipient of the purchase is the phisher.

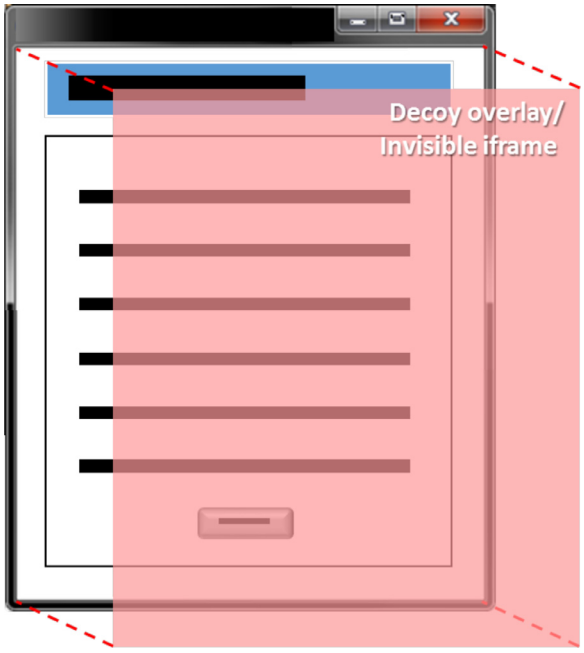

Fig. 3. The UI redressing attack by compromising the target display integrity.

Fig. 4. The UI redressing attack by compromising the pointer integrity.

Instead of deploying the attack using an opaque decoy overlay as mentioned in the previous paragraph, the attacker can alternatively place an invisible iframe overlaying a target element, such as a button (Patil & Patil, 2015). When the user attempts to click on the button, he or she will be clicking on the invisible iframe instead. The user's click is hijacked into performing a different action than intended.

Another method deployed by the attacker is by compromising the pointer integrity (Akhawe et al., 2014; Huang et al., 2012). The attacker introduces a fake pointer in place of the actual pointer at another location as shown in Fig. 4. This will give the user the wrong perception of the actual location of the pointer and lead the user to click on the element of which is not the original intent. Such an attack is also called cursorjacking.

Instead of using a fake pointer, the attacker can position an invisible iframe under the pointer and move along with the pointer (Stone, 2010). No matter where the user clicks, the user will be

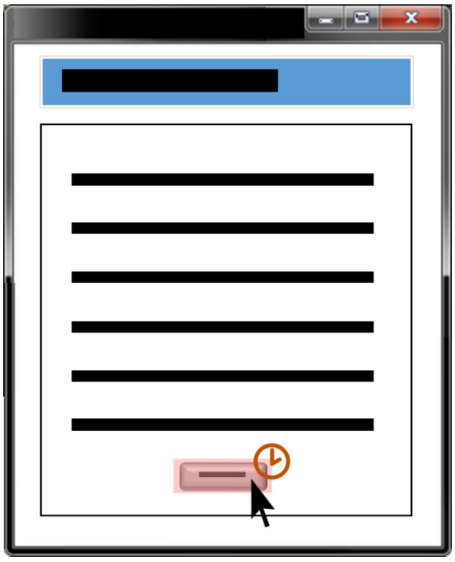

Fig. 5. The UI redressing attack by compromising the temporal integrity.

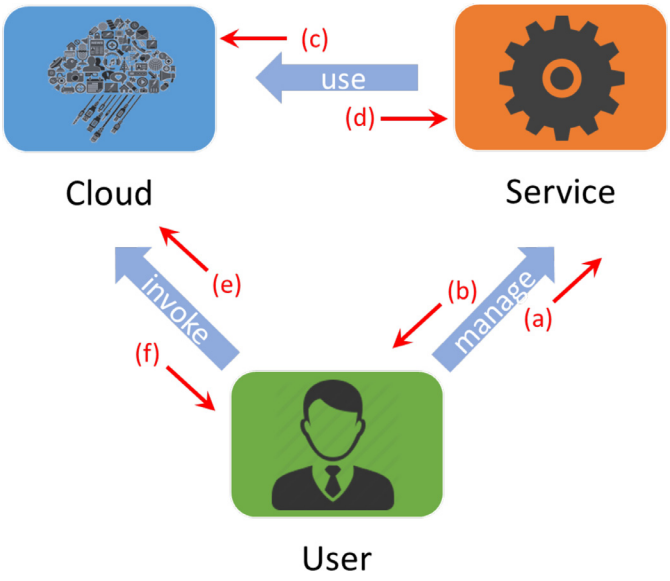

Fig. 6. The relation of the three components in cloud computing, along with the attack surfaces in red (Gruschka & Jensen, 2010). The attack surfaces are (a) service-to-user, (b) user-to-service, (c) cloud-to-service, (d) service-to-cloud, (e) cloud-to-user and (f) user-to-cloud. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

clicking on the invisible iframe instead and performs an action defined by the iframe without the user's knowledge.

The third method that the attacker can use to execute a clickjacking attack is through compromising the temporal integrity (Akhawe et al., 2014; Huang et al., 2012). Here, the attacker takes advantage of the slow response of users in performing a click during certain events. For example, when the user hovers the pointer over an element and clicks or when the user performs a double-click. The attacker exploits the time taken by the user to click after pointing the cursor at the element or the time to click for the second time in a double-click. The attacker inserts a legitimate element, such as the PayPal "Pay" button on top of the decoy button that the user intended to click just before the user performs the click (as shown in Fig. 5).

3.3.3. Cloud computing

Cloud computing is an online service that encompasses three service models which are Software-as-a-Service (SaaS), Platform-as-a-Service (PaaS) and Infrastructure-as-a-Service (IaaS). The adoption of cloud service is increasing with cloud providers adding more services (Weins, 2016). Security of cloud service becomes the top second issue of concern by cloud service users, after lack of resources or expertise (RightScale, 2016).

With the increasing popularity of cloud service adoption and the richness of information in one location, (Symantec, 2016) predicted cloud service will be the target of phishers, exploiting the vulnerability of cloud computing. As cloud service providers rely on a user's email address as account credential, phishers may target the cloud provider to retrieve the user's credentials and use these credentials to access other accounts by using password reuse attack (PhishLabs, 2017). The intrusion into the cloud service can happen through the client or provider (Nagunwa, 2014).

Gruschka and Jensen (2010) identified 6 attack surfaces of cloud computing services. There are three components in cloud computing, which are the users, the service, and the cloud provider. The relation of these three components is shown in Fig. 6. The service instances provide a dedicated interface to the user which allows the user to use the cloud services depending on the service model type, being SaaS, PaaS or IaaS. The user is also able to perform cloud control directly to the cloud to add or delete service instances.

An example of a phishing attack on cloud computing is the incident that targeted Dropbox in 2012 (Gibbs, 2016). The password of a Dropbox employee was compromised through password reuse in the employee's LinkedIn account. Using this password, the phisher gained access to the company's network and then, to the user database. Fortunately, the passwords in the database were encrypted and "salted". Such a phishing attack falls under the attack surface (e) where phisher (user) logged into the cloud provider system directly to access the user database.

Cloud provider vulnerability can be exploited by phishers, like in the case involving Amazon Elastic Cloud Computing (EC2) in 2008 (Gruschka & Iacono, 2009). Amazon EC2 is an IaaS, providing virtual servers to users. It uses a SOAP interface to control the deployed machine, whether to start a new instance or to terminate one. There is a vulnerability in the interface allowing the attacker to modify an eavesdropped message even though the message is digitally signed (McIntosh & Austel, 2005). Such vulnerability allows the attacker to perform malicious commands on behalf of the user and create a botnet. This form of attack also falls under the attack surface (e).

Another example of cloud provider vulnerability is Google Docs (Barkah, 2009; Chow et al., 2009). There was no protection in place for embedded images where even after the deletion of the document which has the embedded image, the image is still accessible. Furthermore, the ID of the resource owner of the image is embedded in the URL of the image. This is a form of Insecure Direct Object Reference, contributing to privacy implication. The person with whom a document is shared has access to the previous revision of the document. This leads to another privacy implication in the case where redaction is performed on the document, but the person is able to access the version before the redaction is performed. Finally, a glitch in the Google Docs user interface implementation may cause the invitation setting to a privacy document to be wrongly set as public, allowing deleted share participants of the document to add themselves back into the document sharing without the consent or awareness of the document's owner.

Information in the cloud database that has been compromised such as email addresses can be used to conduct more phishing attacks as in the case of DocuSign (Henderson, 2017). Email addresses of the DocuSign customers were stolen and the phishers conducted a follow-up phishing attack on these customers by

Table 1
Examples of custom URL XSS attacks (Ollmann, 2004).

Custom URLs	Example
Full HTML substitution	http://legit.com/login?URL=**http://phish.com/login/fake.htm**
Inline embedded scripting content	http://legit.com/login?page=1&client=**<SCRIPT>phishcode...**
Load external scripting code	http://legit.com/login?page=1&response=**phish.com%21phishcode.js...**

sending to these email addresses an email purporting to be from DocuSign with a malware infected Word document.

3.3.4. Cross-site scripting (XSS) attack

Cross-site scripting (XSS) is a vulnerability exploit of a website which allows the phishers to inject malicious code into data fields or make use of custom URL to a website (DigiCert, 2009; Ollmann, 2004). Such vulnerabilities arise from a poorly constructed website that fails to filter out external supplied content which can be in the form of malicious script (Emigh, 2005; Rader & Rahman, 2013). XSS attack is a method to circumvent the *Same-Origin Policy (SOP)* (Ruderman, 2016). SOP is in place to prevent scripts from interacting and accessing information of another origin. Thus, it prevents scripts from malicious websites to access personal information such as login credentials when the user is visiting another website.

Examples of XSS attack through custom URL is given in Table 1 (Ollmann, 2004). In full HTML substitution, the victim visited a legitimate website but an element of the legitimate page such as the login form is referred from the malicious website instead of from the legitimate ones. As for inline, embedded scripting content, the script is embedded in the URL and it is run upon accessing the URL. The script is loaded from an external source as in the case of the third example.

JavaScript is commonly used in XSS attack (Kals, Kirda, Kruegel, & Jovanovic, 2006; Nagunwa, 2014). Through this script, the phisher is able to display a fake form for the victim to enter his or her personal information such as login information as shown in Fig. 7, thereby unknowingly submitting this information to the phisher instead of to the legitimate entity (Elledge, 2007). The phisher can also redirect the victim to a phishing website while the victim still thinks that he or she is visiting a legitimate website (Milletary, 2013). An example of such URL redirection is in the box below:

```
index.php?name=<script>window.onload = function() var link=document.getElementsByTagName("a");
link[0].href="http://phish.com/";</script>
```

Here, the victim thinks he or she is visiting the webpage of *index.php* by inspecting the URL but unknowingly is being redirected to phish.com.

Kals et al. (2006) classified XSS attacks into two forms which are reflected XSS and stored XSS. Patil and Patil (2015) also classified XSS attacks into two types, namely non-persistent XSS attack and persistent XSS attack. In reflected XSS or non-persistent attack, the victim is phished by visiting a custom URL by the phisher as in the examples above. As for stored XSS or persistent XSS attack, malicious script is injected and stored in an application's database. Hence, the script will remain in the database and is executed as that script is being invoked, causing on-going damage until this script is removed. Example of stored XSS or persistent XSS attack is to store the script as a post on a message board. A message board that does not perform input validation and filtering will allow such script to be stored and executed every time a visitor visits that website.

Another way to circumvent the SOP is by tricking the user into divulging his or her personal information. Gelernter and Herzberg (2016) reported on the use of CAPTCHA in a phishing attack. The attack procedure is shown in Fig. 8. On the set of a CAPTCHA attack, the victim visited a malicious website and was also signed in or submitted personal information to a legitimate website in (1). The legitimate website is deemed as target website here. The malicious website loads some of the personal information from the target website using an inline frame (iframe) and utilises cascaded style sheet (CSS) method to display this private information as a CAPTCHA in the malicious website in (2). In (3), the user, unaware that the CAPTCHA is showing his or her private information, completed the CAPTCHA and submitted this information to the phisher. The phisher will need to present the information to the victim in an unrecognisable manner for this kind of attack to succeed.

3.3.5. Drive-by-download

Drive-by-download is a delivery technique to inject a machine with malware, virus or shell code by just visiting a website (Cova, Kruegel, & Vigna, 2010; Patil & Patil, 2015) or viewing an HTML email (Patil & Patil, 2015). Drive-by-download can also be deployed through Internet Relay Chat (IRC) sites (Elledge, 2007). The malicious code is commonly written in JavaScript to target the vulnerability of a browser or a browser's plug-in and is hosted either in a server or injected into a website or an HTML email.

The malware used in phishing attack can come in the form of a Trojan and spyware (Banday & Qadri, 2007; Chaudhry, Chaudhry, & Rittenhouse, 2016; Elledge, 2007; Emigh, 2005; Milletary, 2013; Nagunwa, 2014; Suganya, 2016) or a bot (Chaudhry et al., 2016; Cova et al., 2010; Elledge, 2007; Milletary, 2013; Nagunwa, 2014; Rader & Rahman, 2013; Schiller & Binkley, 2007; Symantec, 2016).

Trojan got its name from an ancient Greek story where Greeks performed stealth infiltration into their enemy's base with a wooden horse as a disguise for an army vessel. Similarly, Trojan is a malicious program disguised as legitimate software that tricks its victim to install or run it. It will then gain access to the victim's machine (Landwehr, Bull, McDermott, & Choi, 1994). Trojan can be utilised by the phishers to install keystroke loggers to record the victim's personal information or screen capture to capture the screen of the system with the victim's personal information on it (Elledge, 2007). An example of keystroke loggers Trojan is the TrojanSpy:MSIL/Omaneat (Spring, 2017) which can record the keystroke, monitor opened applications and track browsing history of a Windows user. Linux.Ekoms.1 (Kovacs, 2016) is an example of a Trojan that can perform a screen capture on a Linux machine. This malware can also come in the form of web Trojan (Banday & Qadri, 2007; Chaudhry et al., 2016; Emigh, 2005; Suganya, 2016). When a victim visits a legitimate login website, a pop up will appear and request the victim to log in through the pop-up windows instead. Thinking that the pop-up is from a legitimate source, the victim unknowingly submits his or her personal information to the phisher.

Bot is a malware that gives the phisher access to the victim's machine, allowing the phisher to command and control (C&C) the victim's machine remotely (Milletary, 2013; Schiller & Binkley, 2007). A machine under C&C is called a botnet. The aim of the phisher using this technique is to infect as many machines as possible, gaining botnets under his or her command. The phisher can then utilise these botnets to perform either of the following: (Milletary, 2013):

- to send phishing emails,
- act as web servers to host phishing websites or to distribute malware,
- act as redirectors to phishing websites, or
- act as proxy services.

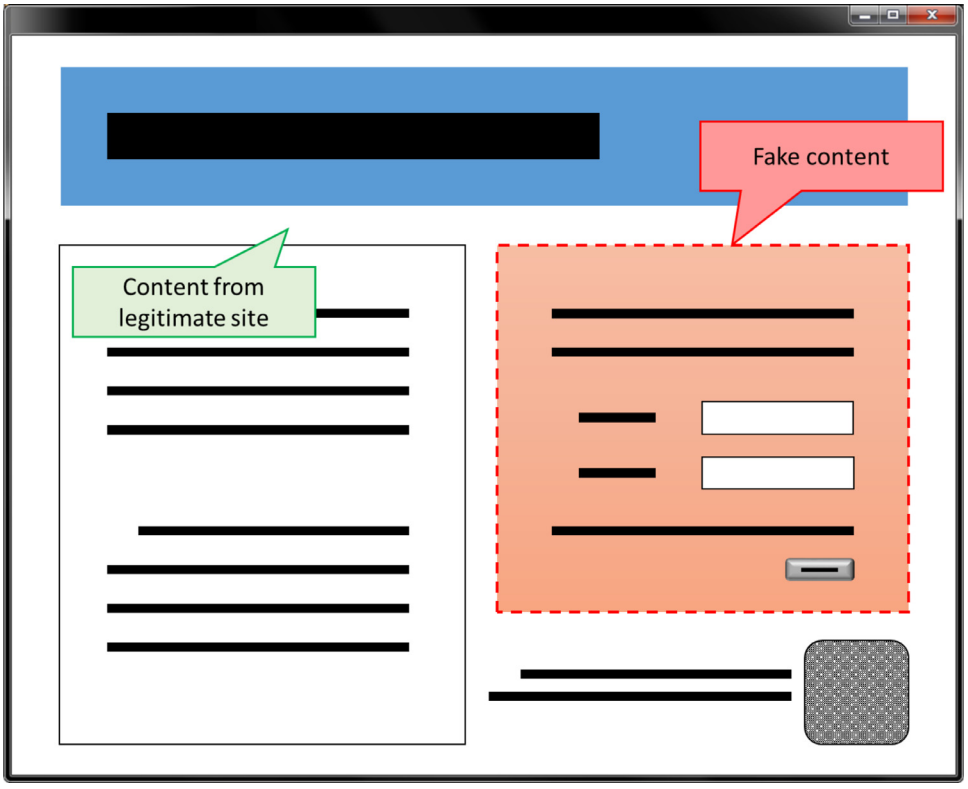

Fig. 7. An example of cross-site script attack on a legitimate website.

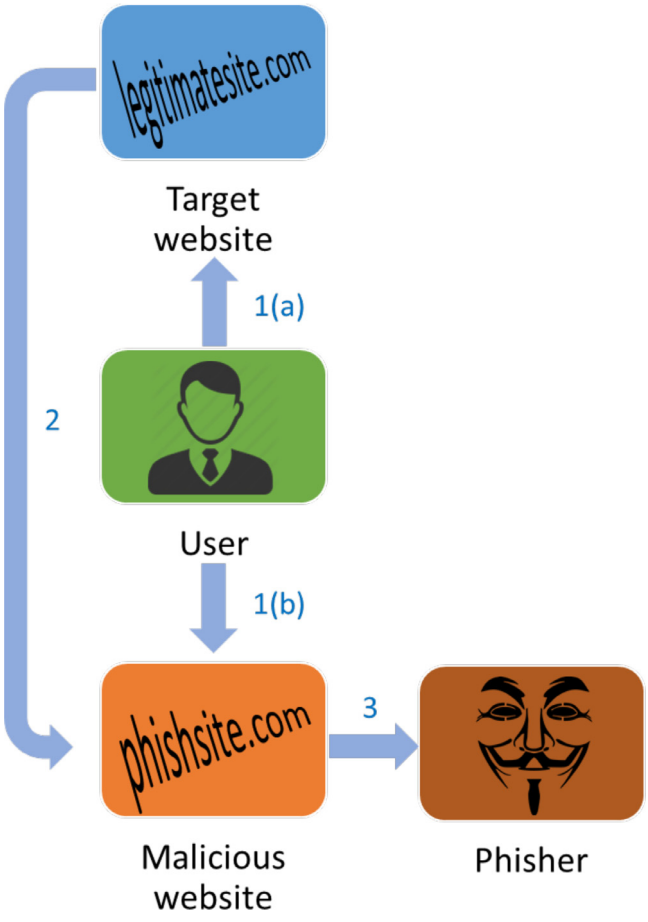

Fig. 8. The CAPTCHA attack.

The lifecycle of a botnet (Schiller & Binkley, 2007) is shown in Fig. 9. Upon infection of a machine and turning a machine into a botnet, it will inform the C&C server of its existence. Then, it will cripple the defence in place such as disabling the anti-virus and hiding from the operating system and other detection systems to avoid detection. The botnet will then wait and listen for the command from a C&C server or other peers. After the issue of command, it will retrieve the payload module which will define the purpose or function of the botnet. It will then execute the command based on the defined function of the botnet. Finally, it will report back to the C&C server upon the completion of the task at hand. The botnet can also be instructed to erase all evidence and terminate its service.

Another function of the botnet to be highlighted here is that it can be used in fast-flux attacks (RSA, 2016; Zhou, 2015). Fast-flux refers to a domain name having different IP addresses that is commonly being used legitimately by an organisation to balance the loads among several servers and also for improved reliability of their services. However, this technique is also being used by phishers (Borgaonkar, 2010; Hao, Feamster, & Pandrangi, 2011) to hide their websites in the maze of a fast-flux service network and also to improve the longevity of their attacks. Botnets are used as proxies to construct this fast-flux service while being in the front line of attack. This way, the C&C server–the puppeteer behind this orchestra – remains hidden behind this fast-flux service network. Also, shutting down some of these botnets will not stop the attacks as the phishing attack can still continue through other botnets in the fast-flux service network. The phisher will only need to recruit more botnets to ensure the longevity of the phishing domain name.

Malware loaded onto a victim's machine through drive-by-download can perform session hijacking (Banday & Qadri, 2007; Chaudhry et al., 2016; Emigh, 2005; Suganya, 2016). Such attacks involve monitoring of the victim's online activity. Once the victim

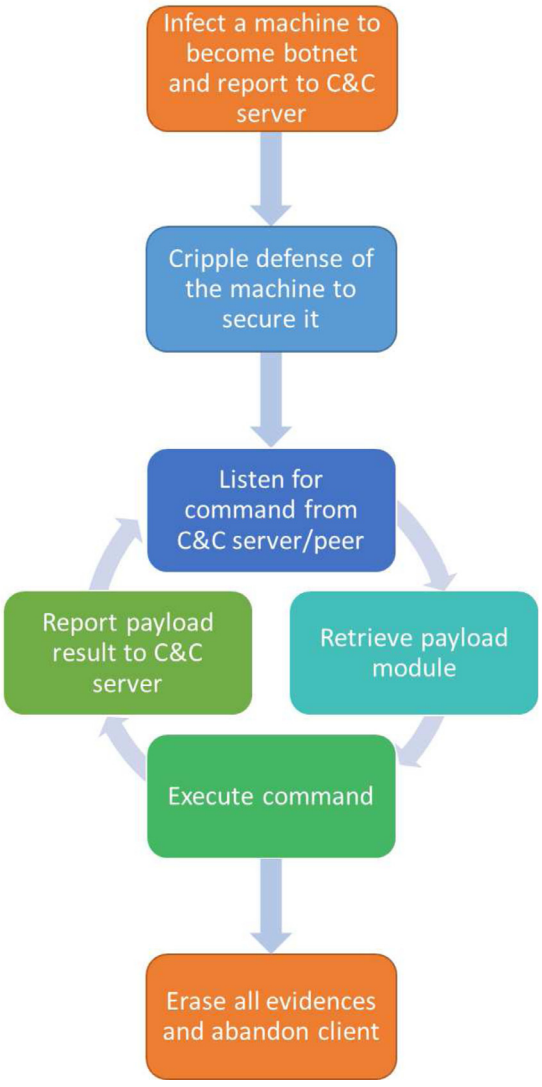

Fig. 9. The lifecycle of a botnet (Schiller & Binkley, 2007).

authenticates into a secure session, the phisher hijacks this session using the malware and performs a transaction using this session without the victim's knowledge.

Other than drive-by-download, malware infection of a machine can also happen by opening a malware-infected attachment in an email or installing a malware infected software, which is normally distributed as a free software (DigiCert, 2009).

3.3.6. Javascript obfuscation

JavaScript is used by phishers to mask or obfuscate the chrome area of the web browser windows (Elledge, 2007). The chrome area consists of the address bar, status bar, toolbar and menu area which is the area surrounding the content area. Using JavaScript, the phisher can spoof the address bar or the status bar to make the URL appear legitimate. Also, the phisher is able to spoof the "https" and the "lock" icon to make the phishing website appear as a secure website. The victim will not be aware that the website he or she is visiting is a phishing version of a legitimate website. A visual check on the URL in the address bar and the information in the status bar will not flag any suspicion. The delivery of such an attack can be in the form of embedded JavaScript in the phishing website that is executed upon visiting the site by a victim. It can also be through a virus that automatically redirects an unaware victim to a phishing website upon entering the URL in the address bar and obfuscate the true identity of the phishing website (Milletary, 2013).

JavaScript can also be used to execute a picture-in-picture attack (Jackson, Simon, Tan, & Barth, 2007). The phisher implements a fake web browser window as the content of a phishing website. The fake web browser window has a realistic-looking chrome area with a legitimate URL in the address bar and lock icon with fake certificate details that can be viewed by clicking on it. This fake web browser window looks like a pop-up where the user can drag it around, but within the content area of the parent page, and it can be closed and navigated. An example of this attack is shown in Fig. 10. Here, the phisher uses such a technique to target the familiarity of a user's online experience. Some websites may open a pop-up to a payment gateway and as the user is accustomed to such behaviour of the website, the user will not feel suspicious entering his or her personal information through the pop-up window.

3.3.7. Malvertising

Malvertising uses online advertisement hosting service as a means to distribute malware to victims (Nagunwa, 2014). The phishers put up advertisements with embedded malware. When a victim clicks on the advertisement, a dynamic malware will infect the victim's machine and exploit the vulnerability of the machine with the objective to steal personal information from the victim. The availability of online advertisement hosting services and easy application process for such services where minimal information is required for subscription makes this technique attractive to the phishers (Symantec, 2016). Furthermore, the phishers are able to distribute their malware through legitimate websites using advertisement without the need to compromise the websites (Symantec, 2013). This way, the user will not feel suspicious in visiting the advertising website, seeing that the advertisement is hosted on a legitimate website. The user may not be aware that the advertisements are actually supplied by the ad networks (Xing et al., 2015) and that there is a lack of verification of the contents of the advertisements (Sood & Enbody, 2011).

3.3.8. Man-in-the-middle (MITM)

In a Man-in-the-Middle (MITM) attack, the phisher places himself or herself in the middle of the communication between the victim and a web-based application, (Banday & Qadri, 2007; DigiCert, 2009; Emigh, 2005; Milletary, 2013; Ollmann, 2004; Suganya, 2016) as shown in Fig. 11. Being the middle person in between a two-way communication, the phisher is able to eavesdrop and collect personal information that the victim is submitting to a web-based application. Such an attack is difficult to detect as the information from the web-based application is still returned to the victim after authentication through the phisher and there is no external indication to signal that something is amiss. This technique can be used to circumvent the two-factor authentication by hijacking the credentials during authentication (Emigh, 2005; Milletary, 2013). Also, HTTPs or SSL web traffic will not be able to protect the victim from such attacks as the phisher can either establish two separate SSL connections (one between itself and the victim and the other between itself and the real server) (Ollmann, 2004) or have only an SSL connection to the real server but not between the victim and itself as the victim may not check for the presence of SSL (Emigh, 2005).

There are several methods to deploy a MITM attack, namely through proxies, DNS cache poisoning, and pop-up attacks. Through proxies, there are several techniques used by phishers which are transparent proxy, URL obfuscation, browser proxy configuration, relay attack through proxy program and PHP cURL module.

A transparent proxy, unlike that of a traditional one, does not require configuration on the client side. This type of proxy inter-

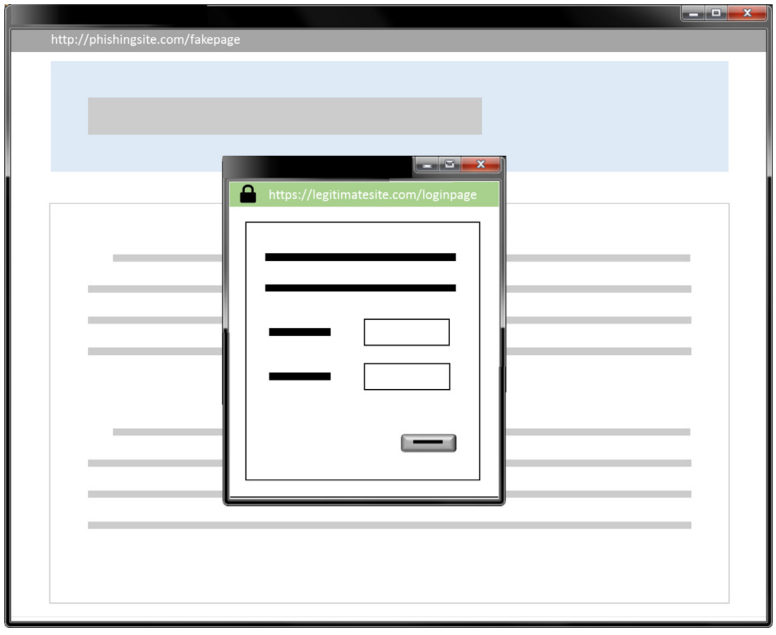

Fig. 10. The picture-in-picture attack.

Fig. 11. The Man-in-the-Middle attack.

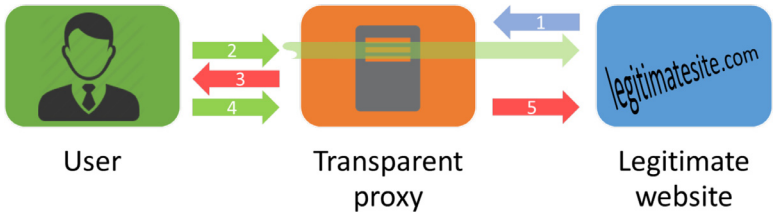

Fig. 12. The MITM attack using a transparent proxy. (1) The phisher uses the contents of the legitimate website to create a spoofed version on the cache proxy. (2) The proxy server intercepts the request to the legitimate website from the user. (3) The proxy returns the spoofed version of the website. (4) The user submits personal information to the proxy. (5) The proxy forwards the personal information to the legitimate website on the user's behalf and captures the information at the same time.

cepts all data before passing it along to the destination. From the client's point of view, the data are directed to the intended destination and the client will be unaware of the presence of the proxy. In a legitimate scenario, it is normally used to perform actions such as caching, redirection and authentication (Gibb, 2015). However, phishers are able to utilise this feature to their advantage to conduct an MITM attack (Ollmann, 2004). The phisher sets up a proxy cache of the real website by using the source code from the real website to design a spoofed version (Rader & Rahman, 2013). When the victim accesses the website, the proxy server intercepts the request and returns the spoofed version of the website. This spoofed website becomes the mediator between the victim and the real website. The spoofed website forwards all data communication between these two parties and at the same time records the personal information that it has intercepted. Fig. 12 shows the working of MITM attack using transparent proxy.

In order to trick potential victims into clicking on the spoofed website, phishers use URL obfuscation (Ollmann, 2004) to either mimic the phishing URL as close as possible to the legitimate URL or to hide the phishing URL if it looks suspicious to the potential victims. There are several obfuscation methods deployed by the phishers which are bad domain names, third-party shortened URLs, and hostname obfuscation (Ollmann, 2004; Rader & Rahman, 2013). Bad domain names involve using numbers or characters substitution to create a new domain name that resembles the legitimate domain name. Examples are as given in Table 2. URL obfuscation can be done using character sets from other languages. For example, the Cyrillic 'o' looks identical to the ASCII 'o' (Ollmann, 2004) and can be used to register a new domain name that looks highly alike to the legitimate one.

Another method that the phisher uses besides the transparent proxy is through browser proxy configuration (Ollmann, 2004; Rader & Rahman, 2013). This involves changing the proxy setting

Table 2
Examples of URL obfuscation.

Type of URL obfuscation	Example
Legitimate domain name	legit.legitimatesite.com
Obfuscation by interchanging the domain and subdomain name	legitimatesite.legit.com
Obfuscation using country code top-level domain	legit.legitimatesite.com.my
Obfuscation using character substitution	http://legit.legitimatesite.com.my
Obfuscation by using part of the legitimate URL as subdomain	legit.legitimatesite.anothersite.com

Fig. 13. The MITM attack using a proxy program that acts as a relay to the legitimate website (Hayashi, 2014). (1) The user is tricked into accessing a phishing URL and thinks it is the legitimate website that he or she intended to visit. The contents of the legitimate website are relayed to the user as he or she browses. (2) When the user accesses the page that requires submitting personal information, a spoofed version is presented to the user. The user's information is submitted to the phisher upon submitting through the spoofed page.

at the client side to force all web traffic to pass through the phishing proxy. To change the setting, the phisher will need to deploy a malware as the first stage of offence. This can be done through drive-by-download when visiting a malicious website (Cova et al., 2010; Patil & Patil, 2015) or an HTML email (Patil & Patil, 2015). Once the browser proxy configuration has been changed to use the proxy address supplied by the phisher, the phisher has placed himself or herself in between the victim and the web server that the victim is communicating with.

Phishers have employed another technique that does not require copying the content of a legitimate website such as in the case of transparent proxy, as discussed before. Instead, the phishers use a proxy program to relay the content of the legitimate website. When the victim accesses the phishing URL and browses through the website, the proxy program relays the contents of the legitimate website to the victim. Upon visiting a webpage that requires submission of the victim's personal information, a spoofed version of that page is supplied to the victim to steal the information. The attack flow using such a technique is shown in Fig. 13. Operation Huyao (Hayashi, 2014) is an example of such technique that was put to use in a phishing attack. Here, the phisher used the SEO technique to place the malicious website imitating a well-known shopping website as one of the top-ranked links in the search engine result. The victim clicked on this link in the search result without suspecting that it was not the actual shopping website. The contents of the actual shopping website were relayed to the victim. When the victim checked out his or her shopping cart, the victim was directed to a phishing webpage requesting the victim's credit card information.

Phishers can also launch a MITM attack using Hypertext Preprocessor client URL (PHP cURL) (RSA, 2016). cURL is a command line tool that uses URL syntax to perform data transfer through various protocols such as HTTP, HTTPS, FTP, and more. By performing reverse engineering on how data are requested and submitted to a legitimate website, a phisher is able to write a script to perform certain actions such as funds transfer from the victim's account to the phisher's account, making the action seem to be performed by an authorised user. Using this technique, the phisher is able to view the victim's account details and steal personal information such as address, phone number, etc.

Domain Name Server (DNS) poisoning is an advanced phishing attack where the DNS records are poisoned by altering the entry inside to point a domain name to a false IP address. Such attack is also known as pharming. DNS poisoning can be done at the client's side through the host file (Banday & Qadri, 2007; Emigh, 2005; Milletary, 2013; Suganya, 2016), network side through the DNS cache (Milletary, 2013; Ollmann, 2004) or directly on the DNS server (Rader & Rahman, 2013). Malware is used to alter the DNS records (Milletary, 2013). The DNS cache is a copy of the DNS records obtained from the DNS server. These DNS records are used to resolve the domain name of a URL to its corresponding IP address, which is the web address of that website. By altering the DNS records, the domain name can be made to point to a phishing website instead of the actual one. Manual entry of the URL will not help to circumvent this attack. The operation of such attack is shown in Fig. 14.

Another form of MITM attack is the pop-up attack (Mohammad et al., 2015; Trusteer, 2008). This attack takes advantage of the user experience where online users commonly encounter pop-up windows to the payment gateway or account login screen as shown in Fig. 15. The phisher takes advantage of this by presenting a pop-up window with a fake login screen. For this attack to be successful, the phisher will need to identify which website the user is currently browsing. This is done by luring the user to visit a compromised website. If the user is browsing a certain website that is of interest to the phisher to phish for personal information, the compromised website will inject a pop-up requesting for the user to log in. The user may think that the previous login has timed out and proceeds to log in again. Any information submitted through the pop-up window will be directed to the phisher.

3.3.9. *Mobile phone*

Performing a phishing attack through mobile phone is gaining popularity among the phishers due to the fast growth of the mobile phone markets (Felt & Wagner, 2011; Niu, Hsu, & Chen, 2008; Vural & Venter, 2011). As the number of mobile phones increase, so do the potential number of victims which the phishers can phish for their personal information. Furthermore, the small screen of the mobile phone limits the amount of information that can be included in the user interface, and the lack of indicators of the identity of an application (Felt & Wagner, 2011) makes distinguishing malicious applications from the legitimate ones difficult.

Felt and Wagner (2011) identified four ways of control transfer that occurs in a mobile phone which can be exploited by phishers in a phishing attack and this is given in Fig. 16. Mobile sender ⇒ Mobile target refers to control transfer from a mobile application to another mobile application. Examples are an application that allows a user to post content to social media through social media

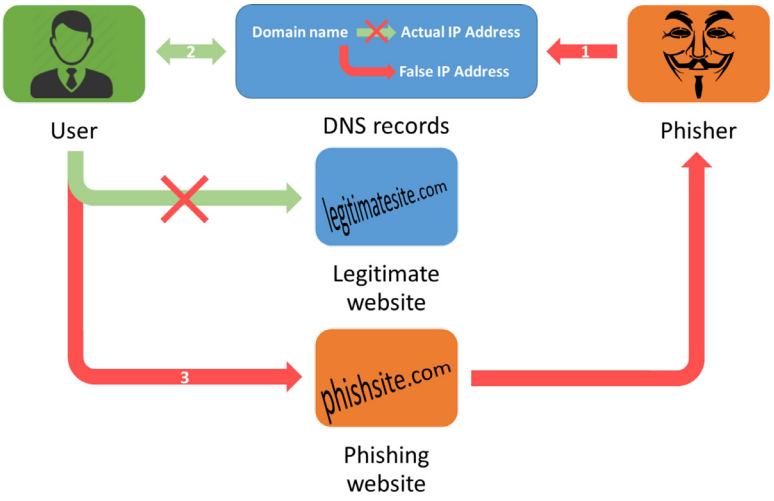

Fig. 14. The DNS poisoning attack. (1) The phisher poisons the DNS records to point a domain name to a false IP address. (2) When the user enters the URL of the affected domain name, the DNS records return the false IP address. (3) The false IP address directs the user to a phishing website where any personal information entered on that website is submitted to the phisher.

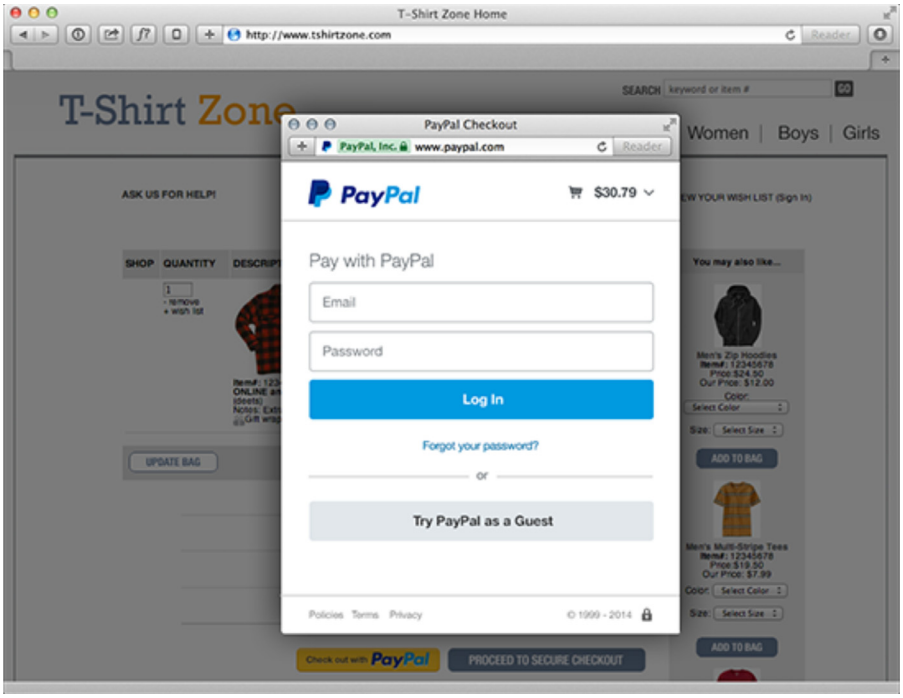

Fig. 15. An example of pop-up login screen from PayPal (PayPal, 2017).

applications such as the Facebook share button, as well as applications that link the user to the application stores for other related applications. This involves the user installing a phishing application on the mobile phone with features that allow the user to perform certain actions such as sharing on social media. When the user performs such actions from this phishing application, a screen requesting the user to log in to the intended application is shown and unknowingly, the user submits the login credentials to the phisher through the phishing application.

Mobile sender ⇒ Web target refers to the control transfer from a mobile application to a web browser. An example is an application that directs the user to a website that requires the user to log in. Similar to the Mobile sender ⇒ Mobile target control transfer, such exploitation of this control transfer requires a phishing application to be installed on the user's mobile phone. Instead of showing a screen requesting the user to log in, the user is directed through the web browser to a phishing website for the user to submit his or her login credentials. The phisher can hide the URL bar of the browser to avoid raising any suspicion from the user.

Web sender ⇒ Mobile target refers to control transfer from a web browser to a mobile application. An example is when a user is following a link on a website which is linked to the application that is installed on the mobile phone. The phisher can trick the user by emulating the screen of a legitimate application on the web browser. When the user sees the familiar screen, the user will think that the link from the website that he or she clicks has directed him or her to the intended application. This spoofed screen will request the user to log in.

Web sender ⇒ Web target refers to the usual operation of accessing a link on a website that directs to another website. The phisher may obfuscate the phishing URL on the web browser by hiding the actual URL bar, displaying a fake URL bar and prevent-

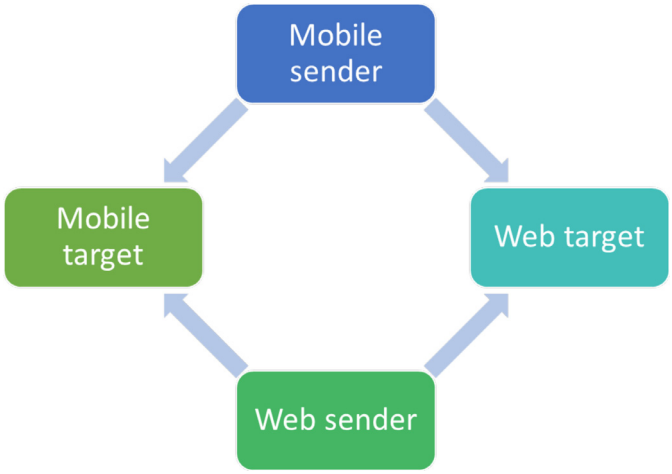

Fig. 16. The four ways of control transfer in a mobile phone.

ing the user from scrolling up to the actual URL bar by jumping back to the fake URL bar when the user attempts to scroll up (Niu et al., 2008; Rydstedt, Gourdin, Bursztein, & Boneh, 2010).

Phishers can also use the notification service on a mobile phone to launch a phishing attack (Xu & Zhu, 2012). Certain mobile operating systems (OS) such as Android and BlackBerry OS allow notification customisation. This allows the phishers, through the installed third-party Trojan application, to display a phishing notification that looks like a Facebook notification and directs the user to a phishing login screen. Besides exploiting the notification service on the mobile phone, malware infected applications can collect information of the user's activities such as the user's personal information, information that the user submitted to a website or the user's browsing activities and relay this information back to the phisher (TrendLabs, 2012). Through the malware infected applications, the phishers are able to obtain the mobile number of the user for a smishing attack or send an SMS to the user directly from the application (Tufts, 2012).

It is observed that for the phisher to launch his or her attack through a mobile phone, it may involve installing a third-party application on the victim's phone. This can be done through smishing (PandaLabs, 2015) or vishing to trick the victim into installing the phishing application. There is another way to install such applications onto the victim's mobile phone without his or her consent or knowledge, which is through cross-over threats (Symantec, 2016). Using the feature where the user is able to install an application directly onto his or her phone from a computer with a web browser and internet connectivity, the phisher can steal the browser cookies to obtain the user's credentials and remotely install this phishing application onto the user's mobile phone.

3.3.10. Phishing kits

Phishing kits are tools that enable phishers to generate phishing website, emails and scripts to obtain user input without any need of advanced programming skills (Milletary, 2013). These kits can be obtained from the cybercriminal marketplace for a price (Symantec, 2016) or through free distribution by the kit developer in underground circles (Cova et al., 2010). However, most of these free phishing kits have backdoors that leak out the personal information harvested by the phishing kit user back to the developer (Chaudhry et al., 2016; Cova et al., 2010). Phishing kits do not play a direct role in phishing for personal information from the victims, but do aid in deploying phishing attacks. This makes launching phishing attacks easily accessible to anyone with or without the in-depth knowledge of programming. An example of such kit is the "Universal Man-in-the-Middle Phishing Kit" (Elledge, 2007; Singh, 2007). This kit allows the phisher to deploy a MITM attack and capture personal information as the victim communicates with a legitimate website.

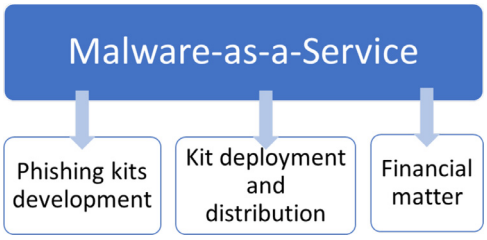

Fig. 17. The components of the Malware-as-a-Service (Moreno, 2016).

Instead of just selling phishing kits in the cybercriminal marketplace, cybercriminals are shifting into a service-based business model on top of the phishing kit itself as a product. This business model is called Malware-as-a-Service (MaaS) (VeriSign, 2012). There are three components or levels under the umbrella of the MaaS: developing the kits, deploying the kits and managing the financial aspect of a phishing attack (Moreno, 2016) as shown in Fig. 17. The first level is the development of the phishing kit for sale by the developer. The second level involves the service provider: (i) to deploy, (ii) to distribute the kit, or (iii) to host the phishing website. The third level is the financially related service provider such as money mules and financial data providers. This business model forms the complete package to conduct a phishing attack, from deploying to the financial matter. This package can be purchased as a whole or by separate components only.

3.3.11. Search engine optimisation

Phishing website can be delivered to potential victims through search engine results (Banday & Qadri, 2007; Chaudhry et al., 2016; Emigh, 2005; Suganya, 2016). The phisher creates a phishing website and optimises it for the search engine's indexing. Potential victims who use search engines to search for the website of a particular service or product provider may click on the phishing link in the search results, thinking that the link will direct to the intended website. The phisher may make the "lure" of the phishing attack more attractive and tempting by offering goods or services at a ridiculous price. Blackhat SEO (Nagunwa, 2014) may be used by the phishers to increase the page rank of the phishing link in the search engine results. This is done by injecting keywords of popular trends or events in their website which ensures their website is ranked as one of the top search results. This will further increase the likelihood of the potential victim clicking on the link.

3.3.12. Session fixation

For stateless protocols such as HTTP and HTTPS, the session of communication between the client and the server is not retained and the connection is lost once the transmission of data is completed. This forms an issue for the user who is accessing his or her account and such protocol will not keep the active session that the user is currently in. For example, when the user logs in on a shopping website and browses for goods, the server needs to have a way to maintain the user's session and keep track of his or her activity such as adding items into the cart and checking out later.

The common way to do this is using Session Identifiers (SIDs). After the user has authenticated and logged into a website, a unique key called SID is assigned to the user's session within the website. This key will identify the session that the user is in while he or she is browsing through several webpages and performing certain actions within the website. This SID can be stored in the form of a cookie, form field or in a URL.

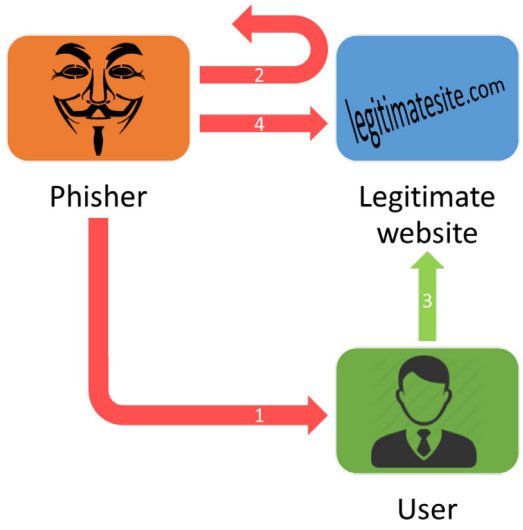

Fig. 18. The Session Fixation attack. (Ollmann, 2004). (1) The phisher sends a URL with a predefined SID for the user to log in. (2) The phisher attempts to access the session through the website and received an error as the session does not exist. (3) If the user is logged into the website using the URL then the session becomes active. (4) When the session is active, the phisher can hijack the session to perform malicious activity.

Phishers are able to target the vulnerability of SID in poor state management systems and perform session hijacking attacks. Such technique is called Session Fixation or Present Session Attack (Ollmann, 2004). This technique requires the user to authenticate a session using a SID specified by the phisher and the phisher is then able to hijack that session to perform certain actions on behalf of the user.

The operation of this attack is shown in Fig. 18. To initiate the attack, the phisher sends an email containing a predefined SID within the URL, requesting the user to log in. Here, the vector used by this technical approach is via email. Then, the phisher attempts to connect to the session in the website using the predefined SID. As long as the user did not access the given URL and authenticated into the predefined session, the phisher will receive an error from the web server as such session is not available. Once the user authenticates the predefined session, the phisher is then able to hijack the session and perform malicious activities such as fund transfer. The user will not be suspicious of the URL given by the phisher as the URL points to a legitimate website.

3.3.13. Social engineering

Social engineering involves the phishers tapping into the victim's trust, emotions such as sympathy or fear, willingness to help and gullibility to achieve their objective (Mitnick & Simon, 2002). The base of social engineering is to divert the victim from making rational choices, which leads the victim to make emotional choices instead (Goel, Williams, & Dincelli, 2017). Examples of such emotions are fear, greed, curiosity, anger, friendship, patriotism, vanity, altruism, community belonging, sense of duty and authority. By manipulating and taking advantage of the victim's emotion, such as instilling fear of losing something valuable, the victim will perform irrational actions such as revealing his or her personal information to the phisher (Kim & Kim, 2013). Such action arose from one's protective nature to take immediate, precautionary action (Leventhal, 1970). For example, a phisher might contact his or her victim about a billing issue that might lead to discontinuation of a service or account suspension, and immediate action is required to settle this issue. The action to be taken by the victim will involve the victim revealing his or her personal information to the phisher.

Greed is another emotional factor that is played upon by the phishers (Hong, 2012). A prime example of the interplay of such emotion is the "Nigerian 419 scams" where the phisher is disguised as a wealthy individual who requires assistance in transferring a large sum of money offshore and will compensate the victim for his or her trouble. However, the victim will need to perform some up-front payment first. Blinded by the greed for a big return after a small up-front payment, the victim proceeds with the upfront payment to the phisher. To increase the effectiveness of the greed emotion factor, it is coupled with a sense of urgency (Cialdini, 1993) to further cloud the victim's judgement.

Phishing attack by social engineering is propagated through the vectors of email, website, instant messaging, smishing and social networks (Almomani et al., 2013; Heartfield & Loukas, 2015). It can also be deployed using eFax and vishing. These vectors can be used to trick victims into clicking a link to a phishing website. The phishing website is carefully designed to avoid raising any suspicion from the victim and to collect the victim's personal information. Such attack is also known as semantic attack (Heartfield & Loukas, 2015) where it involves the "manipulation of user-computer interfacing" with the aim to deceive the victim in breaching his or her own personal information.

3.3.14. Sound-squatting

Sound-squatting (Nikiforakis, Balduzzi, Desmet, Piessens, & Joosen, 2014) is a domain squatting technique that the phishers use by registering domain names that sound similar to a legitimate website. Such similar sounding words are called homonyms. Examples are *air* and *heir*, *ascent* and *assent*, and *base* and *bass*. The phishers may also use the word or digit of a number as homonyms. Examples are www.highfive.com and www.high5.com. The phisher takes advantage of the user's confusion of the homonyms and entering the wrong word, but having the same sound when keying in the URL. The user is then directed to a phishing version of the legitimate website. The phisher may register several domain names with homonyms of a legitimate domain name.

3.3.15. Spear phishing

Spear phishing (Banday & Qadri, 2007; Chaudhry et al., 2016; Elledge, 2007; Goel et al., 2017; Krombholz, Hobel, Huber, & Weippl, 2015; Nagunwa, 2014; Ollmann, 2004; Singh, 2007; Symantec, 2016) is a targeted attack against an individual, a group or an organisation. Spear phishing has become the popular choice (Nagunwa, 2014) by phishers over the conventional phishing using mass and random email phishing. This is because of the high success rate compared to the conventional ones (Krombholz et al., 2015). Spear phishing uses specially crafted email mimicking a sender whom the victim knows. The content of the email is relevant to the victim which will not trigger any suspicion from the victim.

The effectiveness of spear phishing is high because internet users will normally trust email or eFax from the website of a presumed organisation that they used before or have an account with (Downs, Holbrook, & Cranor, 2006). They are more likely to login into a website bearing the identity of an organisation that they logged in before, without checking carefully whether it is a phishing website or not. Also, a higher success rate of phishing is achieved when phishers contact their victim using the identity of their victim's friend (Jagatic, Johnson, Jakobsson, & Menczer, 2007) or colleague (Halevi, Memon, & Nov, 2015). For example, the phisher sends an email to the staff of an organisation using the identity of the IT support personnel in that organisation, requesting the staff to either email back the login details or run the attachment which contains malware. These phishing attempts may

lead to a large malicious attack such as compromising the network or data of an organisation (Elledge, 2007; Ollmann, 2004).

Spear phishing technique is the preferred choice by phishers to deploy an Advanced Persistent Threat (APT) attack (Trend Micro, 2012). This is because APT is a targeted attack and using spear phishing, the phisher can deploy the attack on a specific individual, group or organisation. APT is a long-term attack that infiltrates a specific target of interest, performs a low-profile and slow attack to prevent detection and uses zero-day vulnerability exploits or malware to achieve a set of objectives such as espionage or sabotage (Symantec, 2011).

For the phisher to craft an email that is relevant to the target against whom the phisher wants to launch an attack, he or she will need to gain some information that is related to the target. This can be done using browser sniffing (Jakobsson & Stamm, 2006) to "sniff" which websites had been visited by the target. This can be done through access time for a particular URL, DNS caching and cache cookies (Felten & Schneider, 2000). If the time taken to access a particular URL or to perform a DNS lookup is short, that website has been accessed before, causing the client to have a cache of it for fast access or the DNS cache already has that particular DNS entry. Also, by looking for the presence of cache cookies of a certain website, the phisher will know whether the target has visited that website before. Thus, by knowing the website that the target visited before, it is possible to present to the target a phishing email with content or using an affiliation that the target is familiar with. The phisher is able to deploy such sniffing technique through email with a link to a website with JavaScript embedded in the site, web advertisement, search engine optimisation or HTML email (Felten & Schneider, 2000). The script will measure the access time and feedback to the phisher.

3.3.16. SQL injection

Structured Query Language (SQL) injection technique is the exploitation of vulnerabilities in a database that does not perform filtering properly and allows database commands to be injected and executed, leading to the leakage of information (Emigh, 2005). This is done through injection of SQL commands into an SQL statement or query through the webpage input to alter the original intent of statement (Kals et al., 2006; Patil & Patil, 2015; Rietta, 2006). The injected code is concatenated with the SQL commands through the user-input variables and this dynamic SQL command is executed. The exploitation of this vulnerability can be done by exploiting the current SQL query or adding new query through multiple queries (Nagunwa, 2014).

SQL query:
"SELECT * FROM Users WHERE User = '" + **userName** + "' AND Password = '" + **password** + "'"
SQL query with injection:
"SELECT * FROM Users WHERE User = '**'OR 1=1 --** AND Password = '

An example (Kals et al., 2006; Nagunwa, 2014) above shows the SQL injection at work. The SQL query is given where it retrieves the user's information from the table *Users* based on the login details. The insertion of '*OR 1=1* –' in the user field and leaving the password field empty forms the concatenated query. From the query, the user authentication part is added with an *OR* and an always true statement (1=1). This causes the authentication to be true for all the users' records in the table. '–' is a comment command that makes the remaining string after the command as a comment and will be ignored by the SQL database engine. When this query is executed, the database will retrieve all the information in the *Users* table and display it to the phisher.

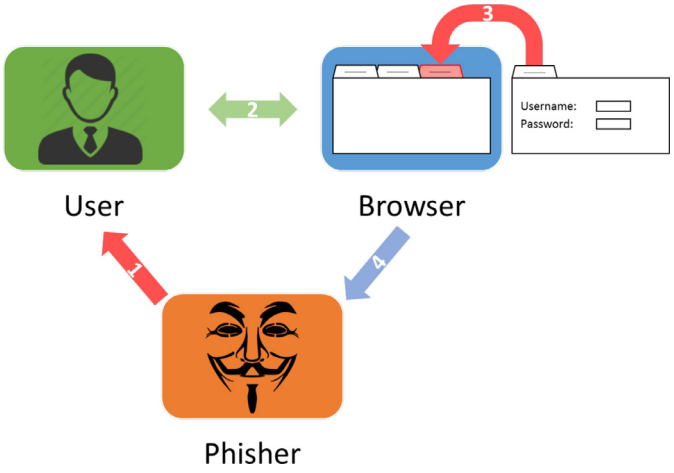

Fig. 19. The Tabnapping attack. (1) The phisher sends a URL to a website with embedded JavaScript. (2) The user clicks on the link and the phishing website is loaded in one of the browser's tabs. (3) When the user switches to another tab and leaves the tab with the phishing website inactive, a script will be executed to load the login page of the website that the user is visiting in the other tab. (4) The user logs in again through the phishing website, thinking that the logged in session has expired.

3.3.17. Tabnapping

Tabnapping (Suri, Tomar, & Sahu, 2012) was first introduced by Aza Raskin, a creative lead of Firefox in 2010. The name tabnapping is a blend word formed by blending *Tab* and *Kidnapping* to describe the attack as a form of "kidnapping" of a tab in a browser. The operation of such an attack is shown in Fig. 19. First, the phisher emails the link of a phishing website to a user. Once the user clicks on the link, a tab in the user's browser will be opened to load the phishing website which looks like an ordinary website. The embedded JavaScript in the phishing website monitors the browsing activity of the user. Once the user navigates to other tabs in the browser and leaves the tab with the phishing website out of focus, the tab loads a phishing login screen and changes the favicon and the title of the tab to spoof a legitimate website such as Gmail. When the user browses through the opened tabs and notices the phishing login screen, the user might think that the logged in session on that website has expired and the user needs to log in again. Thus, the user submits the login credentials through the phishing login screen without realising that it is a phishing website. A demonstration of such attack is given by Aza Raskin on his website (Raskin, 2010). The phisher uses this technique as the user is less suspicious of the previously open tab and is not aware that the previously loaded content can be changed using JavaScript instead of remaining static as the user presumed. Such attack is only successful for the user who opens multiple tabs in the browser and easily loses track of the contents of all the opened tabs that are inactive.

3.3.18. Typo-squatting

Typo-squatting (Banerjee, Barman, Faloutsos, & Bhuyan, 2008; Wang, Beck, Wang, Verbowski, & Daniels, 2006; Waziri Jr, 2015) is a type of domain squatting technique that the phisher uses by registering domain names that are typos of a legitimate domain name. There are five types of possible domain name typo that can be accidentally performed by the user and these are given in Table 3. Such attack affects a user who manually types in the URL in the address bar. When the user mistypes the URL by accidentally pressing the adjacent key or missing a character, the mistyped URL may direct the user to a phishing website that may look like the legitimate website which the user intended to visit or a website that loads a malware onto the user's machine (Wang et al., 2006).

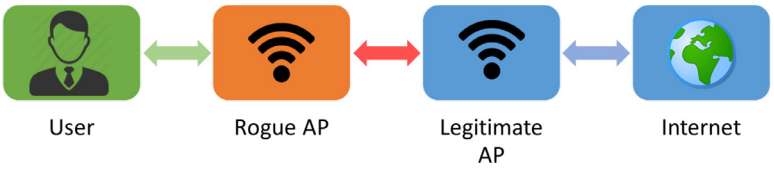

Fig. 20. The WiPhishing/Evil Twins attack.

Table 3
Examples of domain name typos (Wang et al., 2006) in the URL: www.mybank2us.com.

Typo type	URL
Missing dot typos	wwwmybank2us.com
Character omission typos	www.mybankus.com
Character permutation typos	www.mybank2su.com
Character replacement typos	www.mybanl2us.com
Character insertion typos	www.mybank2uss.com

Here, the phisher takes advantage of the typing mistake committed by the user.

3.3.19. Whaling

Whaling is similar to spear phishing in the sense that it is a targeted attack, but the target consists of high-level executives (Ollmann, 2004) who have high privilege access to information or resources within an organisation. Whaling is done through malware that gives the phishers back-door access into the organisation's system or deploys keyloggers. As this is a targeted attack, the phisher will invest more time to craft his or her attack vectors, being by email or eFax, to enhance the success chance of the victim clicking on a link or downloading an attachment containing malware. Whaling is used as a pre-emptive attack to a further malicious attack called business email compromise (BEM) (FBI, 2017). The compromised email from a high-level executive such as CEO is used to instruct the CEO's subordinate to perform unauthorised wire-transfer payments.

3.3.20. Wiphishing/evil twin

WiPhishing or Evil Twin (Sinha, Haddad, Nightingale, Rushing, & Thomas, 2006; Song, Yang, & Gu, 2010) is a technique of phishing that uses a wireless network. The phisher places himself or herself in between the internet user and a legitimate wireless access point (AP) by using a rogue AP as shown in Fig. 20. First, the phisher deploys a rogue AP bearing the same Service Set Identifier (SSID) or network name and the radio frequency of an existing legitimate AP in an area. Software that enables a laptop to be the access point in a wireless network is available (TrueSoftware, 2017). Then, the phisher will place this rogue AP closer to the user which may cause the user to connect to the AP with the strongest signal or having the computer to automatically connect to the rogue AP as it chooses the AP with the strongest signal for a group of APs with the identical name. Finally, the phisher is able to eavesdrop on the information that the user submitted and received through the rogue AP that the user's machine is connected to. The phisher normally will target public places with free hotspot such as cafes, airports, hotels, etc. (Song et al., 2010).

4. Intertwine between the technical approaches

The technical approaches discussed in the previous section may function independently, but a phishing attack can use a combination of them as shown in Fig 21. Here, a discussion of such combination is given.

A: Clickjacking and XSS attack

XSS vulnerability can be used to implement a clickjacking attack on a legitimate website. Such is the case for LinkedIn website (Reeve, 2015). This kind of vulnerability was discovered by Ruben van Vreeland who is the CEO of BitSensor. He discovered that he can bypass the security system in place for the LinkedIn website by using the website's cascading style sheets (CSS). Then, by using XSS attack, a clickable element that overlays the entire webpage can be placed. Any click performed by the user will direct the user to whichever website the phisher desires. It can lead to a phishing website to steal the user's personal information or to download malware onto the user's machine.

B: Clickjacking and mobile phone

Mobile phone is vulnerable to clickjacking attacks due to a lack of defence against it for mobile websites and some web developers felt that clickjacking may not be an issue for mobile websites (Rydstedt et al., 2010). Such attack is called tap-jacking (Rydstedt et al., 2010) as the user taps on the screen of the mobile phone instead of clicking. Both iPhone and Android browsers support JavaScript and frames, fulfilling the requirements to launch a tap-jacking attack.

C: Cloud computing and XSS attack

Cloud computing is subjected to XSS vulnerability (Chow et al., 2009; Grobauer, Walloschek, & Stöcker, 2011). The XSS vulnerability may exist in any cloud service with applications displaying dynamic webpages and having no proper data validation in place such as in the case of Salesforce cloud (Winder, 2015). The vulnerability in the Salesforce cloud allows phishers to utilise the Salesforce application as a platform to phish for the user's credential.

D: Cloud computing and spear phishing

The attack on cloud computing services can start with spear phishing as the first stage of attack (Nagunwa, 2014). The login information to gain access to the cloud service is phished by tricking the person who has the access to divulge his or her login credentials through spear phishing. Using these gained credentials, the phisher has access to the information stored in the cloud.

E: Cloud computing and SQL injection

Cloud computing is also subjected to SQL injection (Chow et al., 2009; Grobauer et al., 2011). With corporations adopting the Database-as-a-Service or Storage-as-a-Service (Agrawal, Das, & Abbadi, 2011) in cloud computing, the vulnerability that exists in a database also exists in the cloud database.

F: Drive-by-download and man-in-the-middle

Malware installed on a machine such as Trojan through drive-by-download can function as a web proxy (Nagunwa, 2014). Using this malware, the phisher is able to perform a MITM attack on the user. The web proxy can redirect the web traffic to the DNS of the phisher which then leads to the phishing website. It can also be used to intercept a transaction and falsify information that is being transmitted to a legitimate server. The malware can also be used to transform a machine into a botnet (Emigh, 2005; Schiller & Binkley, 2007). The phisher is able to use the botnet in a network to poison the address resolution protocol (ARP) cache to set itself as the default gateway. All data transmission will go through the botnet before the data is forwarded to the actual gateway. Using this method, the phisher is able to harvest personal information sent by the user through the phisher's botnet.

G: Social engineering and drive-by-download;

H: Social engineering and man-in-the-middle;

I: Social engineering and mobile phone;

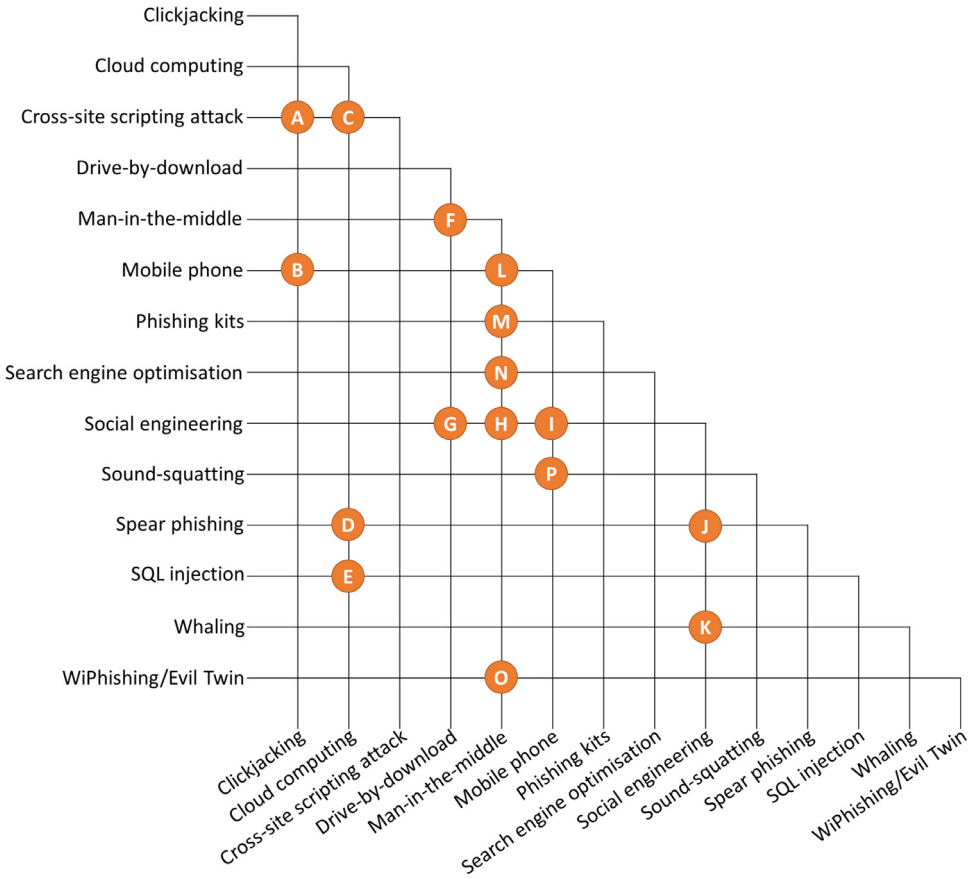

Fig. 21. The combination of technical approaches in a phishing attack, represented by the nodes. The description of these combinations is given in the text, corresponding to the alphabet labels.

J: Social engineering and spear phishing;

K: Social engineering and whaling

Social engineering is normally the forefront of a phishing attack. Such a method does not require much technical knowledge; rather, it uses just an act of deception to trick the user to fall into the trap of phishing. Thus, it is commonly being used with other phishing technical approaches such as drive-by-download (Milletary, 2013; Schiller & Binkley, 2007), spear phishing (Elledge, 2007; Ollmann, 2004), whaling (Ollmann, 2004), MITM (DigiCert, 2009) and mobile phone (Felt & Wagner, 2011). Social engineering is normally the first stage in a phishing attack.

L: Man-in-the-middle and mobile phone

MITM attack can be deployed in a mobile phone through its applications (Conroy, 2013). There are two ways to launch a MITM attack in mobile phones, which are scheme squatting and task interception (Felt & Wagner, 2011). In scheme squatting or activity hijacking (Chin, Felt, Greenwood, & Wagner, 2011), the phisher implements a phishing application that attempts to hijack the scheme or activity of a legitimate application. For example, when the user proceeds to make payment and launches a payment activity, the phishing application responds to that activity and presents the user with a spoofed payment gateway. Personal information of the user can be stolen using this method. In task interception, the phishing application will poll and wait until the user runs a specific legitimate application. Upon running the legitimate application, the phishing application launches itself and presents a phishing screen, mimicking the legitimate application that the user just ran. This method will trick the user into thinking that the phishing screen is from the legitimate application.

M: Man-in-the-middle and phishing kits

MITM attack can be launched using phishing kits (Rader & Rahman, 2013). Phishing kits such as the Universal Man-in-the-Middle phishing kit (Elledge, 2007; Singh, 2007) automate the creation of the phishing website to launch the MITM attack.

N: Man-in-the-middle and search engine optimisation

MITM attack can be launched using SEO. Here, SEO is the first stage of the phishing attack. SEO is used to insert the phishing website into the search engine results as done in Operation Huyao (Hayashi, 2014). This phishing website is used to perform MITM attack on the victims.

O: Man-in-the-middle and WiPhishing/Evil twin

WiPhishing or Evil Twin is used to launch a MITM attack. The phisher is able to intercept any information submitted online by the users who connect to the rogue AP set up by the phisher. Information such as passwords and credit card information can be obtained by the phisher by snooping at the exchange of information in the communication links (Yang, Song, & Gu, 2012).

P: Mobile phone and sound-squatting

Sound-squatting affects user who uses text-to-speech software on a mobile phone, such as Apple's Siri to visit websites. The text-to-speech software is commonly used by the user who cannot view the phone screen at that moment of time such as while driving. The phisher may mask the phishing website with the same sound as the legitimate website. When the text-to-speech software reads out the website link on an email, the user will not be able to know that it is not the legitimate website by hearing the readout. For example, youtube.com and yewtube.com (Nikiforakis et al., 2014).

The user will proceed to open the link to the phishing website by voice command.

5. Conclusion and future work

The phishing techniques deployed in existing phishing attacks can be studied in detail through their characteristics of the medium and vector which they reside in and their technical approaches. A systematic understanding of these approaches will lead to the development of a more effective and holistic manner of anti-phishing technique to tackle the phishing problem. Furthermore, this knowledge will help the general public to take precautionary and preventive actions against these phishing attacks and the policy makers to implement policies to curb any further exploitation by the phishers.

The internet is the most popular medium targeted by phishers with the availability of six out of the total of eight vectors associated with it. However, this does not mean that the other two mediums, which are the short messaging service and voice, can be ignored as phishing attempts through smishing and vishing are still popular with the rapid expansion of the mobile phone market. With the fast-growing market of cloud computing and mobile phone, these two areas will gain the attention of phishers. Anti-phishing effort is needed, especially in these two areas.

It is observed that the technical approaches can be generalised to two methods which are vulnerability exploitation and phishing websites. Several approaches exploit the vulnerability of a system, software, website or database for delivery of malware to steal sensitive information from the victims. Examples of technical approaches related to vulnerability exploitation are browser vulnerabilities, cloud computing, cross-site scripting attack, drive-by-download, malvertising, man-in-the-middle, session fixation and SQL injection. There are technical approaches that trick the victims into visiting a phishing website and giving out their information, for example, clickjacking, cloud computing, malvertising, man-in-the-middle, mobile phone, search engine optimization, social engineering, spear phishing, whaling, tabnapping, typo-squatting, and sound-squatting. The phishers use these technical approaches to lure their victims into visiting the phishing website, and use the elements in the phishing websites to steal sensitive information such as login credentials from the victims. It is noticed that some technical approaches fall under both of the general methods.

Precautionary steps can be taken against technical approaches that fall under vulnerability exploitation. Browser vulnerabilities, cloud computing system vulnerabilities, drive-by-download, malvertising, and man-in-the-middle can be reduced by ensuring the related system or software is patched with the latest updates. As for cross-site scripting attack and SQL injection, precautionary steps can be taken during websites or databases development to ensure the vulnerabilities are removed. Practical guidelines for developing websites and databases can be formulated based on the characteristics of phishing attacks. The World Wide Web Consortium (W3C) is an example of an international community that develops web standards. However, such standards will be useless without enforcement. There is a need for a system to enforce the said standard in web design and development. Also, tools with website vulnerability scanning features will be useful for web developers to do a check of their website and fix the vulnerabilities that exist.

While it is common to associate phishing attacks with social engineering as in the case of technical approaches that fall under phishing websites, there are several phishing approaches that are technical such as clickjacking, malvertising, and man-in-the-middle. For example, clickjacking involves the UI manipulation of a webpage which causes the user to perform a certain action unknowingly when interacting with the webpage. Despite visual inspection from the user, such attacks cannot be identified easily. Relying only on user education as a preventive measure in a phishing attack is highly insufficient. This review shows that the development of intelligent systems to counter against these technical approaches is needed as such countermeasures will be able to detect and cripple both existing discovered attacks and new phishing threats.

In addition, the dependence on anti-phishing approaches that detect phishing websites upon visiting a website is not sufficient. Such detection exposes the victim to drive-by-download attacks where malware, virus or shellcode is injected into the machine upon accessing the website. Instead, detection of phishing website through the link provided in email or social media is more desirable as no access to the website is made. The anti-phishing effort in this direction is needed to curb the phishing attacks.

It is also observed that JavaScript is commonly used in some of the technical approaches, namely, cross-site scripting attack, JavaScript obfuscation, malvertising, and tabnapping. Anti-phishing methods to identify the malicious behaviour of JavaScript in these technical approaches are very much needed, especially for malvertising. Malvertising is mentioned specifically here because internet users are exposed to online advertisements frequently and are easily susceptible to the threat of malvertising.

This review is a stepping stone to gear the research direction to the development of a holistic countermeasure. A survey of the current effort to counter against these phishing attacks is needed as well. Only then, the area where the countermeasure against a certain phishing attack is lacking can be identified. A systematic manner of reviewing the phishing techniques will be extended to the reviewing of current anti-phishing methods from literature as future work. This will give a complete picture of the understanding of phishing and its countermeasures to researchers and developers, allowing for the production of more effective anti-phishing techniques.

Acknowledgment

The funding for this project was made possible through the research grant obtained from Universiti Malaysia Sarawak under the Special FRGS 2016 Cycle [Grant No: F08/SpFRGS/1533/2017]. Dr. Kelvin Yong Sheng Chek is a Recipient of Universiti Malaysia Sarawak Post-Doctoral scheme for the Project entitled “Enhancing Pharming Attack Detection Model Through Website Identity Discovery”.

References

Agrawal, D., Das, S., & Abbadi, A. E. (2011). Big data and cloud computing: Current state and future opportunities. In *Proceedings of the fourteenth international conference on extending database technology* (pp. 530–533). ACM.

Akhawe, D., He, W., Li, Z., Moazzezi, R., & Song, D. (2014). Clickjacking revisited: A perceptual view of UI security. In *Proceedings of the eighth usenix workshop on offensive technologies (woot 14)* (pp. 1–14). USENIX Association.

Almomani, A., Gupta, B. B., Atawneh, S., Meulenberg, A., & Almomani, E. (2013). A survey of phishing email filtering techniques. *IEEE Communications Surveys and Tutorials, 15*(4), 2070–2090.

APWG (2017). *Phishing activity trends report, 4th quarter 2016. White Paper*. Anti-Phishing Working Group.

Australian Computer Emergency Response Team (AusCERT) (2003). Microsoft internet explorer incorrectly displays URLs. Accessed on: Apr. 19, 2017. https://www.auscert.org.au/render.html?it=3680.

Banday, M. T., & Qadri, J. A. (2007). Phishing – a growing threat to e-commerce. *The Business Review, 12*(2), 76–83.

Banerjee, A., Barman, D., Faloutsos, M., & Bhuyan, L. N. (2008). Cyber-fraud is one typo away. In *Proceedings of the twenty-seventh conference on computer communications–IEEE infocom 2008* (pp. 66–70). IEEE.

Barkah, A. (2009). Update to Google docs security issues. Accessed on: Apr. 21, 2017, https://peekay.org/2009/11/13/65/.

Borgaonkar, R. (2010). An analysis of the asprox botnet. In *Proceedings of the fourth international conference on emerging security information system and technology (securware)* (pp. 148–153). Venice, Italy: IEEE.

Chaudhry, J. A., Chaudhry, S. A., & Rittenhouse, R. G. (2016). Phishing attacks and defenses. *International Journal of Security and Its Applications, 10*(1), 247–256.

Chin, E., Felt, A. P., Greenwood, K., & Wagner, D. (2011). Analyzing inter-application communication in android. In *Proceedings of the ninth international conference on mobile systems, applications, and services* (pp. 239–252). ACM.

Chow, R., Golle, P., Jakobsson, M., Shi, E., Staddon, J., Masuoka, R., & Molina, J. (2009). Controlling data in the cloud: Outsourcing computation without outsourcing control. *Proceedings of the ACM workshop on cloud computing and security*, 85–90.

Cialdini, R. B. (1993). *Influence: Science and practice*. New York: Harper Collins.

Conroy, J. (2013). Financial institutions, merchants, and the race against cyberthreats. *White Paper,*. RSA.

Cova, M., Kruegel, C., & Vigna, G. (2010). Detection and analysis of drive-by-download attacks and malicious javascript code. In *Proceedings of the nineteenth international conference on world wide web* (pp. 281–290). New York, NY, USA: ACM.

DigiCert (2009). Phishing: A primer on what phishing is and how it works. *White Paper*. DigiCert, Inc..

Dormann, W. (2005). Vulnerability note vu#356600: Microsoft internet explorer dhtml editing activex control contains a cross-domain vulnerability. Accessed on: Apr. 19, 2017, https://www.kb.cert.org/vuls/id/356600.

Dormann, W., & Manion, A. (2004). Vulnerability note vu#490708: Microsoft internet explorer window.createpopup( ) method creates chromeless windows, Accessed on: Apr. 19, 2017. https://www.kb.cert.org/vuls/id/490708.

Downs, J. S., Holbrook, M. B., & Cranor, L. F. (2006). Decision strategies and susceptibility to phishing. In *Proceedings of the second symposium on usable privacy and security, Pittsburgh, Pennsylvania, USA* (pp. 79–90).

Elledge, A. (2007). Phishing: An analysis of a growing threat. *SANS Institute InforSec Reading Room*, 1–20.

Emigh, A. (2005). Online identity theft: Phishing technology, chokepoints and countermeasures. *ITTC Report on Online Identity Theft Technology and Countermeasures*, 1–58.

FBI (2017). Business e-mail compromise: Cyber-enabled financial fraud on the rise globally. Accessed on: May 12, 2017, https://www.fbi.gov/news/stories/business-e-mail-compromise-on-the-rise.

Felt, A. P., & Wagner, D. (2011). Phishing on mobile devices. In *Proceedings of the w2sp'11: Web 2.0 security and privacy*.

Felten, E. W., & Schneider, M. A. (2000). Timing attacks on web privacy. In *Proceedings of the seventh ACM conference on computer and communication security* (pp. 25–32). ACM.

Gelernter, N., & Herzberg, A. (2016). Tell me about yourself: The malicious CAPTCHA attack. In *Proceedings of the twenty-fifth international conference on world wide web* (pp. 999–1008). International World Wide Web Conferences Steering Committee.

Gibb, R. (2015). What is a transparent proxy?Accessed on: Apr. 28, 2017, https://www.maxcdn.com/one/visual-glossary/transparent-proxy/.

Gibbs, S. (2016). Dropbox hack leads to leaking of 68m user passwords on the internet. Accessed on: Apr. 21, 2017, https://www.theguardian.com/technology/2016/aug/31/dropbox-hack-passwords-68m-data-breach.

Goel, S., Williams, K., & Dincelli, E. (2017). Got phished? Internet security and human vulnerability. *Journal of the Association for Information Systems, 18*(1), 22–44.

Grobauer, B., Walloschek, T., & Stöcker, E. (2011). Understanding cloud computing vulnerabilities. *IEEE Security and Privacy, 9*(2), 50–57.

Gruschka, N., & Iacono, L. L. (2009). Vulnerable cloud: Soap message security validation revisited. *Proceedings of the IEEE International Conference on Web Services*, 625–631.

Gruschka, N., & Jensen, M. (2010). Attack surfaces: A taxonomy for attacks on cloud services. *Proceedings of the IEEE third International Conference on Cloud Computing (CLOUD 2010)*, 276–279.

Halevi, T., Memon, N., & Nov, O. (2015). Spear-phishing in the wild: A real-world study of personality, phishing self-efficacy and vulnerability to spear-phishing attacks. *SSRN*, 1–10. Accessed on Apr. 18, 2017.

Hao, S., Feamster, N., & Pandrangi, R. (2011). Monitoring the initial DNS behavior of malicious domains. In *Proceedings of the ACM SIGCOMM conference on internet measurement conference* (pp. 269–278). Berlin, Germany: ACM.

Hayashi, N. (2014). New phishing technique outfoxes site owners: Operation Huyao. Viewed on 27 March 2017, http://blog.trendmicro.com/trendlabs-security-intelligence/new-phishing-technique-outfoxes-site-owners-operation-huyao/.

Heartfield, R., & Loukas, G. (2015). A taxonomy of attacks and a survey of defence mechanisms for semantic social engineering attacks. *ACM Computing Surveys, 48*(3), 37:1–37:39.

Henderson, N. (2017). Hackers target docusign, bt customers with phishing emails. Viewed on 29 May 2017, http://talkincloud.com/cloud-computing-security/hackers-target-docusign-bt-customers-phishing-emails.

Hillebrand, F. (2010). The creation of the SMS concept from mid-1984 to early 1987. In F. Hillebrand (Ed.), *Short message service (sms): The creation of personal global text messaging* (pp. 23–44). John Wiley & Sons.

Hong, J. (2012). The state of phishing attacks. *Communication of the ACM, 55*(1), 74–81.

Huang, L.-S., Moshchuk, A., Wang, H. J., Schecter, S., & Jackson, C. (2012). Clickjacking: Attacks and defenses. In *Proceedings of the twenty-first usenix security symposium (usenix security 12)* (pp. 413–428). USENIX Association.

j2 Global, I. (2017). How efax works. Accessed on: Apr. 13, 2017, https://www.efax.com/how-it-works.

Jackson, C., Simon, D. R., Tan, D. S., & Barth, A. (2007). An evaluation of extended validation and picture-in-picture phishing attacks. In *Proceedings of the eleventh international conference on financial cryptography and first international conference on usable security* (pp. 281–293). Scarborough, Trinidad and Tobago: Springer-Verlag.

Jagatic, T., Johnson, N., Jakobsson, M., & Menczer, F. (2007). Social phishing. *Communications of the ACM, 50*(10), 94–100.

Jakobsson, M. (2007). The human factor in phishing. *Privacy & security of consumer inform. '07*. Viewed on 28 March 2017.

Jakobsson, M., & Stamm, S. (2006). Invasive browser sniffing and countermeasures. In *Proceedings of the fifteenth international conference on world wide web* (pp. 523–532). ACM.

James, L. (2006). Banking on phishing. In *Phishing exposed* (pp. 1–35). Elsevier Inc..

Kals, S., Kirda, E., Kruegel, C., & Jovanovic, N. (2006). Secubat: A web vulnerability scanner. In *Proceedings of the international conference on world wide web (WWW)* (pp. 247–256). Press.

Kaspersky (2017). Financial cyberthreats in 2016. *Technical Report*. Kaspersky Labs.

Khonji, M., Iraqi, Y., & Jones, A. (2013). Phishing detection: A literature survey. *IEEE Communications Surveys and Tutorials, 15*(4), 2091–2121.

Kim, D., & Kim, J. H. (2013). Understanding persuasive elements in phishing e-mails: A categorical content and semantic network analysis. *Online Information Review, 37*(6), 835–850.

Kovacs, E. (2016). Linux trojan takes screenshots every 30 seconds. Viewed on 26 April 2017, http://www.securityweek.com/linux-trojan-takes-screenshots-every-30-seconds.

Krombholz, K., Hobel, H., Huber, M., & Weippl, E. (2015). Advanced social engineering attacks. *Journal of Information Security and Applications, 22*, 113–122.

Kuosmanen, V. (2017). Browser autofill phishing. Viewed on 19 April 2017,e https://github.com/anttiviljami/browser-autofill-phishing.

LaForge, A. (2016). Flash and chrome. Viewed on 19 April 2017, https://blog.google/products/chrome/flash-and-chrome/.

Landwehr, C. E., Bull, A. R., McDermott, J. P., & Choi, W. S. (1994). A taxonomy of computer program security flaws, with examples. *ACM Computing Surveys, 26*(3), 211–254.

Lastdrager, E. E. H. (2014). Achieving a consensual definition of phishing based on a systematic review of the literature. *Crime Science, 3*(9), 1–10.

Leiner, B. M., Cerf, V. G., Clark, D. D., Kahn, R. E., Kleinrock, L., Lynch, D. C., et al. (2009). A brief history of internet. *ACM SIGCOMM Computer Communication Review, 39*(5), 22–31.

Leventhal, H. (1970). Findings and theory in the study of fear communications. In L. Berkowitz (Ed.), *Advances in experimental social psychology: 5* (pp. 119–187). New York: Academic Press.

Mahemoff, M. (2009). Explaining the "don't click" clickjacking tweetbomb. Viewed on 22 May 2017, http://softwareas.com/explaining-the-dont-click-clickjacking-tweetbomb/.

Marius, M. (2016). Instant messaging: Why it is so popular?Viewed on 17 April 2017, http://www.ict-pulse.com/2016/02/instant-messaging-popular/.

McCabe, J. (2016). FBI warns of dramatic increase in business e-mail scams. Online. Viewed on 3 April 2017, https://www.fbi.gov/contact-us/field-offices/phoenix/news/press-releases/fbi-warns-of-dramatic-increase-in-business-e-mail-scams.

McFedries, P. (2006). Technically speaking: Gone phishin'. *IEEE Spectrum, 43*(4), 80.

McIntosh, M., & Austel, P. (2005). Xml signature element wrapping attacks and countermeasures. *Proceedings of the Workshop on Secure Web Services*, 20–27.

Milletary, J. (2013). Technical trends in phishing attacks. *United State Computer Emergency Readiness Team (US-CERT)*, 1–17.

Mitnick, K. D., & Simon, W. L. (2002). *The art of deception: Controlling the human element of security*. New York: John Wiley & Sons.

Mohammad, R. M., Thabtah, F., & McCluskey, L. (2015). Tutorial and critical analysis of phishing websites methods. *Computer Science Review, 17*, 1–24.

Moreno, M. (2016). Malware as a service: As easy as it gets. Viewed on 4 May 2017, https://www.webroot.com/blog/2016/03/31/malware-service-easy-gets/.

Nagunwa, T. (2014). Behind identity theft and fraud in cyberspace: The current landscape of phishing vectors. *International Journal of Cyber-Security and Digital Forensics (IJCSDF), 3*(1), 72–83.

Nikiforakis, N., Balduzzi, M., Desmet, L., Piessens, F., & Joosen, W. (2014). Soundsquatting: Uncovering the use of homophones in domain squatting. In S. S. M. Chow, J. Camenisch, L. C. K. Hui, & S. M. Yiu (Eds.), *Information Security (ISC 2014), Lecture Notes in Computer Science* (pp. 291–308)). Cham: Springer International Publishing.

Niu, Y., Hsu, F., & Chen, H. (2008). iphish: Phishing vulnerabilities on consumer electronics. In *Proceedings of the first conference on usability, psychology, and security* (pp. 1–8). Berkeley, CA, USA: USENIX Association.

Ollmann, G. (2004). The phishing guide: Understanding & preventing phishing attacks. *Technical Report*. IBM Internet Security Systems.

PandaLabs (2015). PandaLabs' annual report 2015. *Technical Report*. Panda Security.

Patil, D. R., & Patil, J. B. (2015). Survey on malicious web pages detection techniques. *International Journal of u- and e- Service, Science and Technology., 8*(5), 195–206.

PayPal (2017). Paypal express checkout integration. Viewed on 11 May 2017, https://developer.paypal.com/docs/integration/direct/express-checkout/integration-jsv4/.

PhishLabs (2017). 2017 phishing trends & intelligence – Hacking the human. *Technical Report*. PhishLabs.

Purkait, S. (2012). Phishing counter measures and their effectiveness – literature review. *Information Management & Computer Security, 20*(5), 382–420.

Rader, M. A., & Rahman, S. M. (2013). Exploring historical and emerging phishing techniques and mitigating the associated security risks. *International Journal of Network Security & its Applications (IJNSA), 5*(4), 23–41.
Raskin, A. (2010). Tabnabbing: A new type of phishing attack. Viewed on 11 May 2017, http://www.azarask.in/blog/post/a-new-type-of-phishing-attack/.
Reeve, T. (2015). Style sheet vulnerability allowed attacker to hijack LinkedIn pages. Viewed on 4 May 2017, https://www.scmagazineuk.com/style-sheet-vulnerability-allowed-attacker-to-hijack-linkedin-pages/article/534883/.
Rietta, F. S. (2006). Application layer intrusion detection for SQL injection. In *Proceedings of the forty-fourth annual southeast regional conference* (pp. 531–536). ACM.
RightScale (2016). State of the cloud report. *Technical Report*. RightScale.
RSA (2016). A decace of phishing. *Technical Report,*. RSA FraudAction Intelligence.
Ruderman, J. (2016). Same-origin policy. Viewed on 25 April 2017, https://developer.mozilla.org/en-US/docs/Web/Security/Same-origin_policy.
Rydstedt, G., Gourdin, B., Bursztein, E., & Boneh, D. (2010). Framing attacks on smart phones and dumb routers: Tap-jacking and geo-localization attacks. In *Proceedings of the fourth usenix conference on offensive technology* (pp. 1–8). Berkeley, CA, USA: USENIX Association.
Sahoo, D., Liu, C., & Hoi, S. C. H. (2017). Malicious URL detection using machine learning: A survey. 1701.07179v2, 1–21, Viewed on 6 April 2017.
Schiller, C. A., & Binkley, J. (2007). Botnets overview. In C. A. Schiller, & J. Binkley (Eds.), *Botnets: The killer web app* (pp. 29–76). Syngress Publishing, Inc..
SecurityFocus (2003). Internet explorer URL parsing vulnerability. Viewed on 19 April 2017, http://www.securityfocus.com/archive/1/346948.
SecurityTracker (2003). Microsoft ie does not properly display some URLs. Viewed on 19 April 2017, http://www.securitytracker.com/id/1008425.
SecurityTracker (2004). (vendor issues fix) microsoft ie does not properly display some URLs. Viewed on 19 April 2017, http://www.securitytracker.com/id/1008905.
Shah, S. (2016). The history of social networking. Viewed on 17 April 2017, http://www.digitaltrends.com/features/the-history-of-social-networking/.
Singh, N. P. (2007). Online frauds in banks with phishing. *Journal of Internet Banking and Commerce, 12*(2), 1–27.
Sinha, A., Haddad, I., Nightingale, T., Rushing, R., & Thomas, D. (2006). Wireless intrusion protection system using distributed collaborative intelligence. In *Proceedings of the twenty-fifth IEEE international conference on performance, computing, and communications (IPCCC 2006)* (pp. 593–602). IEEE.
Song, Y., Yang, C., & Gu, G. (2010). Who is peeping at your passwords at starbucks? – To catch an evil twin access point. In *IEEE/IFIP international conference on dependable systems & networks (DSN)* (pp. 323–332). IEEE.
Sood, A. K., & Enbody, R. J. (2011). Malvertising – Exploiting web advertising. *Computer Fraud & Security, 2011*(4), 11–16.
Spring, T. (2017). Latest tax scams include phishing lures, malware. Viewed on 26 April 2017, https://threatpost.com/latest-tax-scams-include-phishing-lures-malware/124431/.
Stone, P. (2010). Next generation clickjacking: New attacks against framed web pages. *White Paper*. Context Information Security Ltd.
Suganya, V. (2016). A review on phishing attacks and various anti phishing techniques. *International Journal of Computer Applications - IJCA, 139*(1), 20–23.
Suri, R. K., Tomar, D. S., & Sahu, D. R. (2012). An approach to perceive tabnabbing attack. *International Journal of Scientific & Technology Research, 1*(6), 90–94.
Symantec (2011). Advanced persistent threats: A symantec perspective. *White Paper*. Symantec Corporation.
Symantec (2013). Internet security threat report 2013. *Technical Report*. Symantec Corporation.
Symantec (2016). Internet security threat report 2016. *Technical Report*. Symantec Corporation.
Trend Micro (2012). Spear-phishing email: Most favored APT attack bait. *Research Paper*. Trend Micro Incorporated.
TrendLabs (2012). Evolved Threats in a "Post-PC" World. *Technical Report*. Trend Micro Incorporated.
TrueSoftware (2017). Mypublicwifi: Turn your computer into a WIFI access point with firewall and URL tracking. http://www.mypublicwifi.com/publicwifi/en/index.html.
Trusteer (2008). In Session Phishing Attacks. *Technical Report*. Trusteer, Inc..
Tufts, A. (2012). How to protect your android against "smishing". Viewed on 3 May 2017, https://www.oneclickroot.com/how-to/how-to-protect-your-android-against-smishing/.
VeriSign (2012). 2012 iDefense cyber threats and trends. *White Paper*. VeriSign, Inc..
Vural, I., & Venter, H. (2011). Detecting mobile spam botnets using artificial immune systems. In G. Peterson, & S. Shenoi (Eds.), *Digitalforensics 2011: Advances in digital forensics vii: 361* (pp. 183–192). Berlin, Heidelberg: Springer.
W3C (2018). World wide web consortium (w3c). Viewed on 1 February 2018, https://www.w3.org/.
Wang, Y.-M., Beck, D., Wang, J., Verbowski, C., & Daniels, B. (2006). Strider typo–patrol: discovery and analysis of systematic typo-squatting. In *Proceedings of the second conference on steps to reducing unwanted traffic on the internet: 2* (pp. 31–36). USENIX Association.
Waziri Jr, I. (2015). Website forgery: Understanding phishing attacks & nontechnical countermeasures for ordinary users. In *Proceedings of the IEEE second international conference on cyber security and cloud computing (cscloud)* (pp. 445–450). IEEE.
Weins, K. (2016). Cloud computing trends: 2016 state of the cloud survey. http://www.rightscale.com/blog/cloud-industry-insights/cloud-computing-trends-2016-state-cloud-survey.
Winder, D. (2015). Cross-site scripting vulnerability uncovered in salesforce cloud. Viewed on 24 May 2017, https://www.scmagazineuk.com/cross-site-scripting-vulnerability-uncovered-in-salesforce-cloud/article/535535/.
Xing, X., Meng, W., Lee, B., Weinsberg, U., Sheth, A., Perdisci, R., & Lee, W. (2015). Understanding malvertising through ad-injecting browser extensions. In *Proceedings of the twenty-fourth international conference on world wide web* (pp. 1286–1295). International World Wide Web Conferences Steering Committee.
Xu, Z., & Zhu, S. (2012). Abusing notification services on smartphones for phishing and spamming. In *Proceedings of the sixth usenix conference on offensive technology* (pp. 1–11). Berkeley, CA, USA: USENIX Association.
Yadav, N., & Nagpal, B. (2016). Study on clickjacking attack. *International Journal of Emerging Research in Management and Technology, 5*(6), 37–41.
Yang, C., Song, Y., & Gu, G. (2012). Active user-side evil twin access point detection using statistical techniques. *IEEE Transactions on Information Forensics and Security, 7*(5), 1638–1651.
Zhou, S. (2015). A survey on fast-flux attacks. *Information Security Journal: A Global Perspective, 24*(4–6), 79–97.