A TAXONOMY OF PHISHING RESEARCH

BY

THASPHON CHUENCHUJIT

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Master of Science in Computer Science
in the Graduate College of the
University of Illinois at Urbana-Champaign, 2016

Urbana, Illinois

Adviser:

Associate Professor Michael Bailey

ABSTRACT

Phishing is a widespread threat that has attracted a lot of attention from the security community. A significant amount of research has focused on designing automated mitigation techniques. However, these techniques have largely only proven successful at catching previously witnessed phishing campaigns. Characteristics of phishing emails and web pages were thoroughly analyzed, but not enough emphasis was put on exploring alternate attack vectors. Novel education approaches were shown to be effective at teaching users to recognize phishing attacks and are adaptable to other kinds of threats. In this thesis, we explore a large amount of existing literature on phishing and present a comprehensive taxonomy of the current state of phishing research. With our extensive literature review, we will illuminate both areas of phishing research we believe will prove fruitful and areas that seem to be oversaturated.

In memory of Nunta Hotrakitya.

ACKNOWLEDGMENTS

I would like to express my deepest gratitute to Professor Michael Bailey for guiding this work from start to finish. I also greatly appreciate the essential assistance given by Joshua Mason and Zane Ma.

Finally, I wish to thank my parents for their love and support throughout my life.

TABLE OF CONTENTS

CHAPTER 1

INTRODUCTION

Phishing is a form of social engineering attack in which the attackers, termed "phishers", try to steal user information by impersonating an entity that users trust. The information targeted by phishers includes usernames, passwords, credit card numbers, social security numbers, and so on. The most common form of phishing involves sending an email to a user, with the sender claiming to be a trusted entity (e.g., financial institution). The email's message attempts to convince the user to perform some actions which typically begin with clicking an embedded URL that is included in the body of the email. Clicking on the URL leads the user to a website controlled by the phisher that claims to belong to the trusted entity. The website is often created to mimic the targeted entity's website in order to further deceive the user. Any information that the user provides to the website is captured and later exploited by the phisher.

Phishing is a common phenomenon. In 2007 alone, more than 3 million adults in the United States lost money to phishing attacks [1]. With its widespread nature, phishing attracts considerable attention from both the web security and behavioral sciences communities. At its highest level, research related to phishing can be split into two categories: works that study the characteristics of the attack and works that study mitigation techniques.

Research focused on attack characteristics analyzes the nature of phishing in order to identify elements of attacks that make them successful. These elements include both the techniques employed by the phishers to deceive the user and the characteristics of the users that make them more likely to fall for phishing attacks. Some works in this category also explore the ecosystem of phishers, studying the portion of their operation that happens before the phishing emails are sent and after the credentials are submitted by the victims.

Works that study mitigation techniques focus on the development of coun-

termeasures to reduce the success rate of phishing attacks. A large fraction of the studies in this area focus on the development of various detection mechanisms that can automatically identify phishing websites. With an efficient detection technique, phishing websites could be reported to ISPs to be filtered or taken down in a timely manner, thus reducing the number of possible victims. Another approach to mitigating phishing is to deploy tools on the user side that alert users when they are performing potentially unsafe actions. A number of education efforts also try to determine effective ways to teach users about phishing and how they can better defend themselves.

Traditional approaches to phishing education are ineffective. Typically, users are provided numerous security guidelines they are expected to memorize and there is no discussion of the risks involved. However, phishing education research has shown great promise. In one such work, researchers subjected users to controlled, ultimately harmless phishing attacks. Once a user has been thoroughly duped by the phishing attack and thus has clicked the link, users are shown training materials, thus taking advantage of the teachable (if embarrassing) moment. Alternative education resources such as games and comics were also presented and were shown to be more effective than traditional resources.

In this thesis, we explore phishing literature and present a taxonomy of the current state of phishing research. The taxonomy is comprised of literature that studies both attack characteristics and mitigation techniques, with a primary focus on phishing attacks in the form of spoofed emails and websites. However, since this is not the only technique that phishers employ to steal information from users, we also include a small portion of literature that studies other phishing strategies such as a DNS poisoning attack and attacks on mobile platforms. The rest of this thesis is organized as follows. in chapter 2, we discuss related work. In chapter 3, we explore research that studies different elements of phishing attacks. In section 3.5, we discuss work that studies phishing from the victim's perspective. Mitigation techniques are discussed in chapter 4, and user education approaches are discussed in section 4.4. Ethical approaches to conducting phishing experiments are discussed in chapter 5, and we conclude the thesis in chapter 6.

CHAPTER 2

RELATED WORK

Several authors have published surveys on the different aspects of phishing.

van der Merwe et al.[2] surveyed phishing attack characteristics and the actions and responsibilities that must be taken by users and businesses to prevent phishing attacks. The authors indicate a shift in responsibilities from businesses to users, as phishing presents a unique attack where security is no longer solely the responsibility of the businesses. Although the authors succinctly identified the characteristics of phishing and recommended mitigation techniques for users and businesses, the information presented in the paper is over ten years old, and phishing has evolved considerably during that time. While most of the information provided in the survey is still valid, the survey does not take into account new attack techniques such as the fast-flux attack, and some of the proposed recommendations were discovered to be ineffective in later literature.

Milletary[3] performed a literature survey of technical trends in phishing attacks and countermeasures. The author gave an overview of the tools and obfuscation techniques available to the phisher and proposed countermeasures. This survey provides a similar breadth of information to our work, but it lacks depth. The author mentioned many types of countermeasures in his survey but does not provide concrete examples or implementations of the proposed techniques. In our work, we survey the phishing mitigation systems that have been proposed and categorize them according to the type of protection they provide.

Huang et al.[4, 5] surveyed the landscape of deceptive phishing attacks and its countermeasures. In both of these works, the authors survey phishing countermeasures but lack the breadth of information present in our work. The authors studied many of the detection and defense techniques that are covered in our work, but they failed to discuss most of the machine-learning based detection strategies. They only briefly cover literature that studied

the behaviors of phishing victims, which we cover in greater depth.

A number of papers surveyed phishing defense and detection techniques. Zhang et al.[6] and Khonji et al.[7] analyzed both software and education based anti- phishing techniques. Alomani et al.[8] specifically surveyed phishing email filtering techniques, and Foozy et al.[9] created a taxonomy for phishing detection on mobile devices. These surveys are very comprehensive in terms of coverage of anti-phishing techniques, but they do not discuss the characteristics of the attacks or the targets with the same depth as our work. Specifically, in addition to a survey of mitigation techniques, our work discusses the elements of phishing attacks and the victims that could influence the success rate of the attack.

Most of the aforementioned surveys concentrate on phishing countermeasures, thus painting an incomplete picture of the phishing problem. The other surveys that focus on the breadth of the phishing problem do not discuss each aspect of the problem in detail, giving only an overview of the knowledge in the area. In this work, we provide a taxonomy of phishing research that encompasses both the characteristics of the attack and the mitigation strategies.

CHAPTER 3

ATTACK CHARACTERISTICS

In this chapter, we explore the characteristics of phishing attacks exhibited by the attacker. We explore various elements found in phishing emails that entice the user's trust and how they can act as tunable knobs that allow phishers to increase the effectiveness of their attack. An analysis of a class of phishing attacks that leverage contextual information about the victims is discussed, and an overview of the phishing operation that happen before phishing emails are sent out and after the credentials were submitted by the victim is presented.

3.1 Phishing Cues

In this section, we examine elements of phishing that have been determined in current literature to significantly affect the success rate of phishing campaigns. This includes elements from phishing emails, websites, and URLs. Researchers have identified these phishing elements of interest by analyzing phishing emails and websites, then determining how they affect the success rate of a campaign by constructing fake phishing emails or websites with elements of interest. The fake phishing websites are often evaluated by participants in a lab-based studies, and fake phishing emails can either be evaluated in lab settings or sent out to the participants in the same fashion as the attacker.

3.1.1 Email

The research in this category explores elements of a phishing email and determines how they affect the success rate of the email. A phishing email is considered successful if it is able to trick the user into clicking on the em-

bedded link. Some elements that were considered include the use of social context, urgency cues, personalization, and other real-world elements such as phone numbers.

Harrison et al.[10] conducted an experiment to determine if the perceptions of presence in a phishing attack would influence the victimization rate. In their experiment, the authors sent two types of phishing emails to undergraduate students at a university. Both types have the same message, but the first type of emails contained only the message while the second type is rich in contents. The content-rich email has the university's logo, two images that indicate other means of contact (a phone number and a Skype id), and a security indicator (DocuSign electric signature). Both fake phishing emails achieve a total of 68% victimization rate, with the content- rich phishing email having almost double the success rate of a plaintext phishing email.

Jagatic et al. [11] conducted a social phishing experiment in which phishing emails with spoofed sender addresses were used. The authors started by crawling social network websites to build a database of relationship between the participants, then crafted each phishing email such that each subject received an email that appears to be sent by one of their friends at the university. The authors were able to successfully phish 72% of the participants with the phishing emails that appears to be sent by a friend, compared to a 16% success rate for participants who received a phishing email from a fictitious university account.

Karakasiliotis et al.[12] assessed users' susceptibility to phishing by performing a survey where participants were shown 20 screenshots of emails and were asked to they think each email was legitimate. The result showed that the overall correct classification rate of 42%, along with 26% "don't know" responses. Qualitatively, the participants mentioned that elements such as personalization, grammatical and typographical errors, promotional offers, social proof, and indication of scarcity were influential factors in their decision.

Blythe et al.[13] performed four studies that investigate why people fall for phishing. They found that cues such as spelling and grammatical errors could not be relied on to detect phishing attack, and that phishing emails with logos have significantly lower detection rate in the online survey. The authors also found that blind people can often identify phishing in the first few lines of the email by detecting elements such as the omission of personal

address, spelling or grammatical errors, and an unlikely sounding premise. This result motivated the authors to perform a forth experiment, which draws on literary and critical theory to consider why some phishing strategy remain effective. They argued that the focus of individuals on banking and security services make phishing in this context effective as it plays neatly on the victim's anxieties in both form and content.

Wang et al.[14] developed a framework to explore phishing from the perspective of the victim. The authors identified five key design features of phishing attacks from phishing literature - email argument quality, email title, message appearance, website appearance, and assurance mechanism - and use them to perform a coding-based content analysis of phishing emails. Cluster analysis of the coded phishing emails revealed an evolutionary trend, with the data showing an improvement in quality of the attacks and a shift in target industries and target information over time. Other notable changes in the evolution of the design features of the email and website includes better use of indicators that induced trust in victims and the increase use of words that indicate urgency and impact in email's title and message.

Ferreira and Lenzini[15, 16] identify elements that reflect the effectiveness of phishing from other literature, and categorize them according to the principle of persuasion in social engineering (PPSE). The authors were able to identify 20 elements that constitute successful phishing emails and their associated principles from PPSE, and use them to guide a qualitative analysis of 52 phishing emails. They found that the Distraction principle is most prevalent in phishing emails, followed by Authority even when the most common elements from each categories were different. Distraction principle covered phishing emails that trick victims into focusing on what they could gain or possibly miss out on if they do not act immediately and Authority principle covered emails that use authoritative tones.

Downs et al.[17] conducted a laboratory study where they interviewed 20 non-expert computer users to gain insight into their decision strategy when they encountered suspicious email. The study consisted of two segments: a role-play segment where the participants role-play as someone who is going through their email, and a survey segment where the participants were asked to describe their own online behaviors. Overall, all participants in the study noted various security cues that can be use to determine the legitimacy of a website, such as a lock icon or broken images on web pages, but they did

not necessarily have the skills needed to interpret these cues appropriately. Participants also employed different strategies to make decisions about the trustworthiness of email, but most of these strategies rely on the interpretation of the message rather than the cues in the email's header.

Tsow and Jakobsson[18] conducted an investigation of trust manipulation tactics used by phishing email and web sites. The participants in the experiment were shown 6 email screenshots followed by 6 web page screenshots and were asked to rate their authenticity or phishiness on a five point Likert scale. The results showed that third party endorsement and glossy graphics are effective as authenticity simulators when the email content was short and unsurprising but fail to be significantly effective when applied to a more involved message. One surprising result was a great increase in trust caused by a small legal footprint in an email that already exhibited strong personalization.

Jakobsson[19] performed a qualitative study where the participants speak out their thoughts as they rated phishing stimuli and attempted to determine what caused the subject to decide on the trustworthiness of stimuli. He found that the appearance, including spelling and grammar, URL, personalization, and padlock icons matter to users in determining the authenticity of messages. Participants were suspicious of messages with bad design or grammatical errors, and detect IP addresses in URL as being illegitimate. Personalization and padlock icons in the body of the message increased the trustworthiness of the message as well as the use of brand name endorsement such as Verisign and suggestion of independent channels such as a telephone number. Also, participants often decided the legitimacy of a stimulus based on the content before checking its authenticity; participants reject stimuli that offered monetary rewards or requested credentials but considered email that only contain information as safe. Further, the participants considered email stimuli to be more phishy than web pages, but considered phone calls to be safe.

Overall, many factors were identified in literature that can be used as knobs to adjust the effectiveness of a phishing campaign. Many results suggest that users tend to reject emails that have bad design or have spelling or grammatical errors as being illegitimate, so one of the most important aspects of any successful phishing email is to have a message that is well designed and has little to no errors. Email with messages that convey a sense of urgency or

have an authoritative tone are effective at enticing users to perform security compromising actions. Some elements that can be added to the emails to improve their legitimacy in the eyes of the users are the logo of the target company, lock icon, brand name endorsement (e.g. VeriSign and DocuSign), and legal footprints. The use of real-world content such as phone numbers and personalized greetings as well as social contexts are also found to be effective.

3.1.2 Website

The works in this category aim to identify elements of a phishing website that entice trust from the users. Specifically, the works in this area try to identify the elements that users look for when they are evaluating the legitimacy of a website, and how phishing websites exploit them to trick the user into believing that they are on a legitimate website.

Fogg et al.[20] conducted an online study to investigate how different elements of a website affect a user's perception of credibility. The results indicated that the most effective way to enhance the credibility of a website is to include elements that convey the "real world" aspect of an organization such as physical address and photographs of employees. Additionally, ease of use of the website and the inclusion of elements that express expertise and trustworthiness were also key components of a credible website. However, overly commercial elements and signs of amateurism such as typographical errors and broken links can be damaging to a site's credibility. In another study, the authors conducted an experiment to determine the features of a website that get noticed when people evaluate its credibility[21]. The results indicate that most participants use the website's design, structure, and information presented to assess the credibility of a website. Other participants also mention using a company's motive, usefulness and accuracy of information, name reputation, and advertising as factors in evaluating credibility.

Dhamija et al. [22] ran an experiment in which 22 participants were shown 20 websites and asked to determine which ones are fraudulent. They found that the participants who only look at the website's content performed significantly worse than the participants who also examined other security features presented by the browser. They also found that naive participants made in-

correct judgments because of lack knowledge about computer systems and security indicators, and more experienced participants were tricked by visual deceptions such as a spoofed URL or SSL indicator.

Grazioli et al.[23] conducted an experiment that asked the participants to determine if a website is trustworthy. In the experiment, the participants are split equally into two groups, with one group viewing a real commercial website and the other group viewing a fake website built by the authors. The result suggested that, on average, the participants cannot discriminate between the legitimate website and the fake website, and that successful participants noticed significantly fewer deceptive cues than the unsuccessful participants. The authors believed that this is a result of the successful participants finding conclusive evidences that the website is a deception and stop searching. Overall, they found that successful participants rely on assurance cues (warranties and trust seals) and tend to discount trust cues (customer endorsements), which is the opposite of the unsuccessful participants.

Tsow and Jakobsson[18] conducted an investigation of trust manipulation tactics used by phishing email and web sites. The participants in the experiment were shown 6 email screenshots followed by 6 web page screenshots and were asked to rate its authenticity or phishiness on a five point Likert scale. The result showed a clear preference for a simulated web page whose domain name matched its content over a genuine page whose domain only weakly connected to the same content. The experiment also verified that overuse of security notices have significant negative effect on genuineness. Lastly, one of the most trusted stimuli in the study is a third party web page created to handle embarrassing incidents (a hardware recall) for their corporate clients, which showed that the context of the message enticed more trust in spite of the message's poor personalization, illegitimate URL, and relatively simple layout.

In general, researchers have identified many website elements that entice trust in the user. Real world aspects of the organization such as physical address and photographs of employees as well as the company's motive and the accuracy of the provided information make a website appears more trustworthy to users. Ease of use of a website and signs of professionalism are also important. Elements such as overly commercial elements, typographical errors, and broken links and images can be damaging to a website's credibility. Assurance cues such as warranties and third party trust seals are reported to

entice trust in advanced users, while trust cues such as customer endorsement are more heavily relied on by novice users. The choice of domain name is also important, as users showed a clear preference for a fake web page whose domain name matched its content over a genuine page whose domain only weakly connected to the same content. One experiment also verified that overuse of security notices have a significant negative effect on the perceived genuineness of a website.

3.1.3 URL

The research in this category aim to identify characteristics of phishing URLs and identify techniques employed by the phishers to make phishing URLs appear to be legitimate. Specifically, this includes the techniques employed by the phishers to construct phishing URLs, and techniques that were employed to entice the users to click on the URL.

Spaulding et al.[24] reviewed the landscape of domain name typosquatting, which are domain names that uses typographical variants of other domains deliberately for malicious purposes. These domain names are generated in such a way that exploit common typographical errors made by users, with some of the most common generation models being missing dot typos (e.g. wwwexample.com), character permutation, character substitution, character duplication, and character omission. An extended model would include all domain names with a Damerau-Levenshtein edit distance of one from the target domain name, and a more technically advance techniques exploited random bit errors to redirect victim to a domain name with a hamming distance of one bit from the target domain (e.g. mic2osoft.com). The top-level domain(TLD) portion of a domain name may also be a target for exploitation, where one .com domain name may have a malicious .org counterpart.

Waziri[25] surveyed different types of website forgery techniques, phishing attacks, and their countermeasures. The author identifies that phishing websites can be hosted on various web hosting services or by hijacking other websites or devices, discusses various obfuscation techniques employed on the phishing websites and its countermeasures. These techniques include various URL and domain name obfuscation techniques such as domain typosquatting and inclusion of the target's domain in the subdomain or path, and the use

of images instead of text to display spoofed URLs.

Downs et al.[26] conducted a survey designed to measure user's behavioral response to phishing across a large population. The survey consisted of several sections: email role play, URL evaluation, warning messages reaction, computer knowledge, past web experience, and negative consequences. Overall, the authors found that the better knowledge of web environments and URL structure predicts lower susceptibility to phishing attacks, but does not increased false positive responses to legitimate emails. This implied that users who do not know how to parse URL is more susceptible to the attack.

McGrath and Gupta[27] examine the phishing modus operandi by analyzing phishing URLs from PhishTank[28] and MarkMonitor[29], legitimate URLs from DMOZ, and supportive information from WHOIS and zone files of 4 generic TLDs. DMOZ is a large open- content directory of user-submitted URLs. They found that the distribution of the length of phishing URLs and domain names are different from the legitimate URLs in DMOZ, namely the length of the phishing URLs peak at a larger number of characters, while the phishing domains tend to be shorter than regular domains in DMOZ. Furthermore, the distribution of letters in DMOZ's domains closely resembles the English character distribution while phishing domains have smaller peaks at each of the vowels and fewer number of unique characters. Furthermore, phishing URLs often contain the name of the brand that they are spoofing, and some phishers are abusing the URL shortening services. From the domain registration perspective, most domains registered for the purpose of phishing are put to use almost immediately, and their lifetime on average is a little over three days.

Chhabra et al.[30] analyzed the use of URL-shortening services to obfuscate phishing URLs. Using data from Twitter, bit.ly, and PhishTank, the authors found that the space gain for most phishing URLs are smaller than those of generic URLs, suggesting that the URL shorteners are primarily used by phishers not to shorten the length of the URL, but to hide the identity of the actual phishing URL. Analysis of referrals for the bit.ly URLs that were identified as phishing also showed many websites other than Twitter that are referrer of the URL. Since the other websites did not impose a limit of message length, this finding further affirmed the hypothesis that URL shortening services are being use to hide phishing URLs. They also found that online social media brands are targeted more than financial institutions and

e-commerce websites on Twitter, suggesting a change in target for phishers.

Fu et a.[31] proposed a phishing strategy based on the use of Internationalized Resource Identifier (IRI). In IRI scheme, the glyph of many characters have similar appearance although their underlying Unicode values are different. This would be beneficial to the phisher, as it allowed for a construction of URIs that are visually similar to the phisher's target, but resolve to a different resource controlled by the phisher.

Overall, phishers employ many URL obfuscation techniques to trick the users. Phishing URLs often include the domain name of their target in the URL, either as a subdomain or in the path, to trick users who do not have a good understanding of URL structure. Typosquatted domain are also common, and some research has raised awareness about the use of internationalized resource identifier to construct phishing URL that is visually similar to the target URL. URL shortening services are found to be abused to obfuscate phishing URLs, and phishers employed techniques such as the use of images of URLs to cover up the real destination of the embedded link.

3.1.4 Summary

In conclusion, the works discussed in this section have identified many knobs that the phishers could use to adjust the effectiveness of their phishing campaign. The factor that was highlighted the most is that an email or a website that is not well-designed and exhibit signs of amateurism such as typographical or grammatical errors are almost always regarded as suspicious by the users. For email, messages that convey a sense of urgency or have an authoritative tone are more effective at enticing the users to perform some action, and factors such as brand name endorsement, lock icon, and logo of the target company can make the message appears more genuine. For the website, content that exhibit the real world aspect of the target such as a fake physical address, a fake phone number, or pictures of employee as well as customer endorsement can increase the website's credibility. Having a URL that matches the content of the website can also make it appears more genuine, and phishers are known to use technique such as typosquatting, URL shortener, and inclusion of the target domain name in the phishing URLs to trick the users.

3.2 Spear Phishing

Spear phishing attack, also known as context aware phishing attack, is a form of specialized phishing attack where the phishing email is crafted with contextual information about the target. A spear phishing attack against a user may leverage information such as the user's name and address to create an email that is personalized to the user. Against a company, a spear phishing email may leverage information such as the employee names or current events in the company to make the phishing email appears legitimate. The works in this section study the success rate of spear phishing attacks and techniques that can be leverage by an attacker to find contextual information.

Jakobsson[32] introduced a visualization tool to model and describe threats on a complex system and used it to describe context aware phishing attacks. The visualization tool models a phishing attack by a directed graph where vertices represent knowledge or access rights. An edge from vertex u to vertex v represents a mean of obtaining information or access right in vertex v given the information available in vertex u. Each edge also has a weight assigned to it to represent the probability of a successful attack. The author then used the model to describe context aware attacks targeting eBay and online banking system.

Jakobsson and Stamm[33] discussed browser sniffing attack and proposed countermeasures. Browser sniffing attack is an attack in which an attacker extracts information about websites that a user visited from his browser's cache and history. These information are then used to launch context aware phishing attack at the user. The authors proposed two approaches to defend against browser sniffing: cache pollution and URL customization. Cache pollution purposefully adds a set of URLs to the browser's history for each URL that a user visited, and URL customization utilizes a proxy server to personalize all URLs before sending them to the browser. The authors argue that this scheme provides perfect privacy for internal pages of a protected domain, and some privacy for the entrance pages. Jakobsson also proposed a server-side security techniques called Remote-Harm Detection[34]. RHD probed the client's browser history in the same manner to gauge whether the client may be infected with malware. However, browser sniffing vulnerability was recognized and patched by most browser vendors, so we only include it here for information purposes.

Gupta et al.[35] explored the feasibility and scalability of a phishing attack that abuses phone numbers. The authors enumerated 1.16 million phone numbers, then for each number looked up the owner's information in TrueCaller. TrueCaller is an application that allows a user to search for owner information of a phone number. If a Facebook ID was found in the owner information, the system used it to crawl for public information of the owner. The authors also used Whatsapp, a cross-platform mobile messaging application, as a separate vector by leveraging its address book syncing feature, which can automatically connect the user with other Whatsapp users whose phone number is in the address book. By aggregating the information from these sources, the authors were able to identify 51,409 users who could be the target of social phishing attacks, 180,000 users who could be the target of spear phishing attacks, and 24,464 users who could be the target of normal phishing attacks. The authors also conducted a role-play experiment on Amazon Mechanical Turk to determine the effectiveness of a phishing attack over a messaging application such as Whatsapp. They found success rates of 34.5%, 54.3%, and 69.2% for non-targeted phishing, spear phishing, and social phishing attack, respectively.

Dodge et al. [36, 37] tested the efficacy of their information awareness training by conducting an email phishing exercise on the student body of the United States Military Academy. The average rate at which the students behaved incorrectly was found to be 40% overall. Surprisingly, execution of malicious attachments was shown to be a more successful vector than clicking on an embedded link. The authors also noted that some students reported that they found the email to be suspicious, but since it was signed by a colonel they decided to carry out the actions. This illustrates a possible vulnerability in the institution and indicates that the results may be biased because of the students' mindset.

Jagatic et al. [11] conducted a social phishing experiment in which phishing emails with a spoofed sender address were used. The authors reported a 72% success rate for emails with a spoofed sender, compared to a 16% success rate for generic phishing emails.

Steyn et al.[38] conducted a naturalistic phishing exercise where phishing emails were sent to 400 staff members of a university in South Africa. The emails asked the recipient to confirm the details of his account by clicking on a link in the email. The exercise was conducted after a recent implementa-

tion of a new system at the university, which provided a great opportunity for a spear phishing exercise. The results showed that an alarming 53% of the recipients gave away information to the fake website, and an additional 13% clicked on the link but did not provide information. Only 7.2% of the recipients reported the incident to the university's ICT staff members.

Mohebzada et al.[39] conducted two large scale naturalistic phishing experiments. Phishing emails were sent to students, faculty members, staffs, and alumni totaling over 10,000. One phishing email was spoofed to be sent from the university's IT department and asked the recipient to reset his university password. The other email was spoofed to be from a "research group" and asked the recipient to participate in a survey about his banking experience and financial information. The researchers were able to get a 8.74% success rate for the first experiment and a 2.05% success rate for the second experiment.

Holm et al.[40] performed two naturalistic phishing experiments targeting a Swedish technology consulting company in the electrical power domain. The first experiment simulated a generic phishing attack, and the second experiment simulated a spear phishing attack. The results from the two experiments indicated that while the context aware phishing attack got more victims, it also attracted significantly greater attention. In this case, everyone in the company knew about the intent of the context aware email within 9 minutes, while the traditional phishing email was not discussed or reported at all.

While many of the experiments listed here did not provide a comparison between the success rate of a spear phishing email and that of a normal phishing email, the trend appears to be that spear phishing emails can succeed against significantly more victims. Interestingly, results from one experiment indicate that while spear phishing emails attracted more victims, they were also considerably louder. This implies that although a spear phishing attack allows the attacker to steal more credentials in a shorter amount of time, context free phishing emails may be more suitable when the goal of the attacker is to launch an attack without being noticed. Other works also demonstrate that techniques such as browser sniffing and phone number enumeration can be used to gather contextual information about the targets.

3.3 Other Attack Vectors

In this section, we discuss works that study the characteristics of phishing attacks that do not involve the use of phishing emails. With the widespread adoption of smartphones, researchers have discovered new phishing attack vectors that leverage characteristics of mobile platforms such as the smaller screen and various interactions between applications that are unique to the platform. DNS poisoning is also discussed as an attack vector that direct the user to the phishing website without requiring phishing emails.

3.3.1 Mobile Devices

New attack vectors were introduced with the widespread use of smartphones. Smaller device screen sizes as well as services such as notification systems and context switching between applications introduced unique new phishing attack vectors in this platform that did not exist in traditional desktop browsers. The works in this section survey the behaviors of smartphone applications and notification systems, and identify ways in which they can be exploited to launch phishing attacks.

Felt and Wagner[41] conducted a systematic analysis of the ways in which applications and websites on a mobile device link to each other and evaluated how these interactions can increase the risk of phishing attacks. By manually analyzing control transfers between applications, they found that for many of these control transfers, the website or application that received the transfer would prompt the user for information such as passwords or payment information. This was the case because most of the transfers directed the user to social media platforms or application stores, which required the user to enter his password to authorize access or payment. This process habituated the user to expect to re-enter this information even after control transfer had occurred, making them susceptible to phishing attacks where the sender application visually spoofs the control transfer to capture the user's information. The authors also discussed a man in the middle attack where a malicious application could register itself as a default handler for certain control transfer schemes and capture the user's input before transferring them to the legitimate application.

Xu and Zhu[42] analyzed the vulnerability of notification services on smart-

phone operating systems and demonstrated how a malicious application could create anonymous phishing or spam notifications. Notification on each smartphone operating system was customized to various degree, which allowed a malicious application to abuse these customizations and send anonymized phishing notifications to the user. To illustrate this, the authors implemented an application that created a notification that was visually identical to a notification created by the Facebook application. Once a user clicked on the notification, the malicious application presented the user with a fake Facebook login page, captured the input, terminated itself, and then launched the legitimate Facebook application. To prevent this attack, the authors proposed that the smartphone OS should reserve a portion of each notification to display the icon and name of the application that created the notification and log all notification creation events and creators.

Niu et al.[43] studied the Safari browser on the iPhone and discovered vulnerabilities that exposed users to website spoofing attacks. Specifically, the browser's chrome on the Safari browser is simple and immutable, allowing the phisher to easily spoof it on the phishing website. The browser's chrome also automatically hides when the user scrolls down the web page, an action that can be exploited by malicious JavaScript code to hide the chrome. Additionally, The address bar in the Safari browser shows a truncated version of a long URL by using ellipses, allowing the attackers to visually spoof a legitimate URL by putting the target domain name as a subdomain. The result of a small scale user study indicated that even expert users who could identify all the phishing websites on a desktop failed to notice the spoofed browser's chrome, and many of them disregarded the scrolling problem as a bug in the browser.

In general, the works in this section highlight the need for better security in mobile systems. Since users are trained to expect authentication dialogues when a context switch occurs between applications, defense mechanisms need to be put in place to ensure that users are interacting with a legitimate application. The notification authentication issue also needs to be addressed in a way that allow users to easily identify the real creator of a notification without harming the aesthetic nature of notification customization. Lastly, browsers in smartphone need to adapt many security indicators from their desktop counterparts to display effectively on smaller screen sizes.

3.3.2 DNS Poisoning

DNS poisoning, sometime referred to as pharming, is an attack where the entries in a local DNS server are manipulated to direct users to a locally hosted phishing website. The attack consists of a rouge access point with a local DNS server setup to direct users of a target website to a phishing website that is hosted locally at the access point. This limits the scope of victims to users who are connected to the access point, but it also bypasses the need to send out phishing emails.

Abu-Nimeh et al.[44, 45] proposed an attack to bypass browser's security toolbars and phishing filters via local DNS poisoning. The authors evaluated the attack against five security toolbars and browser filters, and they were able to verify that all the tested security tools were unable to detect the attack, and some even confirmed that the users were at legitimate websites. The authors also noted that since the phishing websites were hosted locally, it was impossible for web crawlers to detect the website and issue a blacklist or a takedown, and there was no need to send phishing emails. To counter this attack, Kim and Huh[46] proposed a phishing detection framework through a new heuristic based on network performance characteristics of the websites. Four aspects of the routing information were considered: the use of a firewall, the mean round-trip time (RTT) of all hops, the local route length in hops, and the total route length in hops. Using routing data for 50 legitimate websites and 500 phishing websites, the authors evaluated multiple machine learning classifiers and found that K-nearest neighbors performed the best with a true positive rate of 99.4% and a false positive rate of 0.7%.

While pharming is its own category of attack, this group of works illustrate how DNS poisoning attack techniques can be utilized for phishing purposes. DNS poisoning attacks allow the attacker to evade many defense mechanisms and direct a specific group of users to a phishing website without the need for phishing email. A defense mechanism against DNS poisoning attack based on network performance analysis is proposed, and the preliminary results are encouraging.

3.3.3 Distributed Phishing Attack

Jakobsson and Young[47] described a novel type of phishing attack called Distributed Phishing Attacks (DPA), with the main characteristics being a per-victim personalization of the website that collects information from the victim and covertly transmits that information to a hidden coordination center runs by the phisher. In this context, the per- victim personalization implies that each victim will be directed to a phishing website hosted on a unique machine with different owner and location, and information about the location of a phishing website cannot be inferred from another phishing website.

3.3.4 Summary

In this section, we have discussed research efforts that identify new attack vectors for phishing. Multiple characteristics of the mobile operating system were identified that could be exploited to launch a phishing attack, and a spoofing attack on a browser on a mobile platform was demonstrated. DNS poisoning attacks were shown to be able to bypass many browser and toolbar defenses, and a countermeasure based on analysis of network characteristics was proposed.

3.4 Behind The Scene

The research in this section explores the characteristics of phishing that happen behind the scenes. Namely, the research in this category aims to answer the questions about phishing that happens before phishing emails are sent out and also after the victims enter their credentials into phishing websites. Some of these questions include “What is the average lifetime of a phishing campaign?”, “Where do phishers host their phishing websites?”, “Where do credentials go after a victim has submitted them?”, and “How do phishers get paid?”.

3.4.1 Lifetime of Phishing Websites

The works in this category study the lifetime of phishing websites and its relationship to the volume of phishing emails.

Moore et al.[48] analyzed the temporal relationship between the sending of spam emails and the lifetime of phishing websites by using spam data from IronPort's SpamCop[49] and phishing website data from PhishTank, APWG[50], and other brand owners. They found that fast-flux attacks accounted for only 3% of distinct campaigns, but accounted for 67% of the total observed spam volume, suggesting that they are a far more serious threat. Fast-flux is a term coined in the phishing community to describe a decentralized botnet with constantly changing DNS records, which are used by the attacker as proxies to hide the real phishing website. This involves the use of domain names that resolve to a large number of constantly changing front proxies to hide the "mothership" that hosts the actual phishing website. Fast- flux attackers also manage their spam campaigns more efficiently, sending out a large volume of spams before the website is discovered and stopping shortly after its removal. Moore's data indicates that website take-down is necessary, as long- lived phishing websites continue to send out spam emails until they are taken down.

Moore and Clayton[51, 52] examined empirical data on phishing website removal time and the number of visitors that a website attracted. The authors gathered a list of reported phishing websites from PhishTank and extracted their visitor statistics from Webalizer, which provides publicly available web page usage information and is usually set up by default on the type of web servers that seem to be regularly compromised. The authors found that the lifetime of phishing websites followed a lognormal distribution, with the fast-flux domains having a greater average lifetime than normal phishing domains, and that phishing websites hosted on free-hosting webspaces have a shorter lifetime than regular phishing websites. They also found that the lifetime of phishing websites have a long tail, with one website being online for over seventeen weeks. This serves to illustrate that even when take-down is happening slowly, it can still reduce the damage done.

In short, while fast-flux phishing campaigns are low in number, they account for more than two-thirds of total phishing email volume, and they have a greater average lifetime than normal phishing websites. This suggests

that fast-flux campaigns are run by well-organized phishers, and efficient detection methods for fast-flux campaigns can drastically reduce the volume of phishing emails. Data also suggest that take-down of phishing websites is critical, as phishers continuously send out phishing emails as long as the phishing website is still live.

3.4.2 Locating Vulnerable Hosts

Phishers often utilize free hosting services to host their phishing websites. However, in order to launch a big campaign or to start a fast-flux attack, phishers need access to a large number of hosts spanning across different ISPs. To accomplish this, phishers need to scan the Internet for vulnerable hosts and exploit them to host the phishing web pages or proxies. The works in this section study the methods that phishers use to identify vulnerable hosts and exploit them to host phishing websites.

Moore and Clayton[53] studied the use of search engines to locate potentially vulnerable hosts that can be exploited to host phishing websites. These 'evil searches' were categorized by the researchers into three distinct types: vulnerability searches that looked for a particular version of a program that is vulnerable to attacks, compromise searches that looked for existing phishing or other compromised websites, and shell searches that looked for PHP 'shells'. By analyzing feeds of phishing website URLs from both major brand owners and other sources including APWG and PhishTank together with web access logs from Webalizer-equipped websites, the authors found a consistent pattern of evil searches appearing in web logs at or before the time of reported compromise, with approximately 17.6% of hosts in their sample having evil search terms in their logs. Furthermore, the authors found that many hosts had multiple phishing websites on them, which suggests that these compromised hosts may be re-compromised, and that hosts reached by evil search face a 21% chance of re-compromise after 4 weeks compared to 14% otherwise. This suggests that vulnerable websites that can be found through search engines are likely to be repeatedly exploit until they are patched. Some mitigation strategies reviewed by the authors include obfuscation of server's application details, evil search penetration testing, blocking of evil search terms, removal of phishing sites from search results, and lowering of

reputation of phished hosts. The authors also reported that phishing website URLs that are made public by the PhishTank database had a statistically significant reduction in their re-compromise rates, suggesting that defenders were able to use this information to reduce criminal attacks[54].

Search engines are a powerful tool, and they are leveraged by phishers to identify vulnerable hosts that can be exploited to host phishing websites. Results from these studies also suggest that vulnerable hosts are exploited repeatedly by multiple phishers to host different phishing websites, and that vulnerable hosts that are reached by evil searches are more likely to be re-compromised over time. Vulnerable hosts that are publicly identified by blacklists are also found to have significantly lower re-compromise rates, suggesting that these public announcements can notify defenders to secure their machines. Some mitigation strategies against evil searches are also proposed.

3.4.3 Dropboxes

After victims submit their credentials to a phishing website, the credentials are often transmitted to the phisher via email. This seems to be the standard method for the phishers who utilize phishing kits, which are comprehensive ready-to-deploy phishing websites. Novice phishers obtain these phishing kits and modify a few lines of code to specify the destination email addresses for the credentials. The email accounts that were setup to receive the stolen credentials from these phishing websites are dubbed dropboxes. The works in this section attempt to identify phishing dropboxes and use them to infer information about the phisher.

Cova et al.[55] studied phishing kits and identified the mechanisms that phishers use to transmit stolen credentials from a phishing website. They also investigated obfuscation techniques that were employed by the kit's creator to hide backdoors. The authors retrieved 353 kits from distribution sites and extracted 150 kits from live phishing websites. They identified 129 kits from distribution sites and 61 live kits that were backdoored to send all stolen credentials to the kit's creator as well. All of the phishing kits were written in PHP and utilized the mail() function to send the stolen information via email to the phisher. The authors also found that the creator's email address in the

backdoored kit was obfuscated by using different encodings or was retrieved from a remote site. The code that was utilized to send the stolen information to the creator was also obfuscated and social engineered by comments such as "// do not change anything here".

Zawoad et al.[56] proposed a clustering algorithm to reveal the relationship between phishing websites based on common dropboxes. By clustering phishing websites based on shared dropboxes, the authors were able to determine the strongest and most pervasive phishers and kit creators as well as the relationship between the kit creators and kit users. This relationship showed that a number of kits were used by a large number of phishers, with each phisher targeting a different website.

Moore and Clayton[57] proposed a technique to identify phishing dropboxes and their associated phishing websites by leveraging a list of known phishing websites and email metadata maintained by an email provider. To detect phishing dropboxes, the authors submitted trackable fake credentials into 170 phishing websites targeting PayPal and monitored the mail provider's email metadata database for the submitted credentials. The authors found 17 distinct dropboxes receiving credentials from 28 phishing websites and indirectly uncovered 24 additional dropboxes by examining other emails with similar subject lines. The authors also proposed a technique to derive the phishing website that controlled the dropbox, dubbed an "intersection attack", by finding the intersection of the URLs that the victims received via email prior to their credentials appearing in the dropbox.

The results from these studies suggest that phishers often run more than one phishing campaign at a time, as evidenced by the number of credentials being sent to different dropboxes. Most of the phishing kits were also found to be backdoored, allowing the creator of the phishing kits to collect the stolen credentials from the phishing website deployed by a kit user, essentially receiving all the benefit and letting the phisher shoulder all the responsibility. By analyzing the data about the dropboxes, a detection mechanism was proposed that utilized the intersection of the set of URLs received by two victims to identify the phishing URL.

3.4.4 Phishing Economy

The research in this section explores the economy of phishing, specifically highlighting the chain of events that occur after a phisher obtains a stolen credential from a victim. This includes a discussion of the phisher's marketplace and the phisher's "cashout" method. Application of economic theory to the phishing ecosystem was also conducted to verify if phishing is indeed a path to riches.

Yu et al.[58] performed a root-cause analysis of the methods used in phishing and the motivation behind the attack, and created fishbone diagrams outlining the cause and methodologies of phishing. The authors identified that the motivation of the phishers is primarily financial gain, with other motivating factors including identity theft, identity trafficking, industrial espionage, malware distribution, password harvesting, fame, and notoriety.

Abad[59] studied the economic and social environment of phishing networks. By analyzing phishing emails, messages from phishing-related chatrooms, and chatroom networks, the author presented an analysis of phishing infrastructure beyond what was normally observable by victims. By crawling IRC, the authors uncovered a maximum spanning tree of chat channels related to phishing, which were used by phishers to buy and sell stolen credentials, as well as to control botnets. The author also presented a flowchart of the phishing process, starting from the discovery of vulnerabilities and ending with cashing out. Cashing out was either done through the selling of the stolen credentials or through the services of cashers, who often play no role in the obtaining of credentials but have the capability to obtain cash using the stolen credentials. The author indicated that the preferred cashing out method for stolen banking credentials was ATM fraud, where the casher would encode the banking information onto an ATM card and withdraw the maximum daily funds from the account.

Herley and Florêncio[60] argued that phishing is far from a path to riches and appears to be a low-skill low reward business. The authors showed that the economics of phishing are subject to the tragedy of the commons, where the pool of dollars (reward) shrinks as a result of the efforts of the phishers. The author applied open access economic theory to phishing, and found that the total dollars lost to phishing is equal to the total cost in terms of income opportunity that phishers gave up in other occupations. In other words, the

average income for a given phisher is estimated to be equal to or slightly less than what he would have made at another occupation that required the same level of skill. Another point that the author made was that as phishers put more effort into the endeavor, the total revenue falls rather than rises, suggesting that increasing volumes of phishing efforts imply a decreasing total revenue for phishers.

Overall, the monetary gain in the phishing ecosystem is based on supply and demand, in which phishers sell the harvested credentials to the cashers, who then use those credentials to withdraw money from banks. Economic theory suggests that the economy of phishing is subject to tragedy of the commons, where more effort put into phishing results in a decrease in the pool of resources, which in this case is the amount of phishable dollars. Open access economic theory also predicts that, on average, the income for a given phisher is no better than what he would have made in other honest occupation.

3.4.5 Others

The following literature discuss other characteristics of phishing that do not fall into the categories above.

Soni et al.[61] showed how a phisher could easily construct a phishing website by downloading the source code of the target web page and making a small modification to create a phishing web page with PHP. The authors also discussed how a chromeless popup can be used to obfuscate the location bar and security indicators in a browser's chrome.

Bursztein et al.[62] studied manual account hijacking and provided evidence supporting the hypothesis that phishing is the primary way for manual hijackers to steal credentials. In order to measure hijackers' response times, the authors manually submitted 200 fake credentials into a random sample of 200 phishing pages that explicitly ask for Google credentials and found that 20% of these accounts were accessed within 30 minutes of submission and 50% were accessed within 7 hours. Surprisingly, the hijackers spent on average 3 minutes to assess the value of the account, and did not attempt to exploit accounts that were deemed to not be valuable enough. The analysis of the IP addresses used by the attackers revealed the systematic efforts made

in order to avoid detection, with each IP address accessing only 9.6 accounts on average over a period of two weeks.

Weaver and Collins[63] applied capture-recapture analysis to phishing feeds to estimate the size of the phishing population. By computing the overlap between the two phishing feeds and applying capture-recapture analysis, the authors were able to estimate the number of unique phishing campaigns and identify /24 netblocks that hosted a high concentration of phishing websites.

3.4.6 Summary

Overall, the works in this section discuss how phishers use search engines to locate and exploit vulnerable machines, forward stolen credentials to email dropboxes, and make money from the stolen credentials. Phishers carefully craft search engine queries that allow them to locate vulnerable hosts without having to scan the IP space, and these vulnerable hosts are shown to be repeatedly exploited until they are patched. Phishers who use phishing kits often have to specify a email dropbox to receive the credentials, and the creator of these kits often put a backdoor into the program that allows them to receive the stolen credentials as well. Phishers then sell the stolen credentials to cashers, who use the information to withdraw money from the victim's bank account. Some results also indicate that, due to the open economic nature of phishing, the phisher are earning no more than what they would earn from legitimate occupation. Other works also show how to quickly create a phishing website, analyze manual account hijacking attacks, and estimate the unique number of phishing campaigns.

3.5 Victim Profiling

In the previous sections, we have established several knobs that phishers could exploit in order to increase the effectiveness of their attacks. In this section, we explore characteristics of victims that could potentially be targeted by phishers in order to increase the effectiveness of their campaigns. Specifically, we want to examine if there exists a group of users with specific characteristics that the attackers can target in order to increase the yield from their attack. These characteristics include demographic factors,

behavioral factors, and personality factors.

3.5.1 Demographic Factors

Demographic factors such as age, gender, and level of education were discussed in many works as factors that may influence phishing susceptibility. While many of the studies did not report any significant correlation between demographic features and susceptibility, the works that are discussed here were able to identify correlation between the user's age, gender, and phishing susceptibility.

Sheng et al.[64] evaluated demographic factors in phishing susceptibility and effectiveness of interventions. The study was hosted on Amazon's Mechanical Turk platform, where participants answered survey questions about their background and their knowledge about phishing, completed a role-play task, received one or more form of training, then completed a second role-play task. They found that some demographic factors such as gender and age played a role in phishing susceptibility. Namely, women were more susceptible than men and participants in the 18-25 age range were more likely to become a victim than those of other age ranges. The authors suggested that since younger people have a lower level of education, fewer years of experience with the Internet, less exposure to training material, and less of an aversion to financial risks, they tended to be more susceptible to phishing.

Blythe et al.[13] performed four studies that investigated why people fall for phishing. In the second experiment, an online survey was conducted where the participants were asked to rate emails as phishing or legitimate on a four points scale. The result indicated a significant difference in detection accuracy between men and women, with men being more accurate, and that this difference was most prevalent among the younger age group. In the third experiment, a small qualitative study was undertaken to investigate whether blind users were more vulnerable to phishing. Based on the use of screen reading technologies to read email, the authors found that blind users could often identify phish in the first few lines of the email.

Holm and Ericsson[40] performed two naturalistic phishing experiments at a Swedish technology consulting company in the electrical power domain. Demographic-wise, the only significant result found by the authors was that

older individuals tends to be less aware of security than younger individual. This finding contradicted some of the earlier results, and the authors speculated that perhaps the subject's experience in IT was more of a contributing factor than age.

While the results from these works showed that age and gender played a role in phishing susceptibility, namely that women and younger users appeared to be significantly more susceptible to phishing, another factor behind these results may simply be the user's experience with computer and the Internet. Younger people have a lower level of exposure to phishing education and fewer years of experience with the internet, thus may be more susceptible to phishing. The same reasoning can also be applied to the observation that older individuals tend to be less security aware than younger individuals: they may not have much experience with phishing and IT systems in general.

3.5.2 Behavioral Factors

Several research works have been done to construct a behaviorally model that accounted for the user's cognitive processing of phishing emails. These studies attempted to identify the behavioral factors that caused the users to be more susceptible to phishing and offered suggestions to both users and system designers about changes of the users' behaviors that may reduce phishing susceptibility.

Vishwanath et al.[65] proposed SCAM, a model that accounts for the cognitive, preconscious, and automatic processes that may potentially lead to phishing-based deception. The model was tested using two experiments where phishing emails were sent to undergraduate students at a university. The email contained an embedded link in the first experiment, and it contained an attachment in the second experiment. The experiment's result revealed that heuristic processing of phishing email decreased the individual's suspicion of the email, and individuals who believed that their actions on cyber space were relatively safe tended to heuristically process emails. Individual's habitual patterns and beliefs about cyber risk also directly influenced suspicion, which in turn influenced susceptibility.

Vishwanath et al.[66] created an integrated information processing model

of phishing susceptibility and validated it with a survey on a sample of intended victims of a phishing attack. The survey result suggested that individuals get phished for two main reasons: because they do not adequately process the information, and because of their media usage habits. Domain specific knowledge had a limited effect when the users did not adequately processed the information, as the application of these knowledge required attention and elaboration. Users who received and responded to more emails are more likely to be deceived because their established email habits caused them to engage in little cognitive deliberation. This will make them more likely to ignore nuances in the email that may have revealed the deception. Hence, the authors suggested that creating safer email rituals may be a more sustainable solution in the long term. The authors further investigated how users processed a phishing email and examined how user's attention to visual triggers and deception indicators influenced their decision[67]. The results indicated that attention to visceral triggers, attention to phishing deception indicators, and phishing knowledge played a critical role in phishing detection. Namely, attention to visceral triggers (i.e. urgency cues) increased the likelihood of responding to a phishing email, while attention to deception indicators and phishing education decreased the chances of getting deceived.

Wright et al.[68, 69] identified behavioral components of successful deception detection in phishing context. The authors conducted an experiment in which each participant received a phishing email emanated from the authors asking for a sensitive information. The participants who expressed disbelief in the phishing email were asked to participate in a follow-up interview. From the interview, the authors found that while the participants do acknowledged the source of the message, the subject line of the email added to its authenticity and enticed the participant to open the email. Once opened, examining the layout and language as well as the nature of the content aroused suspicion. Furthermore, the participants' responses indicated that individual factors and contextual priming encouraged the said suspicion. These results implied that users should not ignore their feelings of apprehension while choosing whether or not to comply with a phishing request, and that users who were suspicious either through personality-based traits or knowledge-based awareness tended to be successful at detecting deceptions.

Watters[70] modeled the trust behaviors of users based on habituation and sensitization. Habituation model described the building of trust between the

user and another entity, with the user's level of distrust decreasing over time as they interacted with the other entity. Once users were habituated to trust the email, they responded to the message without an appropriate level of cognitive processing. To encounter habituation, sensitization model provided one mechanism to rapidly increase distrust level in the user. When an unanticipated stimulus such as a spoofed phishing email from a trusted entity was presented to the user, their level of distrust in the entity would sharply increase. However, more positive interactions between the user and the institution gradually habituated the user again. In summary, users who were habituated did not adequately process phishing cues in emails at sufficient depths to detect the deception. Defensive mechanisms could utilized the sensitization model to counter habituation by attempting to intervene the user's habituated response.

Dong et al.[71] introduced a model to visualize the interaction between users and phishing attacks from the victim's point of view. The model stated that users made two types of decisions in user-phishing interaction. The user first decided a series of actions to take, then decided whether to take the next planned action. Both type of interactions could be further divide into 3 stages: construction of the perception of the situation, generation of possible actions, and generation of assessment criteria and choosing an action. In the case of phishing, the phisher attempts to engineer two false perceptions: the perceived participant and the perceived consequences. For the generation of possible actions, the phisher conveniently provided the victim with a "solution", which was the action that was premeditated by the phisher. Hence, the authors suggested that the construction of an accurate perception was the key to detect phishing attacks. This implied that system designers should focused on providing security tools or indicators that will get the user's attention before they constructed a false perception of the situation.

Overall, heuristic processing of phishing emails was likely to decrease the individual's suspicion, as their domain specific knowledge had a limited effect in this case because the application of these knowledge required attention and elaboration. Individuals who believed that their action on cyber space were relatively safe, and those who established habitual patterns were also less likely to be suspicious of phishing emails. Hence, the results suggest that the construction of an accurate perception of emails in the user is the key

to detect phishing attack, which can be start by establishing a safer email habit.

3.5.3 Personality Factors

The research works in this category aimed to identify correlation between the users' personality and their susceptibility to phishing. This was done through the use of the Big-Five personality framework[72] and the Cognitive Reflection Test[73]. The Big-Five framework is one of the most widely used models for personality, and it consisted of five broad bipolar factors that represent personality at the highest level of abstraction. The factors were Openness, Conscientiousness, Extraversion, Agreeableness, and Neuroticism. The Cognitive Reflection Test uses three questions to measure the test taker's tendency to override their initial response and engage in further reflection to find a correct answer.

Parrish et al.[74] proposed a conceptual framework that utilized the Big-Five personality traits as a way to determine user's susceptibility to phishing attacks. They proposed that the Openness, Extraversion, and Agreeableness attributes directly associated with phishing susceptibility. People with these traits are open to all experiences, like to surround themselves with other people, and are trusting. Conscientiousness and Neuroticism, on the other hand, were speculated to be negatively correlated with phishing susceptibility, as these traits were correlated with being able to follow guidelines and reluctance to share information.

Alseadoon et al.[75] performed a naturalistic phishing experiment to determine factors that affected user's vulnerability to phishing emails. The participants first took a survey about their individual factors, including the big five personality test, then fake phishing emails were sent to the them. The email requested the recipient to provide his private information either by replying to the email or by clicking on a link in the email. The result showed that a total of 7% of the participants fall for the phishing email, with 87% of the victims clicking on the link and the rest responding to the email. The result also indicated that submissive users and users with low experience with email were more likely to not suspect phishing emails, and that users with high extroversion and openness were more likely to respond to phishing

requests.

Pattinson et al.[76] investigated the behavior response of computer users when presented with legitimate emails and phishing emails. Only half of the participants were informed that the purpose of the experiment was to test their ability to identify phishing emails. The participants were also asked to complete a demographic questionnaires, cognitive reflection test, and a personality test. The results indicated that participants who were more familiar with computers were better at approaching phishing email. The more extroverted, more open, and less impulsive participants performed better among the participants who were not informed about the true purpose of the experiment. The participants who were informed about the purpose of the study also performed significantly better than the uninformed group, suggesting that constant reminder of phishing email threat could be helpful to the users.

As part of their experiment, Kumaraguru et al.[77] also asked the participants to take the Cognitive Reflection Test, and found that participants with higher score (correctly answered more questions) were more likely to click on the links in the phishing emails from companies that they did not have an account with.

These results indicated that, personality wise, it was inconclusive to determine if users were more susceptible to phishing based on the Big-Five personality model and the Cognitive Reflection test. Specifically, high extroversion and openness were thought to be correlated with higher likelihood of responding to phishing requests, but experimental results were inconclusive. Similarly, users who scored higher on the Cognitive Reflection Test were thought to be better at recognizing phishing emails because the high score is correlated to being less impulsive, but the results from these studies were conflicting.

3.5.4 Summary

In general, the only factor that was shown to be correlated to the user's susceptibility to phishing was habituation. When users are habituated to their email rituals, they heuristically processed the information presented in the email and thus become less likely to notice phishing cues in the emails.

Demographic wise, women and younger individuals in the 18-25 years age range were reported to be more susceptible to phishing, but this result maybe skewed by factors such as their experience with phishing education and Internet expertise. However, behavioral research suggested that for users to be able to utilized their domain specific knowledge, they need to be paying attention to the task at hand, which was usually not the case once users are habituated to their email routines. Personality wise, there is no consensus in term of the correlation between the results from either the Big-Five personality test or the Cognitive Reflection Test and the test taker's susceptibility to phishing.

3.6 Chapter Summary

In this chapter, we have discussed research that studies characteristics of phishing attacks. Researchers have identified many knobs that the phishers could use to adjust the effectiveness of their phishing campaigns. Spear phishing attacks are found to be generally more effective in terms of the number of victims, but they are also more likely to be recognized and stopped in a shorter amount of time. Multiple alternative attack vectors were proposed, with an emphasis on attacks specific to mobile platforms and DNS poisoning attacks. Aspects of phishing beyond the actual attack are also discussed, including how the phishers locate and exploit vulnerable hosts, retrieve stolen credentials from the phishing websites, and profit from the stolen information. Other works also explore an easy way to create a phishing website, analyze an attacker's behaviors in manual phishing attacks, and quantitatively estimate the size of the phishing community. Behavioral analysis of the users revealed that the only factor found to be correlated to phishing susceptibility is habituation.

CHAPTER 4

MITIGATION TECHNIQUES

In this chapter, we explore the current mitigation techniques proposed to combat phishing. This includes the discussion of detection techniques, defense techniques, and their evaluation. Detection techniques identify phishing URLs, emails, and websites automatically based on the content of the media without the need for actions from the end user. Defense techniques does not automatically identify websites as phishing, but they assist the user in identifying potential phishing websites or propose new authentication protocols designed specifically to counter phishing. Several anti-phishing tools are then evaluated for their coverage and accuracy, and whether they can effectively protect the users.

4.1 Detection Techniques

In this section, we explore mitigation techniques proposed to detect phishing websites. The techniques outlined in this category relied on manual or automatic analysis of the contents obtained through URLs, emails, and websites without requiring users intervention. These techniques offered trade offs between coverage and accuracy, and thus occasionally designed to complement each other.

4.1.1 Blacklist Approach

Blacklist is the most primitive form of defense against phishing attacks. In earlier days, organizations such as PhishTank maintained a list of URLs that were manually verified by humans to be hosting a phishing website. Various security tools would utilized the list to warn users if they are viewing a URL with prospectively malicious contents. Since the verification process

was carried out manually, it incurred a substantial delay between the time in which the URL was first discovered and when it is added to the blacklist. Reducing this delay is a crucial task, as the users who will eventually click on a link in a phishing email often do so within the first hours of the attack. In an attempt to reduce the latency, several mechanisms were proposed to speed up the verification process.

Wardman and Warner[78] proposed a method for automating the verification process through usage of previous phishing websites. Their observation was that components of phishing websites, such as images, css files, and JavaScript files were more often than not reused, so a similarity matching of the MD5 hash of the resources against those of known phishing websites can be use to determine whether a suspicious website is phish without requiring human inspection. The authors evaluated this method with a dataset of 236 phishing URLs, and revealed that the method can identify 30% of the phishing URLs based on the matching of the MD5 value of the main HTML page, and MD5 set matching has the potential to confirm 20% more URLs.

Prakash et al.[79] created PhishNet, a system that appropriated heuristics to enumerate combinations of known phishing URLs to discover new ones. PhishNet has two separate components: an offline URL prediction component and an approximate URL matching component. In the URL prediction component, five heuristics were used to systematically generate new URLs and verified whether the new URLs contained web pages that carries matching contents to those web pages used to generate them. In the approximate matching component, similar heuristics were applied to determine whether a given URL was a phishing site when the URL did not exactly match any URLs in the blacklists. This was done by first breaking the input URL IP address, hostname, directory structure and brand name and matched them against corresponding fragments from URLs in the blacklist to generate one final score. If the score landed above a pre-defined threshold, the URL will then be flagged as phishing. The authors evaluated PhishNet against 6000 URLs from a live feed by PhishTank, and found 18,000 new phishing URLs from 1.5 million URLs generated by the URL prediction component. The approximate matching component was evaluated with phishing URLs from PhishTank and SpamScatter as well as legitimate URLs from DMOZ and Yahoo Random URL generator. The system was able to classify the URLs with less than 3% false negative rate and 5% false positive rate.

Liu et al.[80] proposed Aquarium, a technique used to assist blacklist verification through clustering of similar phishing web pages and crowdsourcing. Aquarium was developed on top of Amazon's Mechanical Turk system (AMT) to collect human-verified label on phishing websites. To increase the throughput of the system, the authors proposed the use of clustering algorithms to create clusters of similar phishing web pages that can be verified as a set instead of individually. This was done through usage of shingling algorithm for similarity measure and DBSCAN for clustering. Through a 2-weeks evaluation of the system with 267 users, casting a total of 33,781 votes, the results showed a true positive rate of 95.4% and a 0% false positive rate. The best median time to label was 0.7 hours. A vote weighting system, which assigned weight to each participant's vote based on their past performance, was also introduced and was shown to be able to reduce the median time to label a URL down to 0.5 hours.

Several techniques were proposed to speed up the verification step of phishing blacklists. Leveraging automation, a method used to discover new phishing URLs based on known URLs in the blacklist, and a method for automatic verification using MD5 matching were proposed. To speed up manual verification, a crowd- source verification system designed on top of Amazon's Mechanical Turk was considered, and a web page clustering approach to reduce verification latency was discussed.

4.1.2 Whitelist Approach

Contrary to blacklist, the whitelist approach aimed to create a list of websites that were verified to belong to legitimate organizations. Creating a global whitelist for all legitimate websites was impractical due to the massive number of websites on the Internet. Hence, the approaches in this section focus on the generation and maintenance of user-specific personal whitelists.

Li and Helenius[81] performed a heuristic evaluation of five anti-phishing toolbars: Google toolbar, Netcraft, SpoofGuard, built-in phishing prevention in Internet Explorer 7 and IEPlug, which was a tool developed by the authors that maintained a whitelist of websites to be protected and alerted the user when they are on a website that may be impersonating those protected website. One general result reported by the authors is that a combination of

blacklist and whitelist approach granted more protection. Since the blacklist approach was prone to false negatives, and the whitelist approach proposed by the user was prone to false positive, the authors proposed that a scheme that utilized both approaches could grant more protection. The authors also noted that a client side tool should not relied solely on redirecting a user away from the phishing website to a safe website because internet connectivity issues may caused the user to remained on the malicious website.

Cao et al.[82] proposed AIWL, an anti-phishing tool that automatically maintained a personalized whitelist. The authors introduced a concept of Login User Interface, which was a 5 tuples consisting of URL, list of resolved IP addresses, DOM of the input widgets, hash of the website's certificate, and hash of HTTP form's source code. When a user successfully login to a website multiple times, the website was added to the whitelist. If the user submitted credentials to a website that was not part of the whitelist, the submitted information is intercepted and a warning is displayed.

Wang et al.[83] proposed APWL, an anti-phishing approach that utilized a user-generated whitelist. APWL used pattern matching to detect sensitive data such as a social security number and blocked the submission of the data to websites that are not on the whitelist. Furthermore, APWL did not allow the user to bypass APWL blocking except by manually adding the URL to the whitelist. Performance analysis showed that the tool only incurred a small performance overhead.

These whitelist approaches mainly relies on the users to specify the websites that they trusted. Potentially unsafe actions on websites that are not part of the whitelist are blocked. One possible problem with this scheme was the overhead incurred to the user to initialize the whitelist. While it has been proposed that the list of websites that a user frequents is relatively small, a warning would still be displayed every time a user try to submit sensitive information on a new legitimate website. This could potentially teach an insecure behavior to the user. If the user sees a warning every time he tries to submit information to a new website, he would soon ignore the warning and simply perform the actions that will make the warning go away.

4.1.3 Heuristic Approach

Heuristic approaches determined the legitimacy of emails or web pages based on the analysis of their content. They are originally designed to complement blacklists, since blacklist-based approaches tend to be ineffective for fresh phishing campaigns that have not been manually verified. From a survey of related literature, heuristic approaches can be split into four main categories: rule-based approach, search-engine assisted approach, content similarity approach, and visual similarity approach.

4.1.3.1 Rule-based Classification

Rule-based classification techniques are the simplest form of heuristic approach to phishing defense. They used a set of static rules to determine if an email or a web page is part of a phishing attempt. The rules are often constructed based on observation and analysis of the phishing media, and they are manually crafted and weighted for use in the detection task.

Chen and Guo[84] proposed LinkGuard, an end-host based anti- phishing system that detect phishing links in email. The heuristics used is based on the information present in the <a> tag of the email, specifically the anchor text and the href attribute. Some rules that were proposed include mismatched domain name in anchor text and in the href attribute, IP address in the href attribute, and URLs that use ASCII code encoding or special symbols. One notable feature of LinkGuard was the use of heuristics to detect cross site scripting in the target website. The URLs that were flagged as potential phishing URLs were check against blacklists, or a similarity check was performed against a list of likely targets in order to draw conclusions. LinkGuard was evaluated against 203 phishing websites, and was able to detect 195 of them.

Yu et al.[85] proposed PhishCatch, a heuristic-based phishing email detection tool. PhishCatch relied on a weighted average of the output from a set of rules to classify phishing emails, and the rules were based on the characteristics of links in the email, the message content, and the use of PhishTank's blacklist. Evaluation of PhishCatch on a dataset of 4804 emails, 3710 of which were phishing emails, showed that PhishCatch could detect 80% of the phishing email with around 1% false positive rate. The authors also

showed that PhishCatch was able to detect many Phishing URLs that were not part of PhishTank's blacklist.

Cook et al.[86] proposed Phishwish, a stateless phishing filter. Phishwish used a score computed from a weighted average of binary results from each of its rules to determine the legitimacy of a website. A website was determined to be phishing if its score is above a certain threshold. Evaluation of the system with 117 unique emails, 81 of which are phishing, demonstrated that the score range for the legitimate emails and phishing emails were distinct enough that Phishwish could classify phishing emails with high accuracy even with all the rules having equal weight. However, Phishwish's rules relied on the email to be in HTML format, and emails containing a large number of advertisements and links may cause false positives.

Chou et al.[87] proposed SpoofGuard, a client-side anti- phishing tool that warned users when requests for data maybe part of a spoof attack. SpoofGuard was implemented as a browser extension that used the suspicious website's domain name, URL, links, and images as well as the browser's history and the request's HTTP referer to evaluate the likelihood that a suspicious website is a spoofed website. SpoofGuard also intercepted and examined form data submissions and alerted the user if they were submitting a credential associated with one domain name to another domain name. Spoofguard was evaluated against fourteen phishing web pages that spoofed eBay, and the results showed that Spoofguard correctly identified all spoofed web page and also intercepted and displayed warning when sample eBay credentials are entered.

Alkhozae and Batarfi[88] proposed a phishing detection scheme based on analysis of the web page's source code, including characteristics of URLs in the page, loading of resources from other domains, email functionality, and the use of popup windows. The detection heuristic weighted the results from each rule, and outputted a number describing the probability that a web page was a phishing web page. No evaluation of the system against real phishing emails was presented.

Atighetchi and Pal[89] proposed PhishBouncer, an attributed-based phishing defense system. PhishBouncer was implemented as a proxy that monitors every HTTP requests and responses and used an ensemble of nine rules to determine if a website was a phishing site. Furthermore, PhishBouncer also had a probe module that could embed proactive detection behaviors into the

proxy.

Overall, these tools detected phish based on rules that were constructed from the observed characteristics of phishing emails or web pages. Various characteristics of the domain names and URLs were mentioned in almost all of the tools, suggesting that phishing URLs were observed to be deterministically different from legitimate URL. However, this also means that, from the phisher's perspective, construction of a phishing URL that imitates most aspects of the legitimate URL defined by these tools would allow him to circumvent most of the tools in this category. Many classification rules in this section were found to also be useful in detecting phish with the machine learning classification tools.

4.1.3.2 Search Engine Assisted Detection

One observation that was often made in regard to phishing websites was that they were visually similar to their target. This is a practical observation, as the goal of a phishing web page is to trick the victim into believing that they are on the website of the phisher's target. Leveraging this observation, the detection mechanisms in this category extracted keywords from a suspicious website, and used them as query to a powerful search engine to determine the websites legitimacy.

Zhang et al.[90] created CANTINA, an anti-phishing toolbar that used the content of the page and a trusted search engine to detect phishing website. CANTINA used the TF-IDF algorithm to identify the most important terms in a web page, and these terms were used to generate a lexical signature of that page. TF-IDF ranked terms based on their frequency in the given document and general importance of the term over all documents. The lexical signature was use as a query to a search engine, and if the domain name of the current web page matched the domain name of any of the the N top search results, then the web page was considered legitimate. CANTINA was implemented as an Internet Explorer extension and evaluated against 100 phishing URLs and 100 legitimate URLs. The result showed that CANTINA can catch about 97% of phishing websites with 6% false positive using TF-IDF alone, and caught about 90% of phishing website with 1% false positive when other heuristics are added.

Xiang et al.[91] proposed a hybrid phishing website detection method based

on information extraction and information retrieval techniques. Similar to CANTINA[90], the method utilized keyword-retrieval using TF-IDF scoring as query in a search engine. A web page was regarded as phishing if its domain name did not appeared in the top N results. A new component added in this method was the use of keyword-retrieval to identify the domain of the page's declared identity and compared it with the page's domain by utilizing search engine to determine legitimacy. Additionally, the author also implemented a login form detection module that classify a web page that contained no HTTP form as not phishing. The system was evaluated against a dataset of 11449 websites, and the result showed that the method achieve a true positive rate of 90.06% with a false positive rate of 1.95%.

Dunlop et al.[92] proposed GoldPhish, a browser plug-in that detected phishing websites by leveraging optical character recognition (OCR) technology and a search engine. Specifically, GoldPhish used OCR to read texts from the screenshot of a web page and submit them to a search engine. If the domain of the current web page was part of the top results, then the web page is determined to be legitimate. The system was evaluated against 100 phishing websites and 100 legitimate websites and achieved an average 98% accuracy rate, 0% false positive rate, and 2% false negative rate. The main limitations of GoldPhish were page rendering delays from the OCR and search engine lookups, vulnerability to PageRank manipulation attacks, and limited performance on web pages with insufficient OCR extractable texts.

Sharifi and Siadati[93] proposed a blacklist generator that utilized keyword extraction and a search engine to detect phishing website. This approach relied on the extraction of keywords from the web page of the URLs presented in the email, and the comparison of the domain name of the URL with those from Google's top search results. The authors did not propose an automated keyword extraction scheme.

Overall, the approaches in this category consisted of a scheme to extract keywords from the suspicious website and a search engine. If the domain of the suspicious website matches that of the websites in the top search results, then the website is deemed to be legitimate. Due to its construction, it is possible for the phisher to game the system by influencing the keyword extraction process or the search engine results. Namely, the phisher can modify the TF-IDF of the web page by adding invisible texts or replacing some of the texts with images. The phisher could also influences the search

engine's results by gaming it's ranking system.

4.1.3.3 Content Similarity Detection

Content similarity approach, as the name implies, are phishing detection techniques that identify phishing websites based on analysis of the website's content. Similar to the search engine approach, content similarity approach leveraged the observation that phishing websites were constructed to be visually similar to their target and sometime even using resources from their target. The techniques proposed in this category mostly aim to identify phishing websites by comparing their contents to those of legitimate websites'.

Ardi and Heidermann[94] developed AuntieTuna, a client-side anti-phishing tool that utilized cryptographic hash of web page contents to detect phishing web site. Leveraging the observation many phishing websites load contents from its target, AuntieTuna identified phishing websites by first generating a list of trusted website specific to the user, then split the DOM Object of each web page into chunks separated by the $< p >$ and $< div >$ tags and stored the SHA256 hash of each chunk. When a user visited an untrusted website, AuntieTuna computed the cryptographic hash of the suspicious web page's contents and compared them to the content hashes of the trusted websites. If the number of matched contents is above a threshold, then the website was flagged as phishing and an active warning was displayed to the user. This approach allowed AuntieTuna to detect 50 out of 124 phishing websites that target PayPal with 0% false positive.

Afroz et al.[95] proposed PhishZoo, a phishing detection system that used profile of trusted websites to detect phishing. A profile of a website was defined to be a combination of different metrics that uniquely identify that site, including SSL certificates, URL, HTML files, and extracted features from logo. The profiles for trusted websites were generated and stored for comparison against suspicious websites. Evaluation of the tool with a dataset of 1000 phishing websites and 200 legitimate websites showed a true positive rate of 90.2% and a low false positive rate.

Wardman et al.[96] described a content-based phishing detection system based on the use of a cadre of matching algorithms modified for the task of phishing detection. The aim of the system was to overcome phishers' obfus-

cation techniques that eluded simplistic hash-based detection by applying, in addition to hash-based matching, two popular file matching and string alignment algorithms: diff and ssdeep. phishDiff was a technique developed by the authors based on the use of the diff command to compute the percentage of different lines in the main index file of a phishing website. ssdeep was an implementation of a context-triggered piece-wise hashing algorithm that employed rolling-hash technique to the fuzzy hash value for two files. A technique called Syntactical Fingerprinting is also used to compare structure components within files. The system is evaluated against 49,840 websites, 17,992 of which are phishing websites, and the result showed that phishDiff, ssdeep, and Syntactical Fingerprinting achieved more than 90% detection rate with a low false positive rate.

Liu et al.[97] proposed a phishing web page detection scheme that identified the target of a phishing web page by using DBSCAN clustering algorithm on a set containing the suspicious web page and all of its associate web pages. Associate web page was defined as web pages which can be reached by a forward link from the given web page, and web pages returned by a search engine when query with representative keywords on the given web page. If a cluster was found around the given web page, then the web page was flagged as a phishing web page. The scheme was implemented as a windows application and evaluated against 8745 phishing URLs from PhishTank that target 76 companies, and 1000 legitimate web pages obtained from Random Yahoo Link. The results showed an accuracy rate of 91.44% and a false positive rate of 3.4%.

Xiang et al.[98] proposed an anti-phishing system that built on the shingling technique, a near-duplication detection algorithm that measured similarity between two web pages by calculating the percentage of matching n-grams of the two web pages. The classification process of the system started by comparing the suspicious website against a small whitelist of legitimate web pages, then compared the suspicious website against a sliding window of blacklist feeds by using the shingling technique. If the suspicious website was flagged as phishing, a TF-IDF analysis similar to that used in CANTINA[90] was performed to verify the result. The system achieved a true positive rate of 67.5% with 0% false positive rate with TF-IDF filtering, and a slightly lower true positive rate of 65% with 0.03% false positive rate when a sliding window of two weeks worth of blacklist feed was used to speed up the

detection process.

Rosiello et al.[99] developed DOMAntiPhish, an anti- phishing tool that leveraged layout similarity between web pages to detect when the user inputted his credential into a fraudulent website. DOMAntiPhish extended AntiPhish, which was an anti-phishing system that alert the user when they entered their saved credential information into an unidentified website. DOMAntiPhish compared the DOM object of the suspicious website against that of the saved website, and only raise an alarm if the suspicious website is structurally similar to the saved website.

Wenyin et al.[100] proposed a framework for the discovery of phishing targets by utilizing semantic link network. The motivation for this work came from the fact that many phishing identification tool required explicit phishing targets for comparison-based detection, but none of the tools actually employed any phishing target discovery mechanism. Hence, the authors focused on the application of SLN theory to explore target of phishing websites by utilizing information from the forward links in the phishing web page and a powerful search engine. In summary, a suspicious web page was determined to be phishing if it established a strong association relation with other web pages rather than itself. This was due to the observation that legitimate web pages often contained a forward link loops with its neighbor web pages, resulting in legitimate web pages having a strongest association location with itself. The system was implemented with a web front end and was evaluated with a dataset of 1000 phishing URLs from PhishTank and 1000 legitimate URLs. The results showed a false negative rate of 16.6% and a false positive rate of 13.8%.

Wenyin et al.[101, 102, 103] proposed an approach to detect phishing web pages based on content similarity. The content similarity between two web pages was measured in three metrics: block level similarity, layout similarity, and overall style similarity. The detection process start with the decomposition of the suspicious web page into a set of salient blocks, which were blocks which contain elements that are visually and semantically consistent within the block but were distinguishable between blocks. The layout similarity was computed from the weighted ratio of the number of matched blocks to the total number of blocks. The overall style similarity was calculated based on the style features such as font family, text alignment, background colors, etc. If the similarity between a suspicious web page and a known legitimate

web page was above a certain threshold, it was flagged as a phishing page. The system was evaluated with a dataset of 8 phishing web pages targeting 6 legitimate websites, and found that the pairs of phishing website and its target website have significantly higher values than other pairs, indicating that the similar assessment metric is suitably defined.

Overall, the techniques discussed in this section all leveraged the content similarity between phishing web pages and their target. One problem with many of these approaches was the reliance on knowledge the website that was the target of the phisher, which could cause scalability issues when the system is deployed in the real world. Also, since the similarity between the websites are computed based on the content, it is possible for a phishing website to be crafted in such a way that its DOM object is significantly different from its target, but the rendered web page is visually similar.

4.1.3.4 Visual Similarity Detection

The techniques in this category identified phishing websites based on their visual similarity to the target website. Contrary to content similarity approach, visual similarity approach involved the visual comparison of the rendered web pages instead of the underlying DOM object.

Chen et al.[104] proposed a visual similarity based anti- phishing scheme based on discriminative keypoint features in web pages. The scheme used Contrast Context Histogram (CCH) descriptors to capture invariant information on the suspect page, then matched them with the descriptors of authentic web pages that were usually targeted by phishers. The authors designed a light weight version of CCH, called L-CCH, and used it for this task. L-CCH extracted key points using grey-level information and a corner detection method and used neighboring pixels' relative brightness to define a keypoint. The key points were then clustered together using k-mean to take the spatial distribution of the matching key points into account.

Hara et al.[105] proposed a phishing detection mechanism based on visual similarity that did not require information about the phisher's target. The approach leveraged the observation that many phishers often targeted the same website, and most phishing websites often tried to closely imitate their target website. This observation allowed for a detection mechanism based on visual similarity between a suspicious website and a known phishing web-

site. The design of the system included an image database that stored the screenshot of websites along with its URL and a label. The label determined if the website is legitimate, phishing, or unknown. A suspicious website was visually compared to the websites in the database using ImgSeek, an image search tool. If the similarity of the images exceeded a threshold value, then the domain name of the suspicious website and the image was compared. If the domain name for both websites were the same, then the label of the stored image is returned, else the suspicious website is classified as phishing. If no visually similar website existed in the database, then the website was classified as legitimate and was stored in the database with unknown label. The evaluation of the system showed an 82.6% accuracy with 18% false positive rate.

Fu et al.[106] proposed a phishing web page detection approach that leveraged visual similarity between a phishing website and its corresponding legitimate website. The approach worked by compressing the image of the full rendered web pages into a small square image (e.g. 100 * 100 pixels) And then a signature of the image was computed. Visual similarity between signatures was then computed using Earth Mover's Distance. The authors evaluated the system against a data set of 10,272 legitimate web pages and 9 phishing web pages that target some of the legitimate web pages. They found that the system can detect 8 out of 9 phishing web pages with a reasonably low rate of false alarm.

Mensah et al.[107] proposed an anti-phishing system based on visual analysis. Their approach involved the use of visual analysis of the suspicious web page (page a) compared to two other web pages on the same domain reachable by a link on suspicious web page (page b and c). The idea behind this approach was that the visual similarity between page b and c was used as the baseline reference on how dissimilar the web pages are expected to be, and the difference between the visual similarity between page a and c was used as a feature for a machine learning classifier. Together with other legacy features, the classifier was able to achieve an accuracy of over 90% with 10-fold cross validation on a dataset of 70 phishing web pages and 70 legitimate web pages.

Medvet et al.[108] proposed a technique to identify phishing websites by visually comparing its similarity to legitimate websites. The technique considered three features to determine page similarity: text pieces, embedded

images, and overall visual appearance. These features formed a signature of a web page, which was then compared to signatures of known legitimate websites using measures such as Levenshtein distance between strings, 1-norm distance between colors, and euclidean distance between positions. After the comparison, a final similarity score was given for a each pair of websites, and a website was flagged as phishing if the score was above a pre-defined threshold. The authors evaluated this technique against a dataset of 27 positive pairs (phishing web pages) and 140 negative pairs (other legitimate web pages). The result showed that the technique achieved a false negative rate of 7.4% and a false positive rate of 0%.

Chen et al.[109] proposed a phishing web page detection based on visual similarity. Unlike other visual similarity detection approaches that performed similarity detection on a set of features from the website, this approach was based on Gestalt viewpoint on human perception, which stated that images were interpreted in a holistic fashion rather than as a set of distinct features. Augmented with a concept of supersignals, which explained how human use holistic interpretation of visual input to drive rapid and frequent decision making, the authors demonstrated that visual supersignals can be encoded by using compression function into simple numerical values, and the "distance" between these values can be considered as the perceived similarity between two web pages. The evaluation of the proposed system on a dataset of 320 phishing web pages targeting 16 legitimate web pages revealed that the Normalized Compression Distance between a phishing page and its target brand was significantly different than the NCD between two different legitimate websites. A smaller scale experiment yielded a 100% accuracy rate.

Geng et al.[110] proposed a phishing detection mechanism based on favicon recognition. Favicon is a graphic image associated with a particular website, and phishing websites often forge the favicon of a legitimate website in order to trick the users. The proposed mechanism detected the favicon in a suspicious web page, then performed a visual similarity evaluation of the favicon against a set of favicons from legitimate websites. Evaluation of the system against a dataset of 3,642 phishing websites and 19585 legitimate websites showed a 99.6% true positive rate and 2.79% false positive rate.

The techniques discussed in this section all leveraged visual similarity between phishing web pages and their targets. Each technique proposed a

different comparison criteria to determine similarity of web pages. Similar to content similarity approach, one problem with some of these techniques is the reliance on the knowledge the websites targeted the phishers, which could cause scalability issues in real-world deployment. Also, depending on the comparison technique, the phisher can game the system and produce a web page that is visually similar to the target to the human observer but not distinguishable by the algorithms.

4.1.3.5 Summary

Overall, heuristic approaches relied on various elements of the phishing web page or email to determine its legitimacy. Rule-based approaches involved the derivation of rules based on characteristics that are unique to phish and manual tuning to apply a weight to each of the rules. This approach is further discussed in the machine learning classification approach, where the weight of each rule can be determined programmatically. Search-engine assisted approaches relied on keyword extraction techniques and a powerful search engine to identify if the suspicious website is legitimate. Content similarity approaches and Visual similarity approaches relied heavily on the observation that phishing websites were almost always visually similar to its target, and many of them required the knowledge of the target websites. The main different between the two is that a content similarity approach examines the similarity between the structure of DOM objects, while a visual similarity approach examines the similarity between the images of rendered web page.

4.1.4 Machine Learning Approach

The research in this category explored the use of machine learning classifiers to determine the legitimacy of suspicious contents. Similar to rule- based heuristic approaches, the works in this area aimed to complement blacklist based detection through content analysis. The features extracted from the content that could be useful for classification purposes are manually identified, and machine learning algorithms were used to determine the effectiveness of the features and assigned appropriate weight to them for classification purposes. Based on the use of different phishing contents for feature extraction, the works in this category can be further classified into those that

utilized URL features, email features, web page features, and other features for classification.

4.1.4.1 Phishing Email Features

The research works in this section explored how machine learning algorithms could be used to classify phishing emails. Mainly, the researchers focused on identifying useful features that can be extracted from emails for classification purposes, and identifying the classifiers that was most suitable for the task.

Chandrasekaran et al.[111] proposed a phishing email classification tool that used a Support Vector Machine classifier and a feature set derived from the email's structural characteristics. A set of 25 features relevant to language, composition, and writing characteristics of the email was proposed, and simulated annealing algorithm was used to remove features that are considered noise from the feature set. A Support Vector Machine classifier was trained and evaluated on a sample from a dataset of 200 legitimate emails and 200 phishing emails gathered over a period of six months. The classifier was able to achieved an accuracy of over 90%

Abu-Nimeh et al.[112] compared the performance of six machine learning classifiers for the task of predicting phishing emails. Each classifier was trained on a dataset of 1171 phishing emails and 1718 legitimate emails using 43 features extracted from an email's body and header. The authors performed 10-fold cross validation on each classifier, and reported the average error rates over all 10 folds. They found that when the penalty for misclassifying any emails are equal, Random Forests outperformed all other classifiers. However, Random Forests also had the worst false positive rate, so another round of the experiment was done where false positives are penalized nine times more than false negatives. Logistic Regression outperformed all other classifiers for the second experiment and Random Forests performed the worst.

Saberi et al[113] proposed a phishing email classification technique that used an ensemble of naive Bayes classifier, Poisson probabilistic theory, and k-nearest neighbor algorithm. The goal of the classifier ensemble was to distinguish phishing emails from scam and ham emails with feature vectors generated using a "bag of words" approach. Evaluation of the ensemble classifier was done on a dataset of 4500 spams, 1500 hams, and 529 phishing

emails using the top 2400 most frequent words from these emails as features. The result showed that the ensemble classifier performed better than any of the three single classifier alone, achieving a 5.6% false negative rate and 0.08% false positive rate.

Fette et al.[114] proposed PILFER, a machine learning approach to phishing email classification that can be deployed on the server side. PILFER used a Support Vector Machine classifier and a set of 10 features extracted from phishing emails. The authors evaluated PILFER using 10-fold cross validation against a dataset of 6950 legitimate emails and 860 phishing emails, and found that PILFER achieves an overall accuracy of 99% with a false positive rate of less than 1%. Further analysis also indicated that while PILFER can be deployed in a stand-alone configuration, deploying it in conjunction with with an existing spam filter such as SpamAssassin can improve the false positive rates.

Basnet et al.[115] evaluated different machine learning algorithms for the phishing email detection tasks. Using structure features of email defined in other literature as well as novel features, the author evaluated several machine learning algorithms with a data set of 3027 legitimate emails and 973 phishing emails. Half of the dataset is used to train the classifiers, while the other half is withheld for evaluation purposes. The results showed that although the performance of all machine learning methods used is comparable, biased Support Vector Machine and neural networks consistently achieved the best results.

Bergholz et al.[116] investigated two model-based features derived from emails and evaluated their ability to detect phishing emails. The first feature is the use of dynamic Markov chain classifier to classified the message as ham or spam. This was done by creating two Markov chain generation models, one for ham email and one for spam. The email was classified as belonging to the model that has a higher probability to have generated it. The second feature was the use of latent Class-Topic Model, a model that identified clusters of words that tend to appear together in emails. Evaluation of support vector machines classifier on these two features and a set of basic features yielded an improvement in accuracy and false positive rates over a previous work by Fette et al.[114]. The authors then improved upon this work by incorporating a large number of features, most notably hidden salting detection, image distortion detection, and logo detection[117].

The new features were evaluated with a sample of 3636 phishing emails and 16364 ham emails by using 10-fold cross validation, and the results showed that salting features alone could identify phishing emails with an F-measure of more than 90%. Overall, the best features from the system could achieve an F-measure of 99.46%. Furthermore, results from a real-life deployment of the system showed that the system could be tuned to perform well in a real life setting both as a phishing filter and a spam filter[118].

Ma et al.[119] proposed a method to determine the provenance of phishing emails by clustering them using orthographic features. Orthographic features mainly consisted of style characteristics that are used to convey the role of words and sentences, which for phishing email included features such as the number of images, links, and fake HTML tags in email. A modified global k-mean clustering algorithm was used to determine the clusters. Experimental results on 2048 emails showed that orthographic features played an important role in the clustering of phishing emails to determine its provenance, especially after dimensional reduction by repeatedly omitting subset of the features.

L'Huiller et al.[120] proposed a game theoretic data mining framework and used it to build an adversary aware classifier for phishing detection. The proposed framework utilized a Weighted Margin Support Vector Machine with a game theoretic knowledge function. The incorporation of the game theoretic approach allowed the classifier to change its parameters dynamically as the game evolved, with the awareness of the adversarial environment.

Ramanathan and Wechsler[121] proposed phishGILLNET, a multi-layered a server-side phishing detection system that employed Probabilistic Latent Semantic Analysis, AdaBoost, and Co-Training. At its core, phishGILLNET used PLSA, a technique for topic discovery that could handle both synonyms and polysemys by mapping a high dimensional vector of word distribution in a document to a lower dimensional vector of topic distribution. The mapped feature vector was used by an EM classifier on the first layer and AdaBoost on the second layer to perform classification. The third layer of phishGILLNET utilized Co-Training, a technique that is used to generate label for unlabeled data. Co-Training reduced the need for manual labeling of a large corpus of data. The authors reported that phishGILLNET3 achieves an F-measure of 100% on a very large dataset of 400K emails.

Olivo et al.[122] proposed a technique to find the minimum set of features

needed for a reliable performance by machine learning based phishing detection tools. The authors identified 11 features that are used in a majority of literature and evaluated the effectiveness of their combinations using a hill climbing approach. The authors observed a 56% reduction in processing time when a set of 4 features are used instead of 11 with an insignificant decrease in detection rates. Hence, depending on the application, the defender could choose a model that is suitable for their resource constraints without a significant loss in accuracy.

Dewan et al.[123] proposed a machine learning based detection model for spear phishing email that used social features extracted from LinkedIn. Since spear phishing emails were context aware phishing attack that leveraged information about the victim, the authors introduced nine features extracted from the LinkedIn profile of the email's recipient to be used in addition to the features extracted from the email. However, the authors found that the classifiers performed slightly worse with the feature set that includes social features, which may be due to the limited amount from information that can be gathered from LinkedIn.

Smadi et al.[124] proposed a phishing email detection model that used a J48 decision tree classifier on 23 features generated from the header and body of the email. Some of these features included the message ID domain, sender domain, message type, number of links and characteristics of URLs in links. The classifier was evaluated on a dataset of 4559 legitimate emails and 4559 phishing emails using 10-fold cross validation, achieving 98.11% accuracy and 0.53% false positive rate.

Verma and Rai[125] proposed a phishing email detection scheme that leveraged the Message-ID field in an email header. Message-ID are globally unique for every email, and usually constructed from the timestamp and the local host's domain name. The format of each Message-ID is ¡LHS@RHS¿, so the symbols ¡, ¿, and @ were removed as a pre-processing step, and each email is left with two attributes: LHS and RHS. n-gram analysis was performed on both LHS and RHS attribute and a new attribute was generated for each length n combination of characters. A dataset of 4,550 phishing emails and 9,706 legitimate emails was used to evaluate several classifiers, and the results indicated that RandomForest outperformed all other classifiers.

In general, although there is no consensus on the best features or classifier to use for the phishing email classification tasks, most of the works in this

section reported a very high accuracy rates with a reasonably low false positive rate. The features were commonly extracted from the email's header and message, but sometime external information such as social media profile or classification result from other classifier such as Markov chain classifier and latent topic classifier were also used as features. Each study proposed their own set of features and identified the best classifier through an evaluation process on a dataset consisted of samples from legitimate and phishing email corpuses. One study identified many common features that were proposed in other literature, then evaluated the effectiveness of subsets of these features and found that using a subset of four features to does not result in a significant different in detection accuracy.

4.1.4.2 Phishing URL Features

The research works in this section explored how machine learning algorithms can be used to classify phishing URLs. The main motivation behind the limitation of features to those extracted from the URL was to avoid the loading of web pages for analysis of the content on behalf of the user because the action may trigger a side effect on the website. The features used in these works are mainly the characteristics of the URLs as well as the information from other services such as WHOIS information and search engine's PageRank.

Garera et al.[126] studied the structure of phishing URLs and found that it was possible to tell if the URL belongs to a phishing attack by using features derived from the structure of the URLs and it's PageRank. The derived features included binary features of whether the URL contains certain words such as signin, login, and webscr, and binary features indicating whether the URL matched one of the four phishing URLs types derived by the authors. These URL types are URLs with IP address instead of hostname, URLs with target domain name in the path, URLs with target domain name in the subdomain, and URLs with unknown or misspelled domain name. By evaluating a classifier with a dataset of 1245 phishing URLs and 1263 benign URLs, the authors showed that the classifier with their feature sets had an average accuracy of 97.31% with a false positive rate of 1.2%.

Ma et al.[127] proposed an online learning approach for detecting malicious websites based on lexical and host-based features of URL. The authors argued that an online learning method was more suited than batch learning

systems because online methods could process large numbers of examples far more efficiently, and could adapt to changes in malicious URLs and their features over time. The proposed features include lexical features such as hostname, TLD, and primary domain, and host-based features such as WHOIS information, hosting location, connection speed, and blacklist membership. These features are used by a number of different online algorithms including perceptron, logistic regression, passive-agressive algorithm, and confidence-weighted algorithm. Each classifier was evaluated against a continuous, real-time feed of malicious URLs from a large web mail provider and a feed of legitimate URL from Yahoo's directory listing. The result indicated that confident-weighted algorithm can be highly accurate with accuracy up to 99%, and that the retraining of algorithms continuously with new features was crucial for adaptability with a constantly evolving stream of URLs and features.

Ma et al.[128] proposed machine learning based phishing URL classification approach that used features extracted from the lexical and host- based properties of the URL. Some of these features included the IP address and domain properties of the URL, length of various parts of the URL, number of dots in the URL, and a bag of words feature for each delimited token in the URL. Classifiers that were considered by the authors include Naive Bayes, Support Vector Machine, and Logistic Regression. The datasets used for evaluation include benign URLs from DMOZ and Yahoo Directory and malicious URLs from PhishTank and SpamScatter. The result showed that different data sets provide different feature distribution for the task of detecting phishing URLs, and that the classifiers performed poorly when the training and testing datasets came from different data providers. Hence, a representative training data was deemed to be important for a real world deployment of the system.

Bulakh et al.[129] described a phishing URL classification approach from the perspective of a spoofed brand. The authors analyzed 260 newly verified phishing websites from PhishTank and found that many phishing websites interacted with the website of the brand that they spoofed in a number of ways such as asset retrieval and credential validation. These interactions from phishing websites could be detected based on a feature set extracted from the URL in the HTTP referer field. Varieties of classifiers were evaluated with the feature set using 5-fold cross validation on a dataset of 7100 phishing

URLs and 7100 legitimate URLs. The authors found that Random Forest turned out to be the best classifier with an accuracy of 96.34% and a false positive rate of 3.39%. Another experiment on a real-world dataset resulted in a classification accuracy of 94.06% and a false positive rate of 6.08%.

Overall, while the classification accuracy for the works in this category may not be as high as the classifiers that used features from the email or the body of a web page, the reported results are promising. With the limited number of features, the phishers could circumvent detection by using a reputable free hosting services or use a URL redirection services to hide the real phishing URL. They could also carefully crafted the phishing URL to avoid being detected. One interesting aspect reflected in some of the works in this category was the exploration of the effect of using different data sets for training and testing. When a classifier was trained using data from one corpus and then evaluated with data from another corpus, the results indicated that the classifier performed significantly worse. Moreover, results from the exploration of online classifier suggested that a constant re-training of the classifier with the most up-to-date data is needed to ensure accuracy. These discoveries revealed the dynamic nature of phishing and implied the need for a detection solution that could adapt to these changes over time.

4.1.4.3 Phishing Web Page Features

The research in this section explored the effectiveness of machine learning algorithms for the task of phishing web page classification. Mainly, the researchers focused on identifying useful features that can be extracted from the content of the web pages and URLs for classification purposes, and choosing the classifier that is most suitable for the task.

Pan and Ding[130] developed a machine learning approach to detect phishing websites based on anomalies presented in the website's identity. The authors defined the identity of a website to be a set of words which uniquely identify the website's ownership in cyberspace. This identity was computed by an algorithm that used keyword extraction techniques from information retrieval literature. A set of features was generated based on the website's identity, and a classification of website was done by a Support Vector Machine classifier. The authors showed that the classifier was able to achieve an accuracy of more than 90% when trained on a dataset of 50 phishing web-

sites and 50 legitimate websites. However, the result also indicated that the performance of the classifier relied heavily on the performance of the identity extractor, as the false positive rate can get as high as 15% when the identity extractor is used but remain less than 5% when the web page identities were manually selected.

Miyamoto et al.[131] evaluated 9 machine learning algorithms using 8 heuristics features from CANTINA[90] to classify phishing web pages. The features used in the evaluation are the age of domain, known images, suspicious URL, suspicious links, IP address, number of dots in URL, Forms, and TF-IDF. The classifiers were evaluated with a dataset of 1500 phishing websites and 1500 legitimate websites using 10 runs of 4-fold cross validation. The results showed that AdaBoost outperformed all other classifiers, and seven out of the nine classifiers outperformed CANTINA.

Aburros et al.[132, 133] presented a model to classify e-banking phishing websites based on fuzzy logic and data mining algorithms. Fuzzy logic is a form of multi-value logic where the truth value may be any number in the range [0,1]. The authors applied this concept to phishing by evaluating e-banking phishing websites using rule-based classification based on these fuzzy characteristics. 27 characteristics were extracted from the suspicious websites and evaluated in a hierarchical rule- based classifier, with the output being the labeling of the suspicious website as one of very phishy, phishy, suspicious, legitimate, or very legitimate. The authors neither specified how they identified the 27 base features nor how the evaluation rules were generated. It was also unclear how the line was drawn between each of the output label.

Aburrous et al.[134] proposed an approach to detect e-banking phishing websites using data mining algorithms. After conducted a social engineering case study and a phishing case study, the authors identified 27 phishing features and indicators then clustered them into six criteria: URL, security & encryption, source code & JavaScript, Page Style & Contents, Web Address Bar, and Social Human Factors. Using these features, the authors evaluated six data mining algorithms with a data set of 412 phishing websites, 288 suspicious websites, and 306 legitimate e-banking websites using 10-fold cross validation. The result showed that MCAR outperformed all other algorithms. The result also revealed a significant relation between the "URL & Domain identity" criteria and the "Security & Encryption" criteria in the

classification task, and that "Page Style & Content" and "Social Human Factor" criteria were insignificant in the classification task.

Whittaker et al.[135] described the design and performance characteristics of a proprietary machine learning classifier developed to detect phishing websites and maintained Google's phishing blacklist. The system collected web pages from users' submissions and Gmail's spam filters and extracted a number of features based on the URL, hosting information, and rendered HTML content. Some features that were highlighted in the paper included Information about whether the URL contains an IP address or matches a high profile whitelist, length of the host component in the URL, Google's PageRank of the URL, term frequency-inverse document frequency of the website, the extent to which the web page link to other domains, and the occurrence of specific words in URL. These features were used by a proprietary logistic regression classifier to make the final determination. The goal of the system was to minimize the number of published false positive, and the authors demonstrated that the system was able to correctly classified more than 90% of new phishing web pages after training is finished with less than 0.1% false positives.

He et al.[136] proposed a phishing web page detection method that used a Support Vector Machine classifier. 12 features were extracted from the web pages, with 9 features adapted from Anomaly method[130], 2 features from PILFER[114], and one feature from CANTINA[90]. The classifier was evaluated with a dataset of 200 legitimate web pages and 325 phishing web pages. With 50 legitimate web pages and 50 phishing web pages reserved for testing, the result showed a 97.33% true positive rate and 1.45% false positive rate. This result implied that the performance of a phishing website detector can be improved by combining several different methods together.

Xiang et al.[137] proposed CANTINA+, an anti-phishing tool that improved upon CANTINA[90] by introducing eight novel features and two heuristic modules. CANTINA+ had three main components: hash- based duplication remover, login form detector, and machine learning classifier. The hash-based duplication remover's aim was to detect phishing websites that were generated by phishing toolkits, which would all have very similar content. The second component was a login form detector, which classified web pages without login form as legitimate. Lastly, the machine learning component classified the web pages with the use of 15 features, 8 of which

are novel features introduced by the authors. Some of these features are the presence of domain name or brand name in unusual parts of the URL, insecure HTML form, bad links, and the presence of the page's URL in search engine's top results. After evaluating 6 machine learning algorithms, the authors found that Bayesian Network classifier outperformed all other classifiers, and the final system could classify phishing websites with over 90% true positive rate and less than 0.4% false positive rate.

Zhang et al.[138] proposed a novel phishing web page detection approach that utilized textual and visual content of the web page to measure the similarity between protected web pages and a suspicious web page. The system consisted of three components: a text classifier, an image classifier, and an algorithm that fused the results from the classifiers into one decision. A Bayesian approach was used for the text classifier, and the image classifier utilized Earth Mover's Distance for determination of visual similarity. The results from the two classifiers were then fused together using an approach based on Bayesian theory. A large scale experiment was conducted to evaluate the performance of each part of the system as well as the overall system. The dataset consisted of over 65,000 web pages targeting eight legitimate companies, and the results indicated that the system has very high accuracy rate and low false negative rate.

Bottazzi et al.[139] proposed MP-Shield, a framework for phishing detection for mobile devices. MP-Shield was implemented as an Android application service on top of TCP/IP stack, essentially acting as a proxy that intercept outbound IP packets from both browsers and other applications. URLs were extracted from the intercepted packets and evaluate against blacklist APIs, and the web page content was fetched to be classify by a machine learning classifier. Evaluation of the system showed a 89% true positive rate with a 14% false positive rate.

Dong et al.[140] proposed a machine learning approach to detect phishing websites based on features extracted from the site's certificate. Some of these features included the issue date and expiration date, information about certificate issuer, cryptographic algorithms, and certificate version. These features were used by an ensemble of six machine learning algorithms, and the final decision for the website was determined based on Average Probability ensemble approach. The system was evaluated against a dataset of 95,490 instances from phishing websites and 113,156 instances from legitimate web-

sites, and the result showed that Average Probabilities ensemble achieved a precision of 93.6% and a recall of 94.2% for the phishing websites.

Nguyen and Nguyen[141] evaluated the effectiveness of five machine learning classifiers. The features that were used in the evaluation are the presence of IP address in the URL, the presence of symbols in the URL, length of the hostname, TF-IDF of the web page content, the presence of suspicious link, the presence of forms, site popularity, and the age of the domain name. A dataset of 4,420 legitimate websites from 5000best.com/website and 5,389 phishing emails from PhishTank were used for the evaluation. The result indicated that lexical features of the URL contributed the least to the accuracy, and Random Forest outperformed all other classifiers with a true positive rates of 98.8% and a false positive rates of 1.2%.

Overall, results from the evaluations of classifiers and features in these works showed that machine learning classifiers can very accurately classified phishing web pages with reasonably low false positive rate. Some features that were considered in most of these works were related to the URLs and domain names of the web pages such as the age of the domain name and various structures of the URL. Since these features were common across many works, we can assume that they have a substantial weight in the classifiers. This implies that one possible attack against these classification systems could be to exploit a domain name with high reputation to host the phishing web page. Another possible attack is to use phishing URLs that are not distinguishable from legitimate URLs by the classifiers. Another observation is that some of the proposed features relied on the phishing web pages to load resources from its target, meaning that phishing websites should hosted their resources locally to avoid detection.

4.1.4.4 Other Features

The works in this category proposed machine learning based phishing detection systems that used novel features other than those extracted from phishing emails, web pages, and URLs.

Moura et al.[142] created nDEWS: a new domains early warning system for TLDs that detect malicious domain registration using unsupervised machine learning algorithm. The motivation for the system came from the observation that domains that were registered for malicious reasons tend to have a spike

in the number of DNS queries over the first few days after their registration. Based on this observation, nDEWS used k-means clustering algorithm on a feature set extracted from DNS requests. The authors specifically chose an unsupervised learning algorithm instead of a supervised learning algorithm because they wanted the system to be able to detect a wide range of malicious domain registrations.

Aggrawal et al.[143] proposed PhishAri, a machine-learning based real-time phishing detection for Twitter. PhishAri utilized features extracted from a URL and those specific to Twitter such as the age of the account, the ratio between follower and followee, the presence of trending hashtags in the tweets, and the number of mentions. Evaluation against a dataset of 1473 phishing tweets and 1500 legitimate tweets showed that PhishAri was able to achieve 95.52% accuracy. PhishAri is implemented as a RESTful API with a front- end browser extension for Google Chrome.

These research offers alternative detection vector for phishing campaigns. Detection based on the domain registration system relied on the behavior that phishers usually put the registered domain to use within a short period of time, which could easily be circumvented if the detection time window is known to the phishers. The development of phishing detection system for Twitter suggests that phishers may be using social media as a new attack vector to deliver malicious URLs, and many cues specific to Twitter were identified that could entice the users to click on the links.

4.1.4.5 Summary

We have discussed machine learning based detection systems for phishing. These systems take advantage of features extracted from URLs, emails, web pages, and other relevant information such as search engine PageRank and WHOIS information. Most of the features used by the machine learning classifiers were identified from empirical analysis of captured phishing emails and websites, and the classifiers were shown to perform well when the training and testing samples are drawn from the same corpus. However, some works have shown that when a classifier was trained using data from one corpus and then evaluate with data from another corpus, the performance of the classifier was significantly worse. Furthermore, results from the works that explored the use of online classifiers suggested that constant re-training of the classifier

with the most up-to-date data was needed to ensure accuracy. Since most works evaluated their classifiers with cross validation techniques, this begs the question about the accuracy of the techniques that use batch training approach: how well does the system perform in real world deployment, and how often does the classifier needs to be re-trained?

4.1.5 Network Level Approach

Network detection techniques for phishing are based on the analysis of network traces from ISP standpoint. Most of the works focused on the detection of fast- flux phishing campaigns. Fast-flux is a term coined in the phishing community to describe a decentralized botnet with constantly changing DNS records, which is used by phishers as proxies to hide the real phishing website. This involved the use of domain names that resolved to a large number of constantly changing front proxies to hide the "mothership" that hosted the actual phishing website. Other works also explored efficient retrospective detection of phishing URLs from network traces.

Zhou et al.[144] proposed a multi-stage decentralized collaborative intrusion detection system to detect and take-down fast-flux phishing campaigns. Since the front proxies can be hosted across many ISPs, the authors proposed a collaborative intrusion detection systems where each participating system shared evidences of malicious front-end proxies from their subnetworks to coordinate a detection of the mothership. The CIDS was build on top of a structured peer-to-peer network for scalability and self-healing property and used a publish-subscribe paradigm to accommodate frequent membership changes. The system also used a multi-stage approach to allow each participant to identify compromised hosts within their subnetwork.

Zhou et al.[145] proposed two approaches to correlate evidence of fast-flux phishing domains: correlation of evidences from multiple DNS servers and correlation of evidences from multiple fast-flux domains. To determined if a domain was a fast-flux domain, we needed to observe that the number of unique IP addresses belonging to that domain was above a certain threshold. This involved multiple DNS query over time, so the authors proposed to speed up this process by querying multiple DNS servers at the same time. Another observation was that multiple fast-flux domains shared the same pool of IP

addresses, so another model of fast-flux domain detection was proposed to classify a suspicious domain as a fast-flux domain if it resolved to some IP addresses that appeared in other fast-flux domains. Evaluation of the system on real world data showed a substantial speedup in terms of the number of queries needed for detection can be achieved and that a decentralized architecture is more scalable than a fully centralized architecture.

McGrath et al.[146] proposed a machine learning based system to detect fast-flux and DNS flux phishing attack. Using features generated from DNS resolution of suspicious URLs over time and a Support Vector Machine as the classifier, the authors found that 77.6% of fluxing web servers were part of a double flux network, which implied that fast-flux phishing occurs not only at the web server level but also at multiple levels in the DNS hierarchy. This suggested that most phishers who uses fast-flux approach are well provisioned against take-down efforts.

Li et al.[147] proposed LARX: a large-scale retrospective data exploration framework that analyzed archived network data. To efficiently gather network traces, the system recorded only the first 15 kilobytes of each connection, which was enough to record most connections in its entirely and enough to record the most useful part of the longer connections. The archived data were anonymized and then split into small chunks by tcpdump for processing. TCP stream were reassembled using tcptrace, and URLs were extracted and compared with a blacklist to identify phishing URLs. LARX's performance was evaluated on two cloud computing platforms, AWS and Eucalyptus, and a physical server using 10GB of network trace data. The authors found that the number of instances needed to process the data in a timely manner on Eucalyptus is impractical, but it was cost-effective to deploy LARX on AWS. However, LARX failed to find any phishing URL from the 10GB data trace that was used for evaluation, which the authors attribute to the changing coverage of blacklists.

Due to the nature of the fast-flux phishing campaigns, their detection required multiple DNS queries over time. Several works proposed a speed up techniques for this detection process by utilizing distributed analysis of several DNS servers, which required collaboration across ISPs. Results from a DNS analysis of fast- flux domain also indicated that the domains that were part of a fast-flux network were fluxing at the subdomain level as well, which implied that fast- flux phishers were well provisioned to avoid detec-

tion. Since fast-flux campaigns were identified by works in other areas to be responsible for more than half of all phishing emails, an effective solution for the take down of fast-flux campaigns would be very beneficial. Other works also discussed how network trace analysis could be done efficiently on cloud computing platforms and that retrospective discovery of phishing URLs using blacklists may not be feasible due to a change in coverage.

4.1.6 Honeypot Approach

Maggi et al.[148] created a data collection system that captured information about phishing campaigns with a particular focus on the voice channel. The system consisted of four modules: a phone module that answers inbound phone calls and record resulting conversations, an email module that collects spam and phishing emails, an IM honeypot that record unsolicited IM messages, and a social network module that crawl social networks and monitor suspicious accounts identified by the other modules. Collectively, the data from the first three modules were processed using a simple natural language processing techniques to extract popular words and sentences. Phone numbers and URLs were also extracted, and the URLs were resolved with longshore.com API to retrieve its redirection chain for external analysis. While the system did not explicitly identify phishing URLs, it supplied a list of suspicious URLs that have a high chance to be phishing.

4.1.7 Summary

Overall, it is clear that different detection mechanisms have varying degree of timeliness and coverage. Blacklist and Whitelist approaches required human intervention to identify legitimacy of websites and offered 0% false positive rates and 0% false negative rates. However, they suffered from high latency between the time when the website is discovered and the time when a decision is made. Many techniques were proposed to speed up the verification latency of blacklists, including the discovery of similar phishing web pages and the clustering of suspicious web pages to be identified at once. For whitelist based approaches, various personal whitelist schemes were proposed to mediate the impracticality of a single global whitelist. To complement the

list-based approaches, heuristic and machine learning based approaches that do not require human intervention were proposed. These approaches offered fast detection of new phishing websites but suffered from a small number of misclassifications. While a small number of misclassifications may seem negligible, a misclassification of a legitimate website as phishing can severely damage the reputation of the website and should be avoided at all cost. Some forms of phishing attacks such as fast-flux cannot be efficiently detected locally, so collaborative network level detection techniques were proposed. Methods for automatic discovery of suspicious URLs were also discussed, which could improve coverage of list-based detection approaches.

4.2 Defense Techniques

This section explores mitigation techniques that have been proposed to defend the user against phishing websites. Specifically, the techniques in this category required actions from the users in order to be effective. These actions can range from the user simply acknowledging a warning displayed by the tool, to a complete change in authentication behaviors.

4.2.1 Offensive Defense Approach

The works in this category aim to detect phishing websites based on the site's response to a submission of a fake credential, and inconvenience the phisher at the same time. By submitting fake credentials that are indistinguishable from real credentials, the tool is actively polluting the phisher's database. Most approaches in this category leverage the observation that just as end users cannot tell legitimate and spoofed emails apart, phishing websites also cannot easily distinguish the real and fake credentials apart. These fake credentials can also be used as honeytokens to alert the target when the phisher tries to verify the credentials on the target's website.

Yue and Wang[149, 150] proposed BogusBiter, a defensive strategy that submits a large amount of fake credentials that are indistinguishable from real credentials to phishing websites, with the goal of pollute the phisher's dataset. This will force the attackers to verify the stolen information, in which case the bogus credentials may also be used as honeytokens to alert the

target institution when the phisher attempted to verify the stolen credentials on the target website. BogusBiter is deployed as a web browser extension, and utilized other phishing detection components of the web browser to decide whether to submit the bogus credentials or not. BogusBiter is evaluated against a dedicated testbed to verify the indiscernibility of the bogus credentials, and against 50 phishing websites and 20 legitimate websites. The results show that the submission of the bogus credentials is not discernable by the phishers at the TCP connection level, and that BogusBiter successfully submitted the bogus credentials to all 50 phishing websites.

Chandrasekaran et al.[151] proposed PHONEY, an anti- phishing approach that send fake information to suspicious websites to determine its genuineness. PHONEY is designed to sit between the mail transfer agent and mail user agent, processing user's emails and discard phishing emails before they reach the user's inbox. PHONEY probed the user's emails and if a form is present on the email or on the web page with a link in the URL, fake information matching the input type is submitted in the form of active honeytokens. The behavioral response of the website is recorded and analyzed by a rule based system to deduce if the response is consistent with those of legitimate websites. PHONEY was evaluated against 17 phishing emails and a number of legitimate emails, and detected all phishing emails with 0 false positives. The authors also boldly claimed that PHONEY was able to successfully detect all email based phishing attacks listed on the antiphishing.org archive at the time of writing.

Joshi et al.[152] proposed PhishGuard, an anti-phishing browser plug-in that identifies phishing websites by submitting random credentials before the actual credentials during the login process. PhishGuard captured the credentials that the user intended to submit to the website and first submit a number of credentials with the same username but a randomly generated password to the website. The expected response of the legitimate website to these random credentials is a 401 authentication error, in which case the real credential is then submitted. However, one weakness of this scheme is when the phishing website always responds with a 401 authentication errors, in which case PhishGuard will continue to submit more random credentials in an attempt to obfuscate the real credential. Since PhishGuard does not store the credentials of the user, it can raise a false alarm when the user mistyped his credentials.

Li et al.[153] presented an anti-phishing framework based on the use of spamtraps and honeytokens. The proposed frameworks utilized four different components: fake credentials to be used as phoneytokens, spamtraps for gathering of phishing emails, phoneybots that routinely submit phoneytokens to the websites, and a honeyed e-banking system that is modified to act as a honeypot when a honeytoken is used to login to the system. The key idea behind the system is for phoneybots to imitate a human user and periodically login to the honeyed e-banking system and perform money transfer transactions. This allows pharmers and malicious software to steal the phoneytokens using DNS poisoning or keyloggers. When the attackers login with the honeytokens and attempt to transfer money into their account, the system will detect the transaction and the unknown account will be flagged as a malicious account. The financial institution can then alert their users who recently transferred money, to the flagged accounts, or give the information to law enforcement agencies.

Knickerbocker et al.[154] proposed Humboldt, a distributed system that poisons phishing websites' database by submitting fake credentials en masse to the phishing websites that are indistinguishable from legitimate credentials. The distributed nature of Humboldt makes it harder for phishers to filter out fake credentials based on the origin of the submission. Humboldt also uses a submission pattern that is undetectable to the phisher based on the coordination of multiple Humboldt clients by a central server. The central server creates a profile of phishing website for each URL in a feed, identifies the type of data requested and submission method, and stores it in a database (HumboldtDB). The server also generates the fake information, then distributes them to the Homboldt clients for submission. The client attempts to submit the given credentials to a phishing website, and reports back to the server. To defend against Humboldt, the phisher could use phishing URLs with unique parameterization, introduce malicious clients into the pool, or attack the Humboldt server.

Shahriar and Zulkernine[155] proposed a PhishTester, a phishing website detection scheme based on a trustworthiness testing approach. Trustworthiness testing of a website is unlike a test where the website is tested against a known input for expected outputs. It instead revolves around checking whether the behavior of the website matches our knowledge of the behaviors of phishing or legitimate websites, in order to determine if the website is legit-

imate or phishing. Specifically, the authors model the behaviors of phishing and legitimate websites as a finite state machine, and develop heuristics based on state, submission response, and form-based features to test the websites. PhishTester is implemented in Java and has been evaluated against 33 phishing websites and 19 legitimate websites. The result shows that PhishTester can classify all websites correctly. Furthermore, PhishTester can also detect phishing websites that exploited cross site scripting vulnerabilities in trusted websites.

McRae and Vaughn[156] proposed an approach to track down the phisher's workstation by utilizing honeytokens and web bugs. Web bugs are HTML image tags that contact a web server with a unique parameter when the web browser tries to render the image. This approach was previously used by spammers to identify active emails addresses. In this work, the authors proposed the submission of web bugs as honeytoken into the form on a phishing web page, with the goal of extracting the phisher's IP address and access time if they view the stolen information in an HTML enabled environment such as a webmail client. To evaluate this framework, the authors submitted the web bugs to 11 phishing web pages and received two referrals. The referral contained the IP address of the phisher's workstation, the time at which the information was accessed, the browser's agent string, and the URL that requested the image. The authors further discussed the easy countermeasures that the phisher could employ such as the use of text viewer or disable the loading of images.

Many techniques are proposed in this section that actively submit fake credentials to suspicious websites, and some of them identify the phishing websites based on the response to the submission of fake credentials. While this approach will not stop victims from giving their credentials to the phishers, it will make it riskier for the phisher to operate and reduce the potential financial gain from the attacks. However, phishers can circumvent the response- based detection method by responding with the legitimate website's response to the wrong input by default, or utilize robot detection schemes such as CAPTCHA, or use JavaScript to grab the credentials before they are submitted. Also, there is a possibility that the user may be locked out of their account as a result of the tool making too many failed authentication attempts, so the legitimate websites may have to increase their maximum tolerance on failed authentication.

4.2.2 Credential Tracking Approach

The works in this category aims to prevent the users from submitting their information to a phishing website by keeping track of the users' information and alerting them when they try to submit the tracked information to a new website. Most of the defense in this category are implemented as a browser plugin or toolbar, and some of them require the users to change their Internet browsing habit.

Kirda and Kruegel[157] created AntiPhish, an anti-phishing browser extension that display a warning to the user when they attempt to submit a saved credential to an untrusted website. After AntiPhish is installed, the user is prompted to set a master password that will be used to encrypt the stored credentials using DES. After this initial setup, AntiPhish can capture and store form information and a mapping to the domains where the information is submitted. Whenever the user enters information that matched the previously captured information, AntiPhish checks the domain of the website and assumes the website is malicious if its domain is not among the stored domains. AntiPhish is configured to disable Javascript on the website when the user starts typing information into a form, then enable it after the form information is verified to circumvent malicious JavaScript from stealing the information. AntiPhish is implemented as a Firefox extension, with an implementation for Internet Explorer planned.

Wu et al.[158] proposed Web Wallet, an anti-phishing browser sidebar that deterred phishing attack by determining where the users intend to submit their sensitive information and alert the users when the current website does not match their intended website. The sidebar worked by disabling any login forms on a web page, forcing the user to use a specific keyboard shortcut to open a Web Wallet interface to submit their information. This interruption of the user's workflow caused the user's attention to be focused on Web Wallet, which will present a warning to the user if they are trying to submit sensitive information to an unfamiliar website. The authors evaluate the effectiveness of Web Wallet by conducting a role-play experiment with 21 subjects. The authors found that Web Wallet significantly lowered the spoof rate of normal attacks from 63% to 7%.

Yee and Sitaker[159] created Passpet, a password management and phishing protection system implemented as a browser extension. Passpet automat-

ically fills in unique passwords for every website based on the user's master secret and the website's label. Using password hashing, the users only have to memorize one master secret, which will be combined with a user-specified label to create unique passwords for the websites. Petnames allow the user to assign a local label to objects, then translate global identifiers for objects into the user's local petnames to make sure that they are on the intended website. Passpet also generates a random animal icon and name for each user, making it harder for attackers to spoof the Passpet interface. Passpet alerts the user when they have entered a label to fill in a password for a website that was not assigned by the label, and effectively breaks the user's habit of entering passwords into websites by making it convenient for them to use Passpet's UI to fill in the password.

Florencio and Herley[160] proposed a phishing detection system that tracks user's passwords re-use behavior at unfamiliar websites. The system consisted of a browser toolbar and a server. The browser toolbar tracks submissions of user's credential for whitelisted websites on unfamiliar websites and report it to the server. These reports contain the domain name of the whitelisted website and the suspicious website, which the server then aggregates across all users and identify pairs of phishing websites and its target according to pre-defined rules. The server then contacts the phishing target and alerts them of the users who were phished. While this scheme does not prevent the phisher from obtaining the victim's credentials, it can then mitigate the damage before the credentials are used.

Bin et al.[161] proposed a DNS based anti-phishing and anti-pharming system specifically designed to protect credit card numbers and login credentials for financial accounts. The system consists an information server, and a browser toolbar. The information server hosts a Bank Information Database, which stores the range of credit card numbers allocated to a specific bank along with the bank's name, IP addresses of the bank's DNS server, and IP addresses of other websites that are allowed to access the card number. The browser toolbar periodically queries the database and stores the data locally. The browser toolbar intercepts packets that contain a digit string that has the format of a credit card number, and performs a reverse DNS lookup on the destination IP address. If the destination IP address belongs to the bank or the trusted websites, then the information is allowed to be submitted, otherwise a warning is displayed to the user. If the user indicates to the warning

that he is submitting a credit card number, then the website is determined to be a phishing website, and the toolbar prevents the submission of the card number.

In general, the tools proposed by these research works track the submission of user's information and alert them when they try to submit the same information to a new website. Since most of these tools are implemented as a browser add-on, one attack that the phisher could launch against the tool is to spoof the tool interface as part of the web page. This is recognized by some tools such as Passpet, which counters this attack by not having a default behavior that can easily be spoofed. Some of these tools also require a change of behavior from the user, which would make their interaction more secure but may prove to be a barrier of entry for some users. Also, if the tool displays warnings on many legitimate websites, which may be the case if the users re-use their passwords, the users may become habituated to the warnings and simply ignore them.

4.2.3 Novel Authentication Protocol

The techniques in this category attempt to defend the user from phishing websites by proposing alternative authentication protocols. While some of the protocols are deployed on the server side and simply provide an integrity check to the user, many of these protocols have implementation on the client side, which may require the users to actively use the tool in order to be secure. The actions required by the user ranges from inputting their credentials in a specified area, to physically pushing the power button on their phone every time they want to sign in to a website.

Herzberg and Jbara[162, 163] created TrustBar, a secure user interface implemented as web browser add-on. TrustBar displays the identity of the website from its SSL certificate along with the name of the certificate authority who identified the site. When the website is not protected by SSL, a warning message is displayed instead. The name of the website and the name of the certificate authority can be substitute by brand icons or images selected by users to help them identify the websites that they frequent.

Dhamija and Tygar[164] proposed a new class of Human Interactive Proofs (HIPs) that allows a human to distinguish one computer from another. The

challenge in this model must be easy for a particular class of computers to pass, but hard for other computers even after a number of successful authentications were observed. This class of HIPs can be used by humans to distinguish a known legitimate website from an unknown one, which would be useful in the detection of phishing attacks. The authors then show that Dynamic Security Skin, a scheme proposed by the authors, satisfies all requirements for the proposed class of HIPs[165]. The client-side of the scheme was implemented as a browser extension to Mozilla Firefox that provides a trusted windows dedicated to username and password input to all websites. Once the user is authenticated, the browser generates a "skin" that is unique to each user and transaction, and displays it as some element on the web page. The browser extension also computed and displayed the same image independently, allowing the user to easily verify the content from the server visually.

Topraka et al.[166] proposed ViWiD: a server-side phishing defense mechanism that embeds a timestamp and a personalized secret word chosen by the user to the website's logo before serving it to user. The unique watermarking allows the user to verify the integrity of the legitimate website, and thwart the "one size fit all" phishing attacks that spoof legitimate websites. However, it is unclear if this scheme is secure against phishers who display resources directly from the target website.

Parno et al.[167] proposed an anti-phishing system that forces the user to enter their credentials through a trusted device. The system consisted of a browser extension and a mobile phone application. Once the registration process, which required a nonce to be sent securely to the user, is done, the user must authenticate via the established public key pair and username/password combination, which is only available when the user initiates the connection on the phone. While this approach may not prevent the users from disclosing their credentials to phishing websites, it will prevent the attack from directly utilizing those stolen credentials on the servers that implemented this system.

Gouda et al.[168] proposed SPP, an anti-phishing single password protocol that allowed the user to safely use one password on all websites. The protocol consisted of three messages: the user sends his username to the server, the server replies with a stored challenge specific to the user, then the user responds with a one-time server-specific ticket for that challenge and a new set of challenge and ticket verification information. The implementation of

the SPP revolved around the use of a random number as the challenge, and the result of a one-way hash function on the number, password, and server identifier to produce a ticket and its verifier.

Tout and Hafner[169] proposed Phishpin: an anti-phishing approach that utilizes mutual authentication to detect phishing websites masquerading as legitimate websites. Phishpin uses partial credentials sharing and client filtering techniques to enable both users and other entities to mutually prove their identity to each other without divulging sensitive information during the process. After certificate validation, the user authenticates oneself to the server by constructing a string that contains one half of the hash of their password, a shared secret value, and a challenge phrase. The hash of the constructed string and the challenge phrase is then sent to the server. On the server side, the plug-in validates the hash with the challenge phrase and the stored user's information, then authenticate itself to the user in a similar fashion using the hash of the other half of the user's password.

Oiwa et al.[170] proposed a password based mutual authentication protocol that can prevent various kind of phishing attacks. The protocol is based on the PAKE cryptographic scheme, which stands for Password-Authenticated Key Exchange. With this protocol, if the user is connected to a phishing site that doesn't know the user's password, then the mutual connection will not succeed, and the submitted information will not allow the phisher to access the user's account on the legitimate website. One possibility of an attack against the protocol is to forge the UI dialog that asks for the user's password, and the authors suggest the use of an extension to the browser that has dedicated password input field instead of dialog.

Crain et al.[171] proposed Trusted Email, an anti-phishing tool that utilized public key cryptography to sign emails from trusted sender. Using public key cryptography, Trusted Email allows legitimate companies to establish a key and self-sign certificate with their customer. All future emails from the trusted company are signed with the company's private key, which allows the user to verify its signature with the established public key. The authors implement the system with a Java proxy server and a plugin for Mozilla Thunderbird, demonstrating that the system can be implemented without modification to the current internet architecture.

Jakobsson and Siadati[172] proposed SpoofKiller, a web browser mechanism that attempts to modify the user's login behavior. The implementation

of the system forced the user to push the physical power button on their phone when they want to login to a legitimate website, which will signal the browser to verify the website's identity. SpoofKiller is implemented as a web browser on Android OS and a usability study was conducted. The participants in the study used SpoofKiller to login to a website everyday, and on the last day the website asked the participant to not perform the action and type in their email password instead. The result showed that 30% of the participants pressed the power button on the last day as quickly as any other day of the study, and 55% enter the same password they've entered into the website.

Khan[173] proposed an anti-phishing approach that utilized a one time password and an authentication token. The approach is similar to two- factor authentication, in that once the user provides their credential to a website, they receive a one-time password via email or SMS and the server sends an encrypted cookie with a short expiration time to the user. The user then has to submit the one-time password along with the unexpired cookie to login to the page. It is unclear if the scheme is secured against a main-in-the-middle attack.

Costigan[174] discussed how behavioral biometric authentication can be used as a way to ensure that attackers cannot fraudulently impersonate users and unlawfully access their information. Behavioral biometrics systems offer a layer of security that allows the users to authenticate themselves seamlessly in the background by tracking the way they interact with the device. Machine learning algorithm trained on the actions such as keystroke patterns and touch screen interactions as well as factors such as geo-location and transaction history is used to track the behavior of the user and report suspicious changes in the behavior to the service provider.

Many novel authentication protocols are designed to defend against phishing. Some of these protocols can deployed in a seamless fashion without any changes in the user's behavior except for the initialization process, and some of them required the user to adapt their behavior to the new system. Protocols that require the user to change their behaviors could be rejected by the user as it gets in the way of their primary tasks, and protocols that require modification of the server may face challenges in getting wide scale adoption. Although many of these protocols are provably secured against phishing, the wide varieties of the implementations required changes to the server of the

user's behavior, making it impractical to be deployed in the real world.

4.2.4 Summary

Overall many techniques were proposed to defend the user from the phisher. Offensive defense techniques attempt to poison the phisher's database with fake credentials, forcing the phisher to verify the credentials before selling them. Based on the approach, the server's response to the submission of fake credential can also be a leverage to identify phishing websites. Credential tracking approaches monitor the information that user provide to each website, and alert them when they are providing known sensitive information to an unverified website. While this approach would have a very low false negative rate, it may habituate the user to ignore the warnings if the warnings are displayed too frequently. Many novel authentication protocols are proposed with the phishing threat in mind, but most of them required modification to the server or the change in user's behaviors, which may hinder their wide adoption.

4.3 Evaluation of Mitigation Techniques

The works in this section focus on the evaluation of various mitigation techniques that are implemented and available for public uses. This includes the evaluation of blacklists, browser warnings, and browser toolbars. These mitigation techniques are evaluated in terms of their coverage against new phishing websites, and whether the technique can successfully alert the user of the threat.

4.3.1 Evaluation of Blacklists

Research efforts in this section evaluate the effectiveness of phishing blacklist approaches in term of coverage and latency, and identify vulnerabilities in blacklists that utilized crowd-sourcing approach.

Ludl et al.[175] tested the effectiveness of phishing URL blacklists maintained by Microsoft and Google. The authors started by collecting 10,000 phishing URLs from PhishTank over a period of 3 weeks, and used them

to determine the accuracy of the blacklists used by Internet Explorer and Firefox, which were maintained by Microsoft and Google respectively. They found that both blacklists were able to correctly identify less than 66% of the phishing URLs, but Google's blacklist was able to identify more than 90% of the phishing URLs that were live at the time of testing. The authors speculate that some of the URLs from PhishTank may not have been blacklisted because they went offline before they were considered.

Moore and Clayton[176] studied PhishTank, a crowd-based website that generate phishing reports based on URLs submitted by its users. These suspicious URLs are also voted on by users, and the result is published if the majority voted the website to be phishing. The authors examine over 200,000 phishing URLs on PhishTank and found that the behaviors of the users are consistent with a power law distribution, which means that a single highly active user's actions can greatly impact a system's overall accuracy. Another observation is that many URLs submitted to PhishTank are duplicates, which is an efficiency issue as users have to vote multiple times for phishing URLs from the same campaign. Also, by comparing PhishTank's performance with another proprietary source of phishing reports, the authors found that the crowd-based nature of PhishTank incurred delay in verification of URLs, and the number of non-overlapping URLs in the two sources motivates the implementation of universal feed shared between organizations.

Moore and Clayton[177] analyzed six months of phishing feeds from multiple sources, including two sub-contracted companies that take down phishing websites. They found that take-down companies were not aware of, or belatedly learned about huge phishing websites that were known to others, causing delay in the take-down of such websites. The authors estimated that by establishing a direct linkage between the longer take down time and the financial risks, around $330 million a year might be made safe if the take down companies were to share information with each other. Considering the reasons why the take down companies may not wish to share information, the authors concluded that the companies would benefit to some extent from data sharing, and recommended that the take down industries change its practice, especially since the major customers of these take down companies will benefit from the sharing of information.

Sheng et al.[178] used an anti-phishing testbed created in[90] to evaluate the effectiveness of phishing blacklists. In their experiment, eight anti-

phishing toolbars that use blacklist were evaluated against 191 fresh phishing websites that were less than 30 minutes old. They found that 63% of these phishing campaigns lasted less than two hours, and that the blacklists were ineffective against these phishing sites as most of the blacklists caught less than 20% of phish at hour zero. Furthermore, while the tools that used heuristics performed significantly better initially compared to those that only used blacklists, it took a long time for phish detected by heuristics to appeared on blacklists.

Overall, the problem with the blacklist approach can be summarized into two broad categories: timeliness, and coverage. The blacklist managed by PhishTank relies on the users to report suspicious URLs and vote on them to determine legitimacy, which can take a long time. Considering the lifetime of most phishing websites, many victims would have fallen for the phishing website before it appears on the blacklist. In term of coverage, it has been shown that blacklists that are available to the take down companies are not comprehensive, and that by sharing feeds with other proprietary companies, the takedown time for many phishing websites can been lowered substantially. Also, due to the user-centric approach of PhishTank, it is vulnerable to attacks in which compromised reputable users can manipulate the blacklist and make it less effective.

4.3.2 Evaluation of Browser Warnings

Research efforts in this section evaluate the effectiveness of various browser warnings and suggest ways to improve them.

Jackson et al.[179] performed a usability study of extended validation phishing defense where 27 users were asked to classify 12 websites as legitimate or fraudulent. Extended validation improves upon normal web certificate by providing not only the owner of a domain name but also the identity of the legitimate business. To test the effectiveness of this feature in helping users determine legitimacy of websites, the users were split into three groups: one group was trained in the use of the green address bar, one group saw the green address bar but received no training, and the last group was not shown any extended validation indicators at all. The authors found that users who received the training were more likely to classify legitimate websites that

aims to confuse the participants as legitimate, and picture-in-picture attacks were more likely to succeed against trained participants because they expected the extended validation warnings to be 100% accurate, which is not the case. Overall, the authors did not find that extended validation provided a significant advantage in identifying phishing attacks with the websites in the study.

Schechter et al.[180] evaluated how users responded to the absence of website authentication features such as HTTPS indicator, lack of authentication image in two-factor authentication, and a browser's security warning page. The result shows that no participants mentioned the absence of HTTPS indicators, while 3% of the participants chose not to enter their password when the authentication image is missing, and 47% of the participants chose not to proceed to the bank's website after seeing the certificate warning page.

Egelman et al.[181] studied the effectiveness of phishing warnings presented by web browsers by simulating a spear phishing attack in a laboratory setting. In the study, the participants were asked to make two purchases with their credit card from Amazon and eBay. After the purchase, the participant receives a phishing email with the sender spoofed to be the seller, asking them to click on a URL in the email to verify their order or it would be cancelled. Depending on their group, the participants were presented with different types of browser security warning when they clicked on the link in the emails. The authors found that significantly more users were helped by the active Firefox warning than the active Internet Explorer (IE) warning, and that displaying passive IE warning is no different than displaying any warning at all. There is also a significantly different in active IE warning compared to the control group, so displaying an active IE warning is still significantly better than displaying no warning.

Sunshine et al.[182] conducted a survey of internet users to examine their understanding and reactions to SSL warnings, and designed two new warnings based on the results of the survey. From a survey of 409 internet users, the authors found that risk perceptions are correlated with decisions to obey or ignore security warnings, and those who understand the warnings perceive different levels of risks associated with each warning. Following this survey, the authors conducted a between-subject laboratory experiment to study the participant's reaction to SSL certificate warnings presented by browsers, and two new warnings designed by the authors. The result shows that many par-

ticipants who used Firefox 3 were unable to override the warning and thus cannot accessed both websites. The results also indicated that the participants understood the risk presented in the novel warning and made their decision based on the website that they were visiting. However, the authors also noted that these warnings do not provide adequate protection against man in the middle attack, and that a better approach may be to minimize the use of SSL warnings by blocking the users from making insecure connections and not showing warnings in benign situations.

Lin et al.[183] investigated the effectiveness of domain highlighting in helping users identify phishing websites. Their experiment consisted of two phases. In the first phase, the participant was shown 16 websites, half of which were phishing websites, and were asked to rate how "safe" they perceived each website to be. In the second phase, the participant was asked to re-evaluate the same set of websites with an instruction to focus on the address bar area. Overall, the authors found that domain highlighting works, but nowhere near as well as they would like. The authors suggest that domain highlighting can be made more effective by making the domain name even more obvious, by drawing the user's focus onto the address bar, by reducing URL complexity, and by educating people about the importance of domain name and various obfuscation methods employed by the attackers.

The overall results agree that passive browser warnings are significantly less effective than active browser warnings, as it has been shown that many users do not notice the absence of HTTPS indicators and authentication images. Active warnings are found to be better because they interrupt the user's primary task and force the user to pay attention to the warning. In some cases, active warnings successfully stop the user from accessing a phishing website because the user does not know how to bypass the warning. Some users also expect the security indicators to be always accurate, rendering them vulnerable to visual spoofing attack of the indicators. It is also the case that if a succinct explanation of the risk is provided in an active warning, then the participants would understand the threat and heed the warning.

4.3.3 Evaluation of Browser Toolbars

Works in this section focus on the evaluation of browser toolbars in terms of its coverage against phishing URLs in the real world, and whether they can effectively protect the users.

Wu et al.[184] determined that security toolbars, as well as browser's address and status bars, failed to prevent users from being spoofed by high quality phishing attacks. In their role-play experiment, participants were asked to process a list of emails, some of which were phishing emails. During the study, participants interacted with a simulated IE browser with security toolbars that are controlled by the researchers. A total of three different types of security toolbars are tested in the study: A Neutral Information toolbar which highlights the website information such as domain name and hostname, SSL-Verification toolbar which differentiates sites that use SSL from those that do not, and System-Decision toolbar that uses a traffic light approach to warn users against potential phishing sites. From the experiment, the authors were able to determine that the security toolbars all failed to prevent users from being spoofed by high quality phishing site, with 45% spoof rate for Neutral-Information toolbar, 38% for the SSL-Verification toolbar, and 33% for the System-Decision toolbar.

Zhang et al.[185] created an anti-phishing tool test bed and evaluated 10 anti-phishing tools against legitimate and phishing URLs. The test- bed consisted of a task manager and a set of workers, where each worker is responsible for evaluating a single tool. A total of 200 phishing URLs from PhishTank and APWG and 516 legitimate URLs were used to to test the effectiveness of 10 anti-phishing tools, and the authors found that the source of the phishing URLs and the freshness of the URLs significantly impact the effectiveness of each tool. Only one tool was able to catch more than 60% of phishing website from both sources, but it still missed 25% of APWG phishing URLs and 32% of PhishTank URLs. One tool had a consistently high catch rate of over 90%, but also had a 42% false positive rate. This result indicated that there is no single technique that will always outperform others, for identifying phishing websites. The authors also noted that every tools that were tested can be exploited by the attacker, namely CDN can be use to circumvent tools that use blacklists and a tool that analyzed a fully loaded web page can be rendered useless by a web page that takes infinitely

long to finish loading.

These evaluations reveal that browser security toolbar cannot adequately protect the users. From the lab experiment, the authors noted that participants failed to continuously check the security indicators, possibly because maintaining security is not their primary goal, and the participants who notice suspicious signs either did not know how to interpret the warning or explained them away. A large scale evaluation of 10 anti-phishing toolbars against verified phishing URLs also reveal a severe lack of coverage, and the researchers also demonstrated that all of the security toolbars can be circumvented.

4.3.4 Evaluation with Eye-Tracking Softwares

The works in this section involved the use of eye tracking software to evaluate browser security indicators and suggest they can be improved.

Darwish and Bataineh[186] studied the behavior of phishing victims' eyes on several predetermined area of interests, and empirically presented users' evaluation of several web pages by using eye tracking softwares. The authors specify the area of interest to be the domain name area, digital certificate indicator, HTTPS indicator, and the content of the web page. They found that participants who correctly identified the phishing websites were interested in the domain name area, and that digital certificate area was ignored by all groups of users except when it is present with contrasting green background. Participants check the HTTPS indicator on half of the websites, and the mean time to fixation indicates that the participants check the HTTPS indicator, domain name, digital signature, and page content in order. However, login area was spotted before any other page contents, which could be a security risk since the participants may be fixated on the task of logging into the website and did not pay attention to the security cues in the other areas of the website's content.

Whalen and Inkpen[187] evaluated the effectiveness of web browser visual feedback security mechanisms by performing an eye-tracking experiment. Specifically, the authors investigated which security cues were ignored, which were easily noticed, and which were hard to find. The results suggest that the lock icon is the security cue that is most often looked at, but only a few

participants interact with it. Certificates are also seldom used, and some of the more experienced users do not notice any security cues. They also found that small browser icons can be easily misidentified, and users tend to stop looking for security cues after they have signed into a website. From this, the authors suggest that standardized location of security cues across browsers would be beneficial to the user, and that maybe certificate data should have its own icon separate from the lock to make it easier to find.

Somewhat surprisingly, the results from these research indicate that users do look at the security indicators on the browser's chrome before the web page content. HTTPS indicator and the lock icon are the first things that users check, and they are the indicators that were looked at most often. Certificates are seldom looked at, and the small icons in the browsers are reported to be misidentified by the users. Users also noticed the login portion of the web page first, suggesting that they may be so focused on the task of logging into the website that they do not check other contents in the web page for security cues.

4.3.5 Summary

Overall, the evaluations presented in this section illustrate that the current mitigation techniques are inadequate at protecting the users. The current blacklists are not updated in a timely manner, and the gap in coverage between public and proprietary blacklists cause many phishing websites to live longer than it should have. Passive browser warnings are shown to be ignored by the users, and active warning that interrupt the users are found to be effective, especially if the users cannot workaround the warning to proceed to the insecure website. The browser toolbars that were evaluated in the studies have a poor coverage, and circumvention techniques were demonstrated for all of them. Results from eye tracking studies suggest that while the users do look at the some security indicators in the browser, they do not take full advantage of them and stop looking for them once they have logged into a website.

4.4 User Education

User education is an important factor in the fight against phishing. Even with all of the technical counter measures that were put in place to assist the users, the last line of defense is always the users themselves. "How can we educate users to recognize signs that can help them identify phishing emails?" is the question that needs to be answered. Several works evaluated the quality of the security tips that are currently available to the users, and identified how they could be improved. New education approaches were also proposed and evaluated in terms of their effectiveness to prevent users from falling for phish. These approaches include embedded training, phishing IQ tests, comics, and games.

4.4.1 Evaluation of Security Tips

The research works in this category aimed to evaluate the effectiveness of the current state of security awareness tips and suggestions.

Alnajim and Munro[188] proposed an effectiveness criteria for evaluation of effectiveness of security tips and attempted to identify a small subset of effective anti-phishing tips that users can focus on. After surveying the online fraud tips for both businesses' and users' tips, they identified 21 anti-phishing tips that were applicable to phishing websites. The tips were evaluated against a sample of 42 phishing websites from APWG's archive, and result indicated that even the best tip did not satisfy all of their effectiveness criteria, as it had a possibility to produce false negative or false positive on its own.

Kirlappos et al.[189] evaluated an anti-phishing tool in a laboratory setting and found that when tempted by a "good deal", participants did not focus on the warning and instead looked for signs that they thought confirmed a site's trustworthiness. In the study, a list of six websites were shown to the participants with an instruction to buy tickets from one of them within five minutes. The participants received more compensation if they managed to purchase the ticket at a lower price. The result from this experiment showed that while most participants who used a anti-phishing tool chose to buy from a website that the tool reported to be safe, a significant number of participants chose to buy from a website labeled as potentially risky but had better

potential rewards. In a debriefing interview, these participants reported that factors such as previous experience with a brand, logos, certifications and social media references influenced their confidence in the websites. These reports suggested that the current model of security advice given to the users were not adequate, in that users may simply used trust indicators to guide their choices without knowing how it verified the website's legitimacy, making them vulnerable to malicious websites that spoofed these trust indicators.

Harley[190] argued that users' rejection of security advices was entirely rational from an economic perspective: the advices prevented them from the direct loss of attacks but burdened them with indirect cost in the form of efforts. Since victimization was rare and imposed a one-time cost while security advice applied to everyone in an ongoing fashion, the burden of following security advices ended up being larger than the burden caused by the loss. In teaching users to recognized phishing sites by reading URLs, the security advices quickly evolved from seemingly reasonable advices such as "watch for IP address in URL" into a requirement that users needed to know how to parse URLs. Also, the actual benefit to users in avoiding this type of phishing attacks depended on how the banks and the users split the loss. It appeared that most banks would shoulder the entire loss in most cases. In another area, the benefit from teaching users about SSL certificate and related browser warnings was a protection against man in the middle attack. However, from the user's perspective, almost all of the certificate errors that they experienced were false positives. From a big picture, the burdens from following the security advices could resulted in a net loss to the user.

These results suggested that a change of direction in security awareness training is needed. To help users avoid harm, the effort should begin with a clear understanding of the actual harm that the users face, and a realistic understanding of the constraints imposed by the advice. The security solutions should also be tailor based on its relevancy to how users make security decision in their everyday activity, instead of flooding them with large amount of information.

4.4.2 Embedded Training

Embedded training is a type of training system that delivers the training material to the users through a simulated phishing email. Aiming to take advantage of the moment that the users fall for a phishing email as a learning opportunity, these emails contain an embedded URL that, when clicked, delivers a training message to the users. The training message can varies from a simple warning about an unsafe action to a full-fledged training materials, depending on the goals of the system designer. A real world deployment of the system can be use not only as a vector to deliver training material, but also to measure the recipient's ability to distinguish phishing emails as well.

Kumaraguru et al.[191] created PhishGuru, an embedded training system that sends out fake phishing emails with an embedded link that display phishing training material when clicked. To evaluate the effectiveness of the system in real life, the authors conducted a study where phishing emails are sent to the participants drawn from a universities population. The authors were able to verified that participants who received the embedded training email could identify phishing emails better than those in the control conditions after they received the first intervention. Participants who saw the intervention twice also performed better than those who saw the intervention once. The training did not affect the participant's ability to identify legitimate emails. The authors also conducted an experiment to determine how well the users retained knowledge gained through PhishGuru, and how well they transferred this knowledge to identify other types of phishing emails[77]. The study was conducted in a laboratory setting, where the participant role played as a user and went through a list of emails addressed to them, behaving as they would in real life. The participants were split into three groups: one group receive an embedded training email, one group receive the same training material in the body of an email, and one group did not receive any training. The result showed that the participants learn more effectively and retain more knowledge about how to avoid phishing when trained with embedded training, and participants who took the embedded training also transfer knowledge better in that they were better at identifying different type of email as phishing. Sheng et al.[64] also evaluated the effectiveness of PhishGuru's style of interventions. They found that participants fell for 47% of the phishing website on average before training and 28% after the

training, which was significantly better than the control group who did not received any training between the two role-plays.

Alnajim and Munro[192] proposed an anti-phishing approach that uses embedded training intervention for phishing websites detection (APTIPWD). The architecture of the proposed anti-phishing system involved a proxy server that sit between the user and the Internet, and a blacklist of known phishing websites. When a user requested a website that is on the blacklist, the proxy listened for a form submission, intercepted it, and displayed an intervention message to the user. The authors evaluated their approach in a role-play experiment with 36 participants, in which he participants were asked to read and handle 14 emails, some of which are training and phishing emails. The result indicated that the participants in all training conditions (no training, information email, embedded training email) were nearly equal in terms of reaction to legitimate and phishing emails before the training, but the participants in the embedded training group performed significantly better after receiving the training.

Gupta and Kumaraguru[193] explored phishing trends and analyzed the effectiveness of an Anti-Phishing landing page. Anti-Phishing landing page was a web page designed to take advantage of the most teachable moment when the user just clicked on a link to a phishing website that was taken down by ISPs. Instead of displaying an error, the ISP redirected the user to the landing page, which informed the user that they just tried to access a phishing website and offered tips about how the user could protect himself against phishing. Analysis of the log files showed that 46% of users clicked lesser number of phishing URLs in April 2014 than in January 2014.

Caputo et al.[194] explored spear phishing awareness and embedded training by sending phishing emails to employees of a corporation. Similar to the experiment by Kumaraguru et al.[77], the authors tested the participants' retention of the training material over the period of 90 days by sending out three phishing emails. All of the emails contained a link that either redirected to a website containing the training materials, or a website that simply alerted the participants that they have just been phished. Their result indicated that the training does not have a significant effect on the likelihood that a participant would fall for phishing email, and that based on the amount of time that participants spend on the training page, many of them did not stay on the page long enough to fully read the training material, and hence they

effectively did not receive the training.

Jansson and Solms[195] conducted a naturalistic phishing experiment to determine whether a simulated phishing attack together with embedded training contribute towards strengthening users' resistance toward phishing attacks. Four different types of phishing emails were used in the experiment: an email claiming a crashed database, an email claiming the recipient won a lottery, an email claiming a virus scanner in the attachment, and an email claiming to have pictures of pretty women. If the recipient of the email reacted insecurely (i.e. by clicking on the link or opening the attachment), they will be presented with a warning page with a link to take additional training. Two rounds of emails were sent out to an institution population of 25,579 a week apart, with 9,273 users recorded being active on the email system on the first week and 8,231 in the second week. The result showed that the number of users who reacted insecurely dropped by 11% on the second week, and that the phishing email that claimed to have pictures of pretty women was the most enticing phishing email.

Overall, the research in this area has shown that embedded training was an effective method: users are able to better identify phishing emails and avoid performing unsafe actions after receiving the training. The users also retained the knowledge from the training from over time, and able to applied it to other kinds of phishing emails. Embedded training also did not affect the way users perceived legitimate email.

4.4.3 Phishing IQ Tests

Phishing IQ test is a test designed to measure the test taker's ability to identify phishing emails. In a phishing IQ test, a sequence of screenshots of email messages is shown to the test taker, and the test taker has to identify whether the email is legitimate or fraudulent. Due to the general and static nature of the test, several works were completed to evaluate the effectiveness of the phishing IQ tests. Modifications to the phishing IQ test were also proposed.

Anandpara[196] argued that phishing IQ tests failed to measure the user's susceptibility to phishing attacks. This insight came as a result of an experiment where participants took a phishing IQ tests that has different number

of questions corresponding to phishing email. The authors found no correlation between the actual number of phishing email's questions and the number of emails that the subject reported to be phishing. Furthermore, after the subjects received the training and retook the test with a different set of questions, the numbers of emails reported to be phishing in the second test was substantially larger those that of the first test, indicating that the measurable effect of phishing education was increased concern, not increased ability.

Robila and Ragucci[197] designed a phishing user education by modifying phishing IQ tests to suit the university's context. By examining browser's history from the university's computer laboratory and other resources that are available to the students on campus, the authors decided to include websites of 12 companies in the IQ survey. 48 students from an Introduction to Computing course took the surveys, achieving a correctness rate of 57%. A post- study survey was administered to evaluate the educational value of the survey, and the responses indicated that 78% of the students were not aware of phishing prior to the IQ survey, and 93% acknowledge receiving possible phishing emails in the past.

Werner and Courte[198] analyzed the effectiveness of phishing IQ test as a learning tool from the perspective of college students. In their experiment, participants took the SonicWall Phishing IQ test, then received a description and explanation of the security indicators in both the legitimate emails and phishing emails. The results indicated that the phishing IQ test significantly improve the participant's perceived ability and confidence in detecting phishing emails, and 80% of the participants reported that the phishing IQ test was helpful. As a follow up to this study, Bekkering et al.[199] evaluated the improvement in the test taker's ability to differentiate between phishing emails and legitimate emails after taking a phishing IQ test. The participants were asked to identified phishing emails from a set of emails, took the phishing IQ test, then performed the identification task again on another set of emails. The results indicated a modest significant increase in the performance of the participants after the training, but the results were not uniformly positive.

Tseng et al.[200] proposed a framework for a dynamic generation of content for an anti-phishing education game. The anti-phishing game described by the authors was similar in structure to a phishing IQ test. The anti-phishing education game helped users build knowledge of phishing based on a few

static phishing examples. This made it difficult for the users to applied the knowledge in the real world. Hence, the authors proposed two models: one that records the features of a phishing web page and one that describes web page obfuscation techniques. These models were then utilized in the game to dynamically generate phishing scenarios.

These results suggested that while a static phishing IQ test may be ineffective at measuring the participant's ability to detect phishing email, modification of the materials presented in the test can be made to improve its efficacy. Specifically, the use benefited from the tests that used websites that closely related to what the users encountered in real life.

4.4.4 Comics

Some works proposed the use of cartoons or comic strips as a medium for teaching computer security. The motivation for this type of user education came from the observation that many security education presented the users with set of instructions to follow without proper explanation. Comics, on the other hand, presented the security advice in a way that the user can relate to: by presenting a situation and implications of different actions to the user.

Srikwan and Jakobsson[201] described the design principles behind SecurityCartoon.com, the first cartoon-based approach to teaching risks to internet users. The authors argued that the cartoon-based approach is likely to produce better long-term effects than current education efforts and identified some problems with the current practice. These problems included the use of advice that are hard to follow, not very important, and may makes the user more vulnerable to other attacks. With these in mind, the goal of the cartoons-based approach is based on four core principles: research driven content selection tailored around observed user's behaviors, education message that are accessible and easy to read, immersion in the content through repeated communication of the message framed in different ways to the user, and adaptability of the material to the ever-changing threat.

Kumaraguru et al.[202] designed an embedded training system to teach users about the risks associated with phishing and how to identify and avoid phishing attacks in emails. The authors evaluated the effectiveness of three different type of intervention, security notice, text and graphic, and comic

strips. In the experiment, each participant was shown 19 email messages consisted of legitimate messages, training messages, spam messages, and actual phishing messages. They found that the participants in the comic strips intervention group perform significantly better than the security notice group and the "text and graphic" group but found no significant different between the performance of the "text and graphic" group and the security notices group.

While there were not many research done in the area, the result of one experiment showed that comic strips were significantly more effective than security notice at educating users about phishing.

4.4.5 Games

Several works explored the use of interactive game as a medium to teach people about phishing. Specifically, there are two games that were implemented on the web and mobile platform that teaches user about the structure of a URL and how to recognize phishing URLs.

Sheng et al.[203] created Anti-Phishing Phil, a game that teaches users about how to identify phishing URLs. In the game, the player assumed the role of Phil, a young fish living in the Interweb Bay. Phil wanted to eat worms. Each real worm in the game was labeled with a legitimate URL, and each fake worm was labeled with phishing URL. The goal of the player was to eat the real worms and rejected the bad worms before running out of time. The game proceeded in rounds, with each round focusing on different types of phishing URLs. A study was conducted to test the effectiveness of the game in training users by using a between-subjects experiment where the subjects were split into three groups: one group was trained with existing training material, one group was trained with Anti-Phishing Phil materials, and one group played the actual game. The authors found that subjects in the game condition performed best overall. This result showed that interactive games could be a promising way of teaching people about strategies to avoid falling for phishing attacks.

Similar to Anti-Phishing Phil, Arachchilage et al.[204] developed a prototype anti-phishing game on a mobile platform as an educational tool to teach users how to recognize phishing URLs. In the game, the user assumed the

role of a small fish in a big pond. The objective of the game is to eat worms that are labeled with legitimate URLs and avoid the works that are labeled with malicious URLs. The authors conducted a user study of the prototype game by having 20 participants identified legitimate URLs from a list of 10 suspicious URLs prior to playing the game. The participants then evaluated the legitimacy of another 10 suspicious URLs after having played the game. The result indicated that the participants were better at identifying phishing URLs after they have played the game.

Overall, several research works have shown promising results that interactive games could be an effective approach to help users detect phishing attempts. The results also indicated that users who were trained by playing the game were significantly better at identifying phishing URLs than those who were trained by reading traditional security notices.

4.4.6 Summary

In conclusion, the works in this area indicated that the current security tips available to the user is ineffective by themselves, and the proposed alternate education methods show promising results in helping the users recognize phishing attempts and avoid falling for them. Current security tips are numerous, and most of them neither provide the users with a proper understanding on how to avoid harms nor is effective on their own. On the other hand, new methods such as embedded training were shown to be effective at reducing the rate in which users fall for phishing attempt. Embedded training can also be use as a tool to evaluate the users' ability, and target the training at those who need it. Phishing IQ tests were shown to be ineffective by itself as an educational tool to help the user identify phishing emails, but adaptation of its framework to use with a set of phishing emails that target a particular population has promising results. Comics and Games were considered as an alternate medium for conveying education materials with encouraging results.

4.5 Chapter Summary

In this chapter, we have discussed detection and defense techniques, as well as the evaluation of some of them. Many detection schemes were proposed with varying degrees of success. While no one scheme is capable of detecting all phishing websites, some of the tools can be deployed together to complement each other. A number of defensive techniques were also proposed, but most of them either required a change to the server infrastructure or a change in user's behavior, both of which are obstacles for wide scale adoption. Evaluations of the anti-phishing tools revealed that most of the tools lack coverage against new phish and can be easily circumvented by the phisher. Some of the tools that are too passive failed to get the user's attention, and tools that interrupt the user's primary task to display the warning are found to be significantly more effective.

CHAPTER 5

RESEARCH ETHICS

Due to its nature, a number of considerations needed to be made in the designing of a phishing experiment.

Finn and Jakobsson[205, 206] described the ethical and procedural aspects of phishing experiment. Naturalistic experiments mimic real phishing attacks where fake phishing emails are sent by the researcher to a group of subjects in a way that cannot be distinguished from real life phishing attacks. This approach have the benefit of being able to measure the danger of attacks but poses an ethical issues since the experiment itself constitute a phishing attack. The main ethical concerns for naturalistic phishing experiments are the use of deception and complete waiver of informed consents. These two components are necessary in order to not alert the subjects of the attack.

Another important aspect is the debriefing process, in which the subjects are given explanation of the nature and purpose of the deception and the researchers attempt to alleviate any discomfort the subjects may experience upon learning that they have been deceived. The authors noted that naturalistic phishing experiment was a unique case, where the primary source of harm to the subject may be the debriefing process itself. In naturalistic experiment, the subjects who were aware of phishing attacks were likely to not be fooled by the experiment, and disregard the email as another one of the many phishing emails that they encountered on a regular basis. Subjects who were not aware of phishing attacks may be fooled by the experiment, but any information that they provided would be discarded by the researcher and no physical or financial harm would result. However, if the latter group of subjects were debriefed, they may be upset, anxious, or angry that they were deceived. Also, depending on the debriefing process, these subjects may not be able to have their immediate concerns addressed by the researchers. Hence, depending on the experiment, a waiver of the debriefing process may be considered. Here, we include a review of one of the authors' case study in

which a waiver of debriefing process was granted.

Case Study Jakobsson and Ratkiewicz [207] designed an experiment that used eBay's internal messaging service to discover the email address of eBay users. eBay's internal messaging service allows eBay users to message each other using only their usernames without revealing the email address of either party. The authors were able to identify an artifact in the messaging system's design that allow an eBay user other than the intended recipient to reply to a message if they have access to the email that was sent to the recipient. The authors used this functionality to craft an apparent phishing email that allowed them to verify if a user clicked on a phishing link and entered their credentials.

In this experiment, the authors chose to avoid the debriefing phase as it would significantly increased the harm done to the subjects. If the subjects detected the apparent fraud attempt, they would simply not provide their credentials to eBay, and they would perceived that nothing unusual had happened. If the subjects provided their credentials to the website, nothing unusual had happened to them as well since they are authenticating into the actual eBay website. Debriefing either group of subjects would greatly increase the harms done to them, which is the opposite of the purpose of debriefing.

CHAPTER 6

CONCLUSION

6.1 Summary

Phishing presents a widespread threat to Internet users and, as such, a myriad of research has studied the field. We created a taxonomy of these works and identified four major categories: attack characteristics, victim profiling, mitigation techniques, and user education.

A large portion of phishing research studies elements that affect campaign efficacy. One of the easiest ways to make a campaign more effective is called contextual phishing. Contextual phishing more thoroughly fools users by including information specific to its targets and was found to be generally more effective in successfully phishing a larger number of victims. We also discussed alternative attack vectors and various aspects of phishing that are not observable through phishing emails and websites.

The only factor that was identified to directly correlate to a user's susceptibility to phishing is habituation. Users are more likely to fall for the attack when they habitually process emails without much attention. Women and younger individuals were reported to be more susceptible to phishing, but this result may be skewed by factors such as their experience with phishing education and Internet expertise. There is no consensus about the influence of personality or cognition reflection to phishing susceptibility.

Many detection and defense schemes were proposed to counter phishing with varying degrees of success. While no one scheme is capable of detecting all phishing websites, many machine learning tools reported a very high accuracy rate with a modest number of false positives. A number of defensive techniques were also proposed, but most of them exhibit requirements that could prohibit wide scale adoption. Evaluations of many anti-phishing tools revealed that most of the tools lack coverage against unseen phishing attacks

and thus can easily be circumvented by the phisher.

Novel user education methods proved effective in helping users recognize and avoid phishing attempts. Embedded training was shown to be effective at reducing the rate at which users fall for phishing attempts. Phishing IQ tests were shown to be ineffective by themselves as an educational tool, but the adaptation of the phishing IQ framework to use with a set of phishing emails that target a particular population has shown promising results. Researchers have also successfully made training more palatable to users by presenting phishing training in comics and games.

6.2 Future Works

Based on our taxonomy, we identified several studies that would be beneficial to conduct in an effort to expand our knowledge of phishing.

Characteristics of Current Phish We found that research we encountered is heavily skewed toward the proposal and implementation of mitigation techniques, with high emphasis on the development of automatic detection frameworks. The research that studies attack characteristics of phishing are also relatively outdated, with most current work focusing on applying machine learning techniques to detect phishing. As phishing is constantly evolving, we believe that investigating trends in characteristics of phishing campaigns both in terms of content and attack vector is needed to better understand the current state of attacks.

Continued Evaluation While many of the proposed detection techniques are implemented and evaluated against a sample of phish captured in the wild, not many systems were analyzed against unseen phishing campaigns over time. As we have seen, phishing evolves over time and the detection techniques that work today may prove not quite as useful in the near future. Hence, we believe that a continued evaluation of the proposed techniques are needed to determine their efficacy in detecting this ever-changing threat.

Change the user's perspective Currently, most financial institutions shoulder most if not all of the financial loss due to phishing by offering fraud

protection to their clients. This is without a doubt beneficial to users, but it also puts them in a mindset where they do not suffer consequences from the phishing attacks, potentially making them care less about securing their information. We believe that a change in user's education is necessary. More emphasis should be put on consequences of clicking on malicious links than on submitting information to a phishing website.

6.3 Conclusion

Phishing is a form of attack that aims to steal information from users by impersonating a trusted entity. This thesis presents a taxonomy of phishing that encompasses research studying the attack, the victim, and the defense. We found that novel education methods such as embedded training are effective at teaching the users about phishing and other risks associated with clicking on links in emails. We believe education can be something of a silver bullet. Many current and proposed phishing defenses (such as those used by financial institutions) that keep users in the dark about the dangers of visiting malicious websites inadvertently open users to other attacks associated with visiting malicious websites. It is essential that users show discretion when visiting websites. Further, we found work on attack characteristics to be outdated and overly focused on attacks associated with phishing emails. Widespread adoption of social networks provides users with alternative communication methods that could be exploited by phishers. Research needs to be conducted that studies current attack characteristics and attack vectors. This also implies the need to evaluate the adaptability of proposed detection techniques to this constantly evolving threat. The accuracy of many machine learning based methods were evaluated based on a sample of phishing attacks which is a static dataset. We argue that future work should focus on the evaluation of these systems over time in order to gauge their efficacy in a real-world environment.

REFERENCES

[1] T. McCall, "Gartner survey shows phishing attacks escalated in 2007; more than $3 billion lost to these attacks," *Stephane GALLAND*, 2007.

[2] A. van der Merwe, M. Loock, and M. Dabrowski, "Characteristics and responsibilities involved in a phishing attack," in *Proceedings of the 4th international symposium on Information and communication technologies*. Trinity College Dublin, 2005, pp. 249–254.

[3] J. Milletary and C. C. Center, "Technical trends in phishing attacks," *Retrieved December*, vol. 1, no. 2007, pp. 3–3, 2005.

[4] H. Huang, J. Tan, and L. Liu, "Countermeasure techniques for deceptive phishing attack," in *New Trends in Information and Service Science, 2009. NISS'09. International Conference on*. IEEE, 2009, pp. 636–641.

[5] H. Huang, S. Zhong, and J. Tan, "Browser-side countermeasures for deceptive phishing attack," in *Information Assurance and Security, 2009. IAS'09. Fifth International Conference on*, vol. 1. IEEE, 2009, pp. 352–355.

[6] J. Zhang, S. Luo, Z. Gong, X. Ouyang, C. Wu, and Y. Xin, "Protection against phishing attacks: A survey," *IJACT: International Journal of Advancements in Computing Technology*, vol. 3, no. 9, pp. 155–164, 2011.

[7] M. Khonji, Y. Iraqi, and A. Jones, "Phishing detection: a literature survey," *Communications Surveys & Tutorials, IEEE*, vol. 15, no. 4, pp. 2091–2121, 2013.

[8] A. Almomani, B. Gupta, S. Atawneh, A. Meulenberg, and E. Almomani, "A survey of phishing email filtering techniques," *Communications Surveys & Tutorials, IEEE*, vol. 15, no. 4, pp. 2070–2090, 2013.

[9] C. F. M. Foozy, R. Ahmad, and M. F. Abdollah, "Phishing detection taxonomy for mobile device," *International Journal of Computer Science*, vol. 10, pp. 338–344, 2013.

[10] B. Harrison, A. Vishwanath, Y. J. Ng, and R. Rao, "Examining the impact of presence on individual phishing victimization," in *System Sciences (HICSS), 2015 48th Hawaii International Conference on*. IEEE, 2015, pp. 3483–3489.

[11] T. N. Jagatic, N. A. Johnson, M. Jakobsson, and F. Menczer, "Social phishing," *Commun. ACM*, vol. 50, no. 10, pp. 94–100, Oct. 2007. [Online]. Available: http://doi.acm.org/10.1145/1290958.1290968

[12] A. Karakasiliotis, S. Furnell, and M. Papadaki, "Assessing end-user awareness of social engineering and phishing," 2006.

[13] M. Blythe, H. Petrie, and J. A. Clark, "F for fake: four studies on how we fall for phish," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2011, pp. 3469–3478.

[14] J. Wang, R. Chen, T. Herath, and H. R. Rao, "An exploration of the design features of phishing attacks," *Information Assurance, Security and Privacy Services*, vol. 4, p. 29, 2009.

[15] A. Ferreira and G. Lenzini, "An analysis of social engineering principles in effective phishing," in *Socio-Technical Aspects in Security and Trust (STAST), 2015 Workshop on*, July 2015, pp. 9–16.

[16] A. Ferreira, L. Coventry, and G. Lenzini, "Principles of persuasion in social engineering and their use in phishing," in *Human Aspects of Information Security, Privacy, and Trust*. Springer, 2015, pp. 36–47.

[17] J. S. Downs, M. B. Holbrook, and L. F. Cranor, "Decision strategies and susceptibility to phishing," in *Proceedings of the second symposium on Usable privacy and security*. ACM, 2006, pp. 79–90.

[18] A. Tsow and M. Jakobbson, "Deceit and design: a large user study of phishing," *Indiana University*, 2007.

[19] M. Jakobsson, A. Tsow, A. Shah, E. Blevis, and Y.-K. Lim, "What instills trust? a qualitative study of phishing," in *Financial Cryptography and Data Security*. Springer, 2007, pp. 356–361.

[20] B. Fogg, J. Marshall, O. Laraki, A. Osipovich, C. Varma, N. Fang, J. Paul, A. Rangnekar, J. Shon, P. Swani et al., "What makes web sites credible?: a report on a large quantitative study," in *Proceedings of the SIGCHI conference on Human factors in computing systems*. ACM, 2001, pp. 61–68.

[21] B. J. Fogg, C. Soohoo, D. R. Danielson, L. Marable, J. Stanford, and E. R. Tauber, "How do users evaluate the credibility of web sites?: A study with over 2,500 participants," in *Proceedings of the 2003 Conference on Designing for User Experiences*, ser. DUX '03. New York, NY, USA: ACM, 2003. [Online]. Available: http://doi.acm.org/10.1145/997078.997097 pp. 1–15.

[22] R. Dhamija, J. D. Tygar, and M. Hearst, "Why phishing works," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '06. New York, NY, USA: ACM, 2006. [Online]. Available: http://doi.acm.org/10.1145/1124772.1124861 pp. 581–590.

[23] S. Grazioli, "Where did they go wrong? an analysis of the failure of knowledgeable internet consumers to detect deception over the internet," *Group Decision and Negotiation*, vol. 13, no. 2, pp. 149–172, 2004.

[24] J. Spaulding, S. Upadhyaya, and A. Mohaisen, "The landscape of domain name typosquatting: Techniques and countermeasures," *arXiv preprint arXiv:1603.02767*, 2016.

[25] I. Waziri, "Website forgery: Understanding phishing attacks and non-technical countermeasures," in *Cyber Security and Cloud Computing (CSCloud), 2015 IEEE 2nd International Conference on*. IEEE, 2015, pp. 445–450.

[26] J. S. Downs, M. Holbrook, and L. F. Cranor, "Behavioral response to phishing risk," in *Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit*. ACM, 2007, pp. 37–44.

[27] D. K. McGrath and M. Gupta, "Behind phishing: An examination of phisher modi operandi." *LEET*, vol. 8, p. 4, 2008.

[28] L. OpenDNS, "Phishtank," available: http://www.phishtank.com/.

[29] "Markmonitor," available: https://www.markmonitor.com/.

[30] S. Chhabra, A. Aggarwal, F. Benevenuto, and P. Kumaraguru, "Phi. sh/$ ocial: the phishing landscape through short urls," in *Proceedings of the 8th Annual Collaboration, Electronic messaging, Anti-Abuse and Spam Conference*. ACM, 2011, pp. 92–101.

[31] A. Y. Fu, X. Deng, and W. Liu, "A potential iri based phishing strategy," in *Web Information Systems Engineering–WISE 2005*. Springer, 2005, pp. 618–619.

[32] M. Jakobsson, "Modeling and preventing phishing attacks," in *Financial Cryptography*, vol. 5, 2005.

[33] M. Jakobsson and S. Stamm, "Invasive browser sniffing and countermeasures," in *Proceedings of the 15th international conference on World Wide Web*. ACM, 2006, pp. 523–532.

[34] M. Jakobsson, A. Juels, and J. Ratkiewicz, "Remote harm-diagnostics," 2006.

[35] S. Gupta, P. Gupta, M. Ahamad, and P. Kumaraguru, "Abusing phone numbers and cross-application features for crafting targeted attacks," *arXiv preprint arXiv:1512.07330*, 2015.

[36] R. C. D. Jr., C. Carver, and A. J. Ferguson, "Phishing for user security awareness," *Computers & Security*, vol. 26, no. 1, pp. 73 – 80, 2007. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0167404806001581

[37] A. J. Ferguson, "Fostering e-mail security awareness: The west point carronade," *Educase Quarterly*, vol. 28, no. 1, pp. 54–57, 2005.

[38] T. Steyn, H. A. Kruger, and L. Drevin, "Identity theftempirical evidence from a phishing exercise," in *New Approaches for Security, Privacy and Trust in Complex Environments*. Springer, 2007, pp. 193–203.

[39] J. Mohebzada, A. El Zarka, A. H. BHojani, and A. Darwish, "Phishing in a university community: Two large scale phishing experiments," in *Innovations in Information Technology (IIT), 2012 International Conference on*. IEEE, 2012, pp. 249–254.

[40] H. Holm, W. R. Flores, and G. Ericsson, "Cyber security for a smart grid-what about phishing?" in *Innovative Smart Grid Technologies Europe (ISGT EUROPE), 2013 4th IEEE/PES*. IEEE, 2013, pp. 1–5.

[41] A. P. Felt and D. Wagner, *Phishing on mobile devices*. na, 2011.

[42] Z. Xu and S. Zhu, "Abusing notification services on smartphones for phishing and spamming." in *WOOT*, 2012, pp. 1–11.

[43] Y. Niu, F. Hsu, and H. Chen, "iphish: Phishing vulnerabilities on consumer electronics." in *UPSEC*, 2008.

[44] S. Abu-Nimeh and S. Nair, "Circumventing security toolbars and phishing filters via rogue wireless access points," *Wireless Communications and Mobile Computing*, vol. 10, no. 8, pp. 1128–1139, 2010.

[45] S. Abu-Nimeh and S. Nair, "Phishing attacks in a mobile environment," *SMU HACNet Lab Southern Methodist University Dallas*, 2006.

[46] H. Kim and J. H. Huh, "Detecting dns-poisoning-based phishing attacks from their network performance characteristics," *Electronics Letters*, vol. 47, no. 11, pp. 656–658, 2011.

[47] M. Jakobsson and A. L. Young, "Distributed phishing attacks." *IACR Cryptology ePrint Archive*, vol. 2005, p. 91, 2005.

[48] T. Moore, R. Clayton, and H. Stern, "Temporal correlations between spam and phishing websites." in *LEET*, 2009.

[49] "Spamcop," https://www.spamcop.net/.

[50] "Anti-phishing working group," http://www.antiphishing.org.

[51] T. Moore and R. Clayton, "Examining the impact of website take-down on phishing," in *Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit*. ACM, 2007, pp. 1–13.

[52] T. Moore and R. Clayton, "An empirical analysis of the current state of phishing attack and defence." in *WEIS*, 2007.

[53] T. Moore and R. Clayton, "Evil searching: Compromise and recompromise of internet hosts for phishing," in *Financial Cryptography and Data Security*. Springer, 2009, pp. 256–272.

[54] T. W. Moore and R. Clayton, "The impact of public information on phishing attack and defense," *Communications and Strategies*, no. 81, pp. 45–68, 2011.

[55] M. Cova, C. Kruegel, and G. Vigna, "There is no free phish: An analysis of" free" and live phishing kits." *WOOT*, vol. 8, pp. 1–8, 2008.

[56] S. Zawoad, A. K. Dutta, A. Sprague, R. Hasan, J. Britt, and G. Warner, "Phish-net: Investigating phish clusters using drop email addresses," in *eCrime Researchers Summit (eCRS), 2013*. IEEE, 2013, pp. 1–13.

[57] T. Moore and R. Clayton, "Discovering phishing dropboxes using email metadata," in *eCrime Researchers Summit (eCrime), 2012*. IEEE, 2012, pp. 1–9.

[58] W. D. Yu, S. Nargundkar, and N. Tiruthani, "A phishing vulnerability analysis of web based systems," in *Computers and Communications, 2008. ISCC 2008. IEEE Symposium on*. IEEE, 2008, pp. 326–331.

[59] C. Abad, "The economy of phishing: A survey of the operations of the phishing market," *First Monday*, vol. 10, no. 9, 2005.

[60] C. Herley and D. Florêncio, "A profitless endeavor: phishing as tragedy of the commons," in *Proceedings of the 2008 workshop on New security paradigms*. ACM, 2009, pp. 59–70.

[61] P. Soni, S. Firake, and B. Meshram, "A phishing analysis of web based systems," in *Proceedings of the 2011 International Conference on Communication, Computing & Security*. ACM, 2011, pp. 527–530.

[62] E. Bursztein, B. Benko, D. Margolis, T. Pietraszek, A. Archer, A. Aquino, A. Pitsillidis, and S. Savage, "Handcrafted fraud and extortion: Manual account hijacking in the wild," in *Proceedings of the 2014 Conference on Internet Measurement Conference*. ACM, 2014, pp. 347–358.

[63] R. Weaver and M. P. Collins, "Fishing for phishes: Applying capture-recapture methods to estimate phishing populations," in *Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit*. ACM, 2007, pp. 14–25.

[64] S. Sheng, M. Holbrook, P. Kumaraguru, L. F. Cranor, and J. Downs, "Who falls for phish?: A demographic analysis of phishing susceptibility and effectiveness of interventions," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '10. New York, NY, USA: ACM, 2010. [Online]. Available: http://doi.acm.org/10.1145/1753326.1753383 pp. 373–382.

[65] A. Vishwanath, B. Harrison, and Y. J. Ng, "Suspicion, cognition, and automaticity model of phishing susceptibility," *Communication Research*, p. 0093650215627483, 2016.

[66] A. Vishwanath, T. Herath, R. Chen, J. Wang, and H. R. Rao, "Why do people get phished? testing individual differences in phishing vulnerability within an integrated, information processing model," *Decision Support Systems*, vol. 51, no. 3, pp. 576–586, 2011.

[67] J. Wang, T. Herath, R. Chen, A. Vishwanath, and H. R. Rao, "Research article phishing susceptibility: An investigation into the processing of a targeted spear phishing email," *Professional Communication, IEEE Transactions on*, vol. 55, no. 4, pp. 345–362, 2012.

[68] R. Wright, S. Chakraborty, A. Basoglu, and K. Marett, "Where did they go right? understanding the deception in phishing communications," *Group Decision and Negotiation*, vol. 19, no. 4, pp. 391–416, 2010.

[69] R. T. Wright and K. Marett, "The influence of experiential and dispositional factors in phishing: An empirical investigation of the deceived," *Journal of Management Information Systems*, vol. 27, no. 1, pp. 273–303, 2010.

[70] P. A. Watters, "Why do users trust the wrong messages? a behavioural model of phishing," in *eCrime Researchers Summit, 2009. eCRIME'09.* IEEE, 2009, pp. 1–7.

[71] X. Dong, J. A. Clark, and J. Jacob, "Modelling user-phishing interaction," in *Human System Interactions, 2008 Conference on.* IEEE, 2008, pp. 627–632.

[72] M. K. Mount and M. R. Barrick, "The big five personality dimensions: Implications for research and practice in human resources management," *Research in personnel and human resources management*, vol. 13, no. 3, pp. 153–200, 1995.

[73] S. Frederick, "Cognitive reflection and decision making," *The Journal of Economic Perspectives*, vol. 19, no. 4, pp. 25–42, 2005.

[74] J. L. Parrish Jr, J. L. Bailey, and J. F. Courtney, "A personality based model for determining susceptibility to phishing attacks," *Little Rock: University of Arkansas*, 2009.

[75] I. Alseadoon, T. Chan, E. Foo, and J. Gonzales Nieto, "Who is more susceptible to phishing emails?: a saudi arabian study," in *ACIS 2012: Location, location, location: Proceedings of the 23rd Australasian Conference on Information Systems 2012.* ACIS, 2012, pp. 1–11.

[76] M. Pattinson, C. Jerram, K. Parsons, A. McCormac, and M. Butavicius, "Why do some people manage phishing emails better than others?" *Information Management & Computer Security*, vol. 20, no. 1, pp. 18–28, 2012.

[77] P. Kumaraguru, Y. Rhee, S. Sheng, S. Hasan, A. Acquisti, L. F. Cranor, and J. Hong, "Getting users to pay attention to anti-phishing education: evaluation of retention and transfer," in *Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit.* ACM, 2007, pp. 70–81.

[78] B. Wardman and G. Warner, "Automating phishing website identification through deep md5 matching," in *eCrime Researchers Summit, 2008.* IEEE, 2008, pp. 1–7.

[79] P. Prakash, M. Kumar, R. R. Kompella, and M. Gupta, "Phishnet: predictive blacklisting to detect phishing attacks," in *INFOCOM, 2010 Proceedings IEEE.* IEEE, 2010, pp. 1–5.

[80] G. Liu, G. Xiang, B. A. Pendleton, J. I. Hong, and W. Liu, "Smartening the crowds: computational techniques for improving human verification to fight phishing scams," in *Proceedings of the Seventh Symposium on Usable Privacy and Security*. ACM, 2011, p. 8.

[81] L. Li and M. Helenius, "Usability evaluation of anti-phishing toolbars," *Journal in Computer Virology*, vol. 3, no. 2, pp. 163–184, 2007.

[82] Y. Cao, W. Han, and Y. Le, "Anti-phishing based on automated individual white-list," in *Proceedings of the 4th ACM workshop on Digital identity management*. ACM, 2008, pp. 51–60.

[83] Y. Wang, R. Agrawal, and B.-Y. Choi, "Light weight anti-phishing with user whitelisting in a web browser," in *Region 5 Conference, 2008 IEEE*. IEEE, 2008, pp. 1–4.

[84] J. Chen and C. Guo, "Online detection and prevention of phishing attacks," in *Communications and Networking in China, 2006. ChinaCom'06. First International Conference on*. IEEE, 2006, pp. 1–7.

[85] W. D. Yu, S. Nargundkar, and N. Tiruthani, "Phishcatch-a phishing detection tool," in *Computer Software and Applications Conference, 2009. COMPSAC'09. 33rd Annual IEEE International*, vol. 2. IEEE, 2009, pp. 451–456.

[86] D. L. Cook, V. K. Gurbani, and M. Daniluk, "Phishwish: a stateless phishing filter using minimal rules," in *Financial Cryptography and Data Security*. Springer, 2008, pp. 182–186.

[87] N. Chou, R. Ledesma, Y. Teraguchi, J. C. Mitchell et al., "Client-side defense against web-based identity theft." in *NDSS*, 2004.

[88] M. G. Alkhozae and O. A. Batarfi, "Phishing websites detection based on phishing characteristics in the webpage source code," *International Journal of Information and Communication Technology Research*, vol. 1, no. 6, 2011.

[89] M. Atighetchi and P. Pal, "Attribute-based prevention of phishing attacks," in *2009 Eighth IEEE International Symposium on Network Computing and Applications*. IEEE, 2009, pp. 266–269.

[90] Y. Zhang, J. I. Hong, and L. F. Cranor, "Cantina: a content-based approach to detecting phishing web sites," in *Proceedings of the 16th international conference on World Wide Web*. ACM, 2007, pp. 639–648.

[91] G. Xiang and J. I. Hong, "A hybrid phish detection approach by identity discovery and keywords retrieval," in *Proceedings of the 18th international conference on World wide web*. ACM, 2009, pp. 571–580.

[92] M. Dunlop, S. Groat, and D. Shelly, "Goldphish: Using images for content-based phishing analysis," in *Internet Monitoring and Protection (ICIMP), 2010 Fifth International Conference on.* IEEE, 2010, pp. 123–128.

[93] M. Sharifi and S. H. Siadati, "A phishing sites blacklist generator," in *Computer Systems and Applications, 2008. AICCSA 2008. IEEE/ACS International Conference on.* IEEE, 2008, pp. 840–843.

[94] C. Ardi and J. Heidemann, "Auntietuna: Personalized content-based phishing detection," 2016.

[95] S. Afroz and R. Greenstadt, "Phishzoo: Detecting phishing websites by looking at them," in *Semantic Computing (ICSC), 2011 Fifth IEEE International Conference on.* IEEE, 2011, pp. 368–375.

[96] B. Wardman, T. Stallings, G. Warner, and A. Skjellum, "High-performance content-based phishing attack detection," in *eCrime Researchers Summit (eCrime), 2011.* IEEE, 2011, pp. 1–9.

[97] G. Liu, B. Qiu, and L. Wenyin, "Automatic detection of phishing target from phishing webpage," in *Pattern Recognition (ICPR), 2010 20th International Conference on.* IEEE, 2010, pp. 4153–4156.

[98] G. Xiang, B. A. Pendleton, J. Hong, and C. P. Rose, "A hierarchical adaptive probabilistic approach for zero hour phish detection," in *Computer Security–ESORICS 2010.* Springer, 2010, pp. 268–285.

[99] A. P. Rosiello, E. Kirda, C. Kruegel, and F. Ferrandi, "A layout-similarity-based approach for detecting phishing pages," in *Security and Privacy in Communications Networks and the Workshops, 2007. SecureComm 2007. Third International Conference on.* IEEE, 2007, pp. 454–463.

[100] L. Wenyin, N. Fang, X. Quan, B. Qiu, and G. Liu, "Discovering phishing target based on semantic link network," *Future Generation Computer Systems*, vol. 26, no. 3, pp. 381–388, 2010.

[101] L. Wenyin, G. Huang, L. Xiaoyue, Z. Min, and X. Deng, "Detection of phishing webpages based on visual similarity," in *Special interest tracks and posters of the 14th international conference on World Wide Web.* ACM, 2005, pp. 1060–1061.

[102] L. Wenyin, G. Huang, L. Xiaoyue, X. Deng, and Z. Min, "Phishing web page detection," in *Document Analysis and Recognition, 2005. Proceedings. Eighth International Conference on.* IEEE, 2005, pp. 560–564.

[103] W. Liu, X. Deng, G. Huang, and A. Y. Fu, "An antiphishing strategy based on visual similarity assessment," *IEEE Internet Computing*, vol. 10, no. 2, p. 58, 2006.

[104] K.-T. Chen, J.-Y. Chen, C.-R. Huang, and J.-Y. Chen, "Fighting phishing with discriminative keypoint features," *Internet Computing, IEEE*, vol. 13, no. 3, pp. 56–63, 2009.

[105] M. Hara, A. Yamada, and Y. Miyake, "Visual similarity-based phishing detection without victim site information," in *Computational Intelligence in Cyber Security, 2009. CICS'09. IEEE Symposium on*. IEEE, 2009, pp. 30–36.

[106] A. Y. Fu, L. Wenyin, and X. Deng, "Detecting phishing web pages with visual similarity assessment based on earth mover's distance (emd)," *Dependable and Secure Computing, IEEE Transactions on*, vol. 3, no. 4, pp. 301–311, 2006.

[107] P. Mensah, G. Blanc, K. Okada, D. Miyamoto, and Y. Kadobayashi, "Ajna: Anti-phishing js-based visual analysis, to mitigate users excessive trust in ssl/tls."

[108] E. Medvet, E. Kirda, and C. Kruegel, "Visual-similarity-based phishing detection," in *Proceedings of the 4th international conference on Security and privacy in communication netowrks*. ACM, 2008, p. 22.

[109] T.-C. Chen, S. Dick, and J. Miller, "Detecting visually similar web pages: Application to phishing detection," *ACM Transactions on Internet Technology (TOIT)*, vol. 10, no. 2, p. 5, 2010.

[110] G.-G. Geng, X.-D. Lee, W. Wang, and S.-S. Tseng, "Favicon-a clue to phishing sites detection," in *eCrime Researchers Summit (eCRS), 2013*. IEEE, 2013, pp. 1–10.

[111] M. Chandrasekaran, K. Narayanan, and S. Upadhyaya, "Phishing email detection based on structural properties," in *NYS Cyber Security Conference*, 2006, pp. 1–7.

[112] S. Abu-Nimeh, D. Nappa, X. Wang, and S. Nair, "A comparison of machine learning techniques for phishing detection," in *Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit*. ACM, 2007, pp. 60–69.

[113] A. Saberi, M. Vahidi, and B. M. Bidgoli, "Learn to detect phishing scams using learning and ensemble? methods," in *Proceedings of the 2007 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Workshops*. IEEE Computer Society, 2007, pp. 311–314.

[114] I. Fette, N. Sadeh, and A. Tomasic, "Learning to detect phishing emails," in *Proceedings of the 16th International Conference on World Wide Web*, ser. WWW '07. New York, NY, USA: ACM, 2007. [Online]. Available: http://doi.acm.org/10.1145/1242572.1242660 pp. 649–656.

[115] R. Basnet, S. Mukkamala, and A. H. Sung, "Detection of phishing attacks: A machine learning approach," in *Soft Computing Applications in Industry*. Springer, 2008, pp. 373–383.

[116] A. Bergholz, J. H. Chang, G. Paass, F. Reichartz, and S. Strobel, "Improved phishing detection using model-based features." in *CEAS*, 2008.

[117] A. Bergholz, J. De Beer, S. Glahn, M.-F. Moens, G. Paaß, and S. Strobel, "New filtering approaches for phishing email," *Journal of computer security*, vol. 18, no. 1, pp. 7–35, 2010.

[118] A. Bergholz, G. Paaß, L. DAddona, and D. Dato, "A real-life study in phishing detection," in *Proceedings of the Conference on Email and Anti-Spam (CEAS)*, vol. 1, 2010, pp. 1–10.

[119] L. Ma, J. Yearwood, and P. Watters, "Establishing phishing provenance using orthographic features," in *eCrime Researchers Summit, 2009. eCRIME'09.* IEEE, 2009, pp. 1–10.

[120] G. L'Huillier, R. Weber, and N. Figueroa, "Online phishing classification using adversarial data mining and signaling games," in *Proceedings of the ACM SIGKDD Workshop on CyberSecurity and Intelligence Informatics*. ACM, 2009, pp. 33–42.

[121] V. Ramanathan and H. Wechsler, "phishgillnetphishing detection methodology using probabilistic latent semantic analysis, adaboost, and co-training," *EURASIP Journal on Information Security*, vol. 2012, no. 1, pp. 1–22, 2012.

[122] C. K. Olivo, A. O. Santin, and L. S. Oliveira, "Obtaining the threat model for e-mail phishing," *Applied Soft Computing*, vol. 13, no. 12, pp. 4841–4848, 2013.

[123] P. Dewan, A. Kashyap, and P. Kumaraguru, "Analyzing social and stylometric features to identify spear phishing emails," in *Electronic Crime Research (eCrime), 2014 APWG Symposium on*. IEEE, 2014, pp. 1–13.

[124] S. Smadi, N. Aslam, L. Zhang, R. Alasem, and M. Hossain, "Detection of phishing emails using data mining algorithms," in *2015 9th International Conference on Software, Knowledge, Information Management and Applications (SKIMA)*. IEEE, 2015, pp. 1–8.

[125] R. Verma and N. Rai, "Phish-idetector: Message-id based automatic phishing detection."

[126] S. Garera, N. Provos, M. Chew, and A. D. Rubin, "A framework for detection and measurement of phishing attacks," in *Proceedings of the 2007 ACM workshop on Recurring malcode*. ACM, 2007, pp. 1–8.

[127] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Identifying suspicious urls: an application of large-scale online learning," in *Proceedings of the 26th annual international conference on machine learning*. ACM, 2009, pp. 681–688.

[128] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond blacklists: learning to detect malicious web sites from suspicious urls," in *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2009, pp. 1245–1254.

[129] V. Bulakh and M. Gupta, "Countering phishing from brands' vantage point," in *Proceedings of the 2016 ACM on International Workshop on Security And Privacy Analytics*. ACM, 2016, pp. 17–24.

[130] Y. Pan and X. Ding, "Anomaly based web phishing page detection," in *null*. IEEE, 2006, pp. 381–392.

[131] D. Miyamoto, H. Hazeyama, and Y. Kadobayashi, "An evaluation of machine learning-based methods for detection of phishing sites," in *Advances in Neuro-Information Processing*. Springer, 2008, pp. 539–546.

[132] M. Aburrous, M. A. Hossain, K. Dahal, and F. Thabtah, "Intelligent phishing detection system for e-banking using fuzzy data mining," *Expert systems with applications*, vol. 37, no. 12, pp. 7913–7921, 2010.

[133] M. Aburrous, M. A. Hossain, F. Thabatah, and K. Dahal, "Intelligent phishing website detection system using fuzzy techniques," in *Information and Communication Technologies: From Theory to Applications, 2008. ICTTA 2008. 3rd International Conference on*. IEEE, 2008, pp. 1–6.

[134] M. Aburrous, M. A. Hossain, K. Dahal, and F. Thabtah, "Predicting phishing websites using classification mining techniques with experimental case studies," in *Information Technology: New Generations (ITNG), 2010 Seventh International Conference on*. IEEE, 2010, pp. 176–181.

[135] C. Whittaker, B. Ryner, and M. Nazif, "Large-scale automatic classification of phishing pages." in *NDSS*, vol. 10, 2010.

[136] M. He, S.-J. Horng, P. Fan, M. K. Khan, R.-S. Run, J.-L. Lai, R.-J. Chen, and A. Sutanto, "An efficient phishing webpage detector," *Expert Systems with Applications*, vol. 38, no. 10, pp. 12 018–12 027, 2011.

[137] G. Xiang, J. Hong, C. P. Rose, and L. Cranor, "Cantina+: A feature-rich machine learning framework for detecting phishing web sites," *ACM Transactions on Information and System Security (TISSEC)*, vol. 14, no. 2, p. 21, 2011.

[138] H. Zhang, G. Liu, T. W. Chow, and W. Liu, "Textual and visual content-based anti-phishing: a bayesian approach," *Neural Networks, IEEE Transactions on*, vol. 22, no. 10, pp. 1532–1546, 2011.

[139] G. Bottazzi, E. Casalicchio, D. Cingolani, F. Marturana, and M. Piu, "Mp-shield: A framework for phishing detection in mobile devices," in *Computer and Information Technology; Ubiquitous Computing and Communications; Dependable, Autonomic and Secure Computing; Pervasive Intelligence and Computing (CIT/IUCC/DASC/PICOM), 2015 IEEE International Conference on*. IEEE, 2015, pp. 1977–1983.

[140] Z. Dong, A. Kapadia, J. Blythe, and L. J. Camp, "Beyond the lock icon: real-time detection of phishing websites using public key certificates," in *Electronic Crime Research (eCrime), 2015 APWG Symposium on*. IEEE, 2015, pp. 1–12.

[141] H. H. Nguyen and D. T. Nguyen, "Machine learning based phishing web sites detection," in *AETA 2015: Recent Advances in Electrical Engineering and Related Sciences*. Springer, 2016, pp. 123–131.

[142] Giovane C. M. Moura, Moritz Muller, Maarten Wullink, and Cristian Hesselman, "nDEWS: a New Domains Early Warning System for TLDs," in *IEEE/IFIP International Workshop on Analytics for Network and Service Management (AnNet 2016), co-located with IEEE/IFIP Network Operations and Management Symposium (NOMS 2016)*, April 2016.

[143] A. Aggarwal, A. Rajadesingan, and P. Kumaraguru, "Phishari: Automatic realtime phishing detection on twitter," in *eCrime Researchers Summit (eCrime), 2012*. IEEE, 2012, pp. 1–12.

[144] C. V. Zhou, C. Leckie, S. Karunasekera, and T. Peng, "A self-healing, self-protecting collaborative intrusion detection architecture to trace-back fast-flux phishing domains," in *Network Operations and Management Symposium Workshops, 2008. NOMS Workshops 2008. IEEE*. IEEE, 2008, pp. 321–327.

[145] C. V. Zhou, C. Leckie, and S. Karunasekera, "Collaborative detection of fast flux phishing domains," *Journal of Networks*, vol. 4, no. 1, pp. 75–84, 2009.

[146] D. K. McGrath, A. Kalafut, and M. Gupta, "Phishing infrastructure fluxes all the way," *IEEE Security & Privacy*, no. 5, pp. 21–28, 2009.

[147] T. Li, F. Han, S. Ding, and Z. Chen, "Larx: Large-scale anti-phishing by retrospective data-exploring based on a cloud computing platform," in *Computer Communications and Networks (ICCCN), 2011 Proceedings of 20th International Conference on*. IEEE, 2011, pp. 1–5.

[148] F. Maggi, A. Sisto, and S. Zanero, "A social-engineering-centric data collection initiative to study phishing," in *Proceedings of the First Workshop on Building Analysis Datasets and Gathering Experience Returns for Security*. ACM, 2011, pp. 107–108.

[149] C. Yue and H. Wang, "Anti-phishing in offense and defense," in *Computer Security Applications Conference, 2008. ACSAC 2008. Annual*. IEEE, 2008, pp. 345–354.

[150] C. Yue and H. Wang, "Bogusbiter: A transparent protection against phishing attacks," *ACM Transactions on Internet Technology (TOIT)*, vol. 10, no. 2, p. 6, 2010.

[151] M. Chandrasekaran, R. Chinchani, and S. Upadhyaya, "Phoney: Mimicking user response to detect phishing attacks," in *Proceedings of the 2006 International Symposium on on World of Wireless, Mobile and Multimedia Networks*. IEEE Computer Society, 2006, pp. 668–672.

[152] Y. Joshi, S. Saklikar, D. Das, and S. Saha, "Phishguard: a browser plug-in for protection from phishing," in *Internet Multimedia Services Architecture and Applications, 2008. IMSAA 2008. 2nd International Conference on*. IEEE, 2008, pp. 1–6.

[153] S. Li and R. Schmitz, *A novel anti-phishing framework based on honeypots*. IEEE, 2009.

[154] P. Knickerbocker, D. Yu, and J. Li, "Humboldt: A distributed phishing disruption system," in *eCrime Researchers Summit, 2009. eCRIME'09.* IEEE, 2009, pp. 1–12.

[155] H. Shahriar and M. Zulkernine, "Trustworthiness testing of phishing websites: A behavior model-based approach," *Future Generation Computer Systems*, vol. 28, no. 8, pp. 1258–1271, 2012.

[156] C. M. McRae and R. B. Vaughn, "Phighting the phisher: Using web bugs and honeytokens to investigate the source of phishing attacks," in *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on.* IEEE, 2007, pp. 270c–270c.

[157] E. Kirda and C. Kruegel, "Protecting users against phishing attacks with antiphish," in *Computer Software and Applications Conference, 2005. COMPSAC 2005. 29th Annual International*, vol. 1. IEEE, 2005, pp. 517–524.

[158] M. Wu, R. C. Miller, and G. Little, "Web wallet: preventing phishing attacks by revealing user intentions," in *Proceedings of the second symposium on Usable privacy and security.* ACM, 2006, pp. 102–113.

[159] K.-P. Yee and K. Sitaker, "Passpet: convenient password management and phishing protection," in *Proceedings of the second symposium on Usable privacy and security.* ACM, 2006, pp. 32–43.

[160] D. Florêncio and C. Herley, "Evaluating a trial deployment of password re-use for phishing prevention," in *Proceedings of the anti-phishing working groups 2nd annual eCrime researchers summit.* ACM, 2007, pp. 26–36.

[161] S. Bin, W. Qiaoyan, and L. Xiaoying, "A dns based anti-phishing approach," in *Networks Security Wireless Communications and Trusted Computing (NSWCTC), 2010 Second International Conference on*, vol. 2. IEEE, 2010, pp. 262–265.

[162] A. Herzberg and A. Gbara, "Trustbar: Protecting (even naive) web users from spoofing and phishing attacks," Cryptology ePrint Archive, Report 2004/155. http://eprint. iacr. org/2004/155, Tech. Rep., 2004.

[163] A. Herzberg and A. Jbara, "Security and identification indicators for browsers against spoofing and phishing attacks," *ACM Transactions on Internet Technology (TOIT)*, vol. 8, no. 4, p. 16, 2008.

[164] R. Dhamija and J. D. Tygar, "Phish and hips: Human interactive proofs to detect phishing attacks," in *Human Interactive Proofs.* Springer, 2005, pp. 127–141.

[165] R. Dhamija and J. D. Tygar, "The battle against phishing: Dynamic security skins," in *Proceedings of the 2005 Symposium on Usable Privacy and Security*, ser. SOUPS '05. New York, NY, USA: ACM, 2005. [Online]. Available: http://doi.acm.org/10.1145/1073001.1073009 pp. 77–88.

[166] M. Topkara, A. Kamra, M. J. Atallah, and C. Nita-Rotaru, "Viwid: Visible watermarking based defense against phishing," in *Digital Watermarking*. Springer, 2005, pp. 470–483.

[167] B. Parno, C. Kuo, and A. Perrig, *Phoolproof phishing prevention*. Springer, 2006.

[168] M. G. Gouda, A. X. Liu, L. M. Leung, and M. A. Alam, "Spp: An anti-phishing single password protocol," *Computer Networks*, vol. 51, no. 13, pp. 3715–3726, 2007.

[169] H. Tout and W. Hafner, "Phishpin: An identity-based anti-phishing approach," in *Computational Science and Engineering, 2009. CSE'09. International Conference on*, vol. 3. IEEE, 2009, pp. 347–352.

[170] Y. Oiwa, H. Takagi, H. Watanabe, and H. Suzuki, "Pake-based mutual http authentication for preventing phishing attacks," in *Proceedings of the 18th international conference on World wide web*. ACM, 2009, pp. 1143–1144.

[171] J. Crain, L. Opyrchal, and A. Prakash, "Fighting phishing with trusted email," in *Availability, Reliability, and Security, 2010. ARES'10 International Conference on*. IEEE, 2010, pp. 462–467.

[172] M. Jakobsson and H. Siadati, "Spoofkiller: You can teach people how to pay, but not how to pay attention," in *Socio-Technical Aspects in Security and Trust (STAST), 2012 Workshop on*. IEEE, 2012, pp. 3–10.

[173] A. A. Khan, "Preventing phishing attacks using one time password and user machine identification," *arXiv preprint arXiv:1305.2704*, 2013.

[174] N. Costigan, "The growing pain of phishing: is biometrics the cure?" *Biometric Technology Today*, vol. 2016, no. 2, pp. 8–11, 2016.

[175] C. Ludl, S. McAllister, E. Kirda, and C. Kruegel, "On the effectiveness of techniques to detect phishing sites," in *DIMVA*, vol. 7. Springer, 2007, pp. 20–39.

[176] T. Moore and R. Clayton, "Evaluating the wisdom of crowds in assessing phishing websites," in *Financial Cryptography and Data Security*. Springer, 2008, pp. 16–30.

[177] T. Moore and R. Clayton, "The consequence of non-cooperation in the fight against phishing," in *eCrime Researchers Summit, 2008*. IEEE, 2008, pp. 1–14.

[178] S. Sheng, B. Wardman, G. Warner, L. F. Cranor, J. Hong, and C. Zhang, "An empirical analysis of phishing blacklists," 2009.

[179] C. Jackson, D. R. Simon, D. S. Tan, and A. Barth, "An evaluation of extended validation and picture-in-picture phishing attacks," in *Financial Cryptography and Data Security*. Springer, 2007, pp. 281–293.

[180] S. E. Schechter, R. Dhamija, A. Ozment, and I. Fischer, "The emperor's new security indicators," in *Security and Privacy, 2007. SP'07. IEEE Symposium on*. IEEE, 2007, pp. 51–65.

[181] S. Egelman, L. F. Cranor, and J. Hong, "You've been warned: an empirical study of the effectiveness of web browser phishing warnings," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2008, pp. 1065–1074.

[182] J. Sunshine, S. Egelman, H. Almuhimedi, N. Atri, and L. F. Cranor, "Crying wolf: An empirical study of ssl warning effectiveness." in *USENIX Security Symposium*, 2009, pp. 399–416.

[183] E. Lin, S. Greenberg, E. Trotter, D. Ma, and J. Aycock, "Does domain highlighting help people identify phishing sites?" in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*. ACM, 2011, pp. 2075–2084.

[184] M. Wu, R. C. Miller, and S. L. Garfinkel, "Do security toolbars actually prevent phishing attacks?" in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '06. New York, NY, USA: ACM, 2006. [Online]. Available: http://doi.acm.org/10.1145/1124772.1124863 pp. 601–610.

[185] Y. Zhang, S. Egelman, L. Cranor, and J. Hong, "Phinding phish: Evaluating anti-phishing tools." ISOC, 2006.

[186] A. Darwish and E. Bataineh, "Eye tracking analysis of browser security indicators," in *Computer Systems and Industrial Informatics (ICCSII), 2012 International Conference on*. IEEE, 2012, pp. 1–6.

[187] T. Whalen and K. M. Inkpen, "Gathering evidence: use of visual security cues in web browsers," in *Proceedings of Graphics Interface 2005*. Canadian Human-Computer Communications Society, 2005, pp. 137–144.

[188] A. Alnajim and M. Munro, "An evaluation of users tips effectiveness for phishing websites detection," in *Digital Information Management, 2008. ICDIM 2008. Third International Conference on*. IEEE, 2008, pp. 63–68.

[189] I. Kirlappos and M. A. Sasse, "Security education against phishing: A modest proposal for a major rethink," *IEEE Security & Privacy*, no. 2, pp. 24–32, 2011.

[190] C. Herley, "So long, and no thanks for the externalities: the rational rejection of security advice by users," in *Proceedings of the 2009 workshop on New security paradigms workshop*. ACM, 2009, pp. 133–144.

[191] P. Kumaraguru, J. Cranshaw, A. Acquisti, L. Cranor, J. Hong, M. A. Blair, and T. Pham, "School of phish: A real-world evaluation of anti-phishing training," in *Proceedings of the 5th Symposium on Usable Privacy and Security*, ser. SOUPS '09. New York, NY, USA: ACM, 2009. [Online]. Available: http://doi.acm.org/10.1145/1572532.1572536 pp. 3:1–3:12.

[192] A. Alnajim and M. Munro, "An anti-phishing approach that uses training intervention for phishing websites detection," in *Information Technology: New Generations, 2009. ITNG'09. Sixth International Conference on*. IEEE, 2009, pp. 405–410.

[193] S. Gupta and P. Kumaraguru, "Emerging phishing trends and effectiveness of the anti-phishing landing page," in *Electronic Crime Research (eCrime), 2014 APWG Symposium on*. IEEE, 2014, pp. 36–47.

[194] D. Caputo, S. Pfleeger, J. Freeman, and M. Johnson, "Going spear phishing: Exploring embedded training and awareness," *Security Privacy, IEEE*, vol. 12, no. 1, pp. 28–38, Jan 2014.

[195] K. Jansson and R. von Solms, "Phishing for phishing awareness," *Behaviour & Information Technology*, vol. 32, no. 6, pp. 584–593, 2013.

[196] V. Anandpara, A. Dingman, M. Jakobsson, D. Liu, and H. Roinestad, "Phishing iq tests measure fear, not ability," in *Financial Cryptography and Data Security*. Springer, 2007, pp. 362–366.

[197] S. A. Robila and J. W. Ragucci, "Don't be a phish: steps in user education," in *ACM SIGCSE Bulletin*, vol. 38, no. 3. ACM, 2006, pp. 237–241.

[198] L. A. Werner and J. Courte, "Analysis of an anti-phishing lab activity," *Information systems Education Journal*, vol. 8, no. 11, 2010.

[199] E. Bekkering, D. Hutchison, and L. Werner, "A follow-up study of detecting phishing emails," in *Proceedings of the Conference on Information Systems Applied Research. Washington DC*, 2009.

[200] S.-S. Tseng, K.-Y. Chen, T.-J. Lee, and J.-F. Weng, "Automatic content generation for anti-phishing education game," in *Electrical and Control Engineering (ICECE), 2011 International Conference on*. IEEE, 2011, pp. 6390–6394.

[201] S. Srikwan and M. Jakobsson, "Using cartoons to teach internet security," *Cryptologia*, vol. 32, no. 2, pp. 137–154, 2008.

[202] P. Kumaraguru, Y. Rhee, A. Acquisti, L. F. Cranor, J. Hong, and E. Nunge, "Protecting people from phishing: The design and evaluation of an embedded training email system," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems*, ser. CHI '07. New York, NY, USA: ACM, 2007. [Online]. Available: http://doi.acm.org/10.1145/1240624.1240760 pp. 905–914.

[203] S. Sheng, B. Magnien, P. Kumaraguru, A. Acquisti, L. F. Cranor, J. Hong, and E. Nunge, "Anti-phishing phil: the design and evaluation of a game that teaches people not to fall for phish," in *Proceedings of the 3rd symposium on Usable privacy and security*. ACM, 2007, pp. 88–99.

[204] N. A. G. Arachchilage, S. Love, and K. Beznosov, "Phishing threat avoidance behaviour: An empirical investigation," *Computers in Human Behavior*, vol. 60, pp. 185–197, 2016.

[205] P. Finn and M. Jakobsson, "Designing ethical phishing experiments," *Technology and Society Magazine, IEEE*, vol. 26, no. 1, pp. 46–58, Spring 2007.

[206] M. Jakobsson, P. Finn, and N. Johnson, "Why and how to perform fraud experiments," *Security & Privacy, IEEE*, vol. 6, no. 2, pp. 66–68, 2008.

[207] M. Jakobsson and J. Ratkiewicz, "Designing ethical phishing experiments: A study of (rot13) ronl query features," in *Proceedings of the 15th International Conference on World Wide Web*, ser. WWW '06. New York, NY, USA: ACM, 2006. [Online]. Available: http://doi.acm.org/10.1145/1135777.1135853 pp. 513–522.