Proactive Phishing Sites Detection

Akihito Nakamura*
University of Aizu
Aizu-Wakamatsu, Fukushima, Japan
nakamura@u-aizu.ac.jp

Fuma Dobashi†
Infosec Corporation
Minatoku, Tokyo, Japan
fuma.dobashi@infosec.co.jp

ABSTRACT

Phishing is one of the social engineering techniques to steal users' sensitive information by disguising a fake Web site as a trustworthy one. Previous research proposed phishing mitigation techniques, such as blacklist, heuristics, visual similarity, and machine learning. However, these kinds of methods have limitation on the detection of a zero-hour phishing site, a phishing site that no one has noticed yet.

This paper presents a new approach to the detection of zero-hour phishing sites: proactive detection. If those malicious sites are detected as early as possible, shutdown by the specialized agencies and mitigation of user damages are expected. We also present a method and system of efficient phishing site detection based on the proactive approach. The method is composed of two major parts: suspicious domain names generation and judgment. The former predicts likely phishing Web sites from the given legitimate brand domain name. The latter scores and judges suspects by calculating various indexes. That is, zero-hour phishing sites can be detected by hypothesis and test cycles. As a result of the preliminary experiment, we detected several zero-hour phishing sites disguising as major brands, including eBay, Google, and Amazon.

CCS CONCEPTS

• **Security and privacy** → **Phishing**; *Social network security and privacy*.

KEYWORDS

phishing, social engineering, proactive detection, heuristics

ACM Reference Format:
Akihito Nakamura and Fuma Dobashi. 2019. Proactive Phishing Sites Detection. In *IEEE/WIC/ACM International Conference on Web Intelligence (WI '19), October 14–17, 2019, Thessaloniki, Greece.* ACM, New York, NY, USA, 6 pages. https://doi.org/10.1145/3350546.3352565

1 INTRODUCTION

Phishing attack is one of the social engineering techniques used to deceive users. The main purpose of this attack is to steal users' sensitive information typically at a fake Web site [13]. Examples include user ID, password, credit card number, and phone number. The average number of the detected phishing sites is about 50 thousand per month in 2018 [6].

Typical steps to a phishing attack are as follows. First, the attacker builds a phishing Web site in imitation of the target legitimate site. Then, the attacker sends phishing e-mails which contain a link to the site with falsifying the sender, e.g. legitimate financial institute. Next, users receive the phishing e-mails and some of them click the link. This link is faked that it does not connect to the legitimate site, but to the phishing site. The displayed page requires users to enter personal information. Finally, the input information, such as credit card numbers, is sent to the attacker. After some incidents are found out and reported, the specialized agencies alert users to the phishing site and try to shut down the site.

There are two major classes of countermeasures against phishing: user training approach and software approach. The former has limitations and it is difficult to be an effective method. People who have conducted phishing discrimination education have detected 28% of false negative [14]. The latter helps human to make better judgment by alerting or blocking phishing messages and sites.

Previous research proposed software-based phishing detection techniques. They include blacklist [5, 15], heuristics [1, 16], visual similarity [7, 20], and machine learning [11, 18]. These approaches are *reactive*, i.e. a phishing site is detected after some users accessed that site and then judged using these methods. *Zero-hour phishing* remains an open issue. A zero-hour phishing site is an unknown phishing site that no one has noticed yet.

In this paper, we present a new counter approach to the phishing attack: *proactive detection of zero-hour phishing sites*. Our method first makes a prediction about likely phishing domain names from a given brand's one, then accesses those sites, evaluates those authenticity, and finally reports the judgment for suspects. In other words, we make a specific prediction about likely phishing domain names and test them. That is, phishing sites are detected by *hypothesis and test cycles* in a proactive way. As a result, it becomes detectable zero-hour phishing sites to shut down as early as possible. Early countermeasures by the specialized agencies and mitigation of user damage are expected. At the hypothesis phase, likely phishing domain names are generated by an algorithm based on the prior occurrences maintained in PhishTank [13], a public phishing information database. At the test phase, we combine various techniques to increase the accuracy of judgement.

The remainder of this paper is organized as follows. Section 2 describes our approach to phishing detection and method. In section 3, we briefly describe our software implementation. We show a result of the preliminary experiment and evaluation of the proposed method in section 4. Section 5 describes the related work and section 6 concludes the paper.

WI '19, October 14–17, 2019, Thessaloniki, Greece

ACM ISBN 978-1-4503-6934-3/19/10…$15.00
https://doi.org/10.1145/3350546.3352565

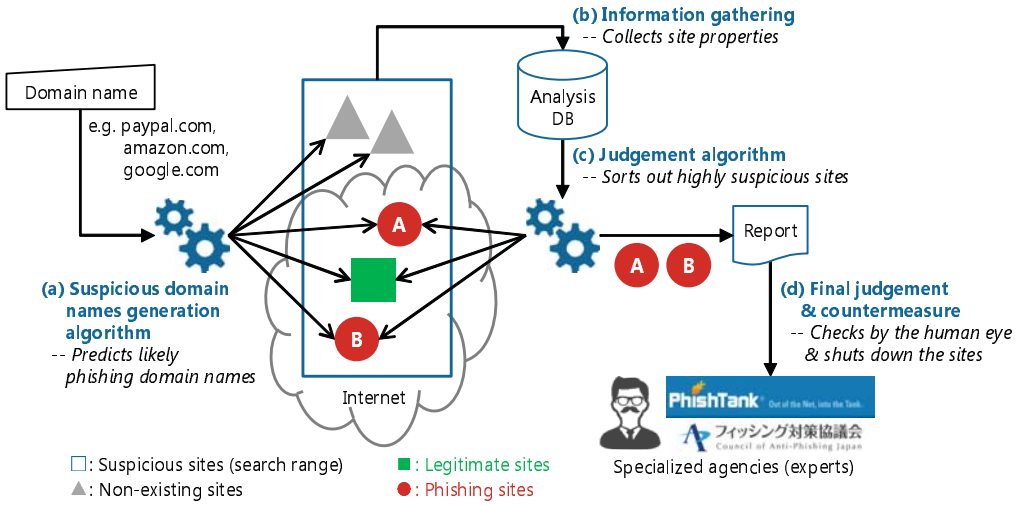

Figure 1: Proactive Phishing Detection Method

2 PROACTIVE PHISHING DETECTION METHOD

In this section, we describe our method to detect phishing sites in a proactive way. Figure 1 represents the primary components and the execution steps. Our method is composed of two phases: *hypothesis and test.*

In the hypothesis phase, the likely domain names of phishing sites are generated from a given brand domain name, e.g. PayPal.com (Figure 1 (a)). These domain names are candidate for the phishing sites. For each candidate, the existence of the Web site is checked. If it exists, the properties of the site and the Web pages are collected (Figure 1 (b)).

In the test phase, the existing candidate sites are evaluated and scored by calculating various indexes (Figure 1 (c)). The higher the score is, the more likely the site is phishing. Then, the highly suspicious sites are sorted out and reported to the responsible person to check by the human eye. Finally, the most likely phishing sites are reported to the specialized agencies as appropriate (Figure 1 (d)).

2.1 Suspicious Domain Names Generation

The *suspicious domain names generation algorithm* generates likely phishing domain names from a given legitimate domain name (Figure 1 (a)). The generated domain names are suspects to be investigated. This algorithm consists of two techniques: *brand name deformation and top-level domain replacement.*

2.1.1 Brand Name Deformation. First, we focus on the *brand name* part of a domain name. A brand name appears at the second-level domain in case of a generic top-level domain (TLD) or at the third-level in case of a country code TLD. For example, amazon is the brand name part of the domain names amazon.com and amazon.co.uk.

For a given brand domain name, the algorithm generates multiple *deformed brand names* based on the 14 rules shown in Table 1. The deformation rules are derived by analyzing the prior occurrences of phishing domain names registered with PhishTank [13]. That is, at the moment, this algorithm is designed by *heuristics*. We analyzed 7,131 domain names of 20 brands which are the top 14 most frequently appeared phishing sites in PhishTank and other six popular brands. Among them, 696 domain names could be generated using these deformation rules. B0 and B1 cover about 38%, respectively. That is, totally about 76% of these domain names can be generated using these two rules.

For example, let us consider the domain name paypal.com. The brand name is paypal. According to B1, deformed brand names pypl.com, pay.com, and ppl.com are generated by omitting some characters.

With B3 and B6, some characters in a brand name are replaced and added using *look-alike character* sets shown in Table 2, respectively. A look-alike character set is a set of characters whose elements are look-alike each other. That is, the characters in each set are indistinguishable to the human eye at a glance.

For example, three brand names, poypal, paypai, and pavpal are generated from the given brand name paypal by replacing characters in the same sets C1 ('a' with 'o'), C3 ('l' with 'i'), and C4 ('y' with 'v'), respectively.

This kind of character resemblance problem or domain spoofing attack is referred to as *homograph attack* [3]. At the moment, we have not taken into account the internationalized domain name homograph attack [17] using the non-Latin letters, i.e. the Universal Character Set (Unicode) and the Internationalized Resource Identifier (IRI) [2],

2.1.2 Top-Level Domain Replacement. Another technique to generate suspicious domain names is *top-level domain (TLD) replacement.* A TLD is one of the domains at the highest level in the hierarchical domain name system. For example, .com and .net are generic TLDs. Also, .br and .ru are country code TLDs used for Brazil and Russia,

Table 1: Brand Name Deformation Rules

ID	Definition	Examples (of paypal)	PhishTank Coverage
B0	Using the brand name as it is	paypal	38%
B1	Omitting characters	pypl, pay, ppl	38%
B2	Swapping characters	palpay	0%
B3	Replacing characters with look-alike ones †	poypal, paypai, pavpal	4%
B4	Replacing characters with random ones	payxal, kaypal	1%
B5	Duplicating characters	ppaypal, paypall	1%
B6	Adding look-alike characters †	pavypal, paypail	1%
B7	Adding overlook-prone characters †	parypal, piaypal	1%
B8	Adding random characters	paypalx, pakypal	1%
B9	Adding a hyphen	pay-pal, pa-ypal	1%
B10	Replacing a part with a service-related word	paymentpal	1%
B11	Replacing a part with a non-service-related word	payhop, jobpal	1%
B12	Replacing a part and omitting a part	payment	4%
B13	Combining rules	pavypcl (B3 and B6)	11%

† See Table 2.

Table 2: Character Sets: Look-Alike (C1 – C12) and Overlook-Prone (C13)

ID	Character set
C1	{ a, e, c, o }
C2	{ b, d, cl, k, h, 9 }
C3	{ I (uppercase i), 1 (one), l (lowercase L), i, j, t }
C4	{ v, w, y, u }
C5	{ g, q, p, o }
C6	{ t, f }
C7	{ n, m }
C8	{ h, ln }
C9	{ b, lo }
C10	{ d, ol }
C11	{ w, vv }
C12	{ o, 0 (zero) }
C13	{ l (lowercase L), i, r, t }

Table 3: Replacement Top-Level Domains (TLDs)

ID	TLD
T1	.com
T2	.net
T3	.org
T4	.br (Brazil)
T5	.ru (Russia)
T6	.info
T7	.au (Australia)
T8	.in (India)
T9	.es (Spain)
T10	.uk (United Kingdom)
T11	.biz

respectively. A *second-level domain* (SLD) is a domain that is directly below a TLD.

Replacements of TLDs, and together with SLDs in some cases, are often used to create phishing domain names. We incorporate the top 11 TLDs with the most frequent entries registered with PhishTank (Table 3). For example, domain names amazon.com.br and ebay.ru are generated from amazon.com and ebay.com by replacing the TLD .com to .com.br and .ru , respectively.

The algorithm combines the brand name deformation and TLD replacement to generate likely phishing domain names. The brand name deformation rule B0 is a pseudo one for cases in which only the TLD replacement is used.

2.2 Phishing Judgment

In the next phase, we make a judgment on whether each Web site of the generated suspicious domain name is phishing or not. Each site is evaluated and scored using the *judgment algorithm* (Figure 1 (c)). The higher the score is, the more likely the site is phishing. For the purpose, the properties of the site are collected if it exists (Figure 1 (b)).

This algorithm takes the 17 indexes into account to calculate the score (Table 4). We formulate three indexes J1–J3 and adopt the others from the previous work [7, 20]. J0 is a special index to choose zero-hour phishing sites.

A suspicious score S is a real number in the range of 0–100 given by Equation 1, where $I(J_i)$ is a value of index J_i, $max(I(J_i))$ is the maximum value, and w_i is the weight value for J_i in a real interval $[0, 1]$. The algorithm attaches more weight to more important rules.

$$S = \left\{ \sum_{i=1}^{15} w_i I(J_i) + \sum_{i=16}^{17} \frac{w_i I(J_i)}{max(I(J_i))} \right\} \times \frac{100}{\sum_i^{17} w_i} \quad (1)$$

We also provide qualitative suspicious *rankings* to deliver a simple message to human administrators; *high* for $[80, 100]$, *medium* for $[60, 80)$, and *low* for [0,60). Finally, administrators checks the reported sites by the human eye (Figure 1 (d)).

Table 4: Judgment Scoring Indexes

ID	Category†	Definition	Value
J0	–	Domain name is not registered with PhishTank	0 (false) or 1 (true)
J1	W	Brand name is included in HTML text	0 or 1
J2	D	Domain name was registered after the legitimate one	0 or 1
J3	U	URL scheme is http (not https)	0 or 1
J4	D	Domain age is early (less than 12 months)	0 or 1
J5	D	Country code in WHOIS entry is different from the legitimate one	0 or 1
J6	D	TTL value of the DNS A records is short (less than 1800 seconds)	0 or 1
J7	W	HTML text contains form elements	0 or 1
J8	W	HTML text contains JavaScript code	0 or 1
J9	W	HTML text contains external links (not the same origin)	0 or 1
J10	W	JavaScript code contains long strings	0 or 1
J11	U	URL contains '-'	0 or 1
J12‡	U	Host name in URL is IP address	0 or 1
J13‡	U	URL contains many '.' (dots) (more than five)	0 or 1
J14‡	U	URL contains many digits (more than five)	0 or 1
J15‡	U	URI contains ".exe"	0 or 1
J16	W	HTML textual similarity	real number [0, 1]
J17	W	HTML visual similarity	real number [0, 1]

† W: Web contents (HTML and JavaScript), U: URL, D: Domain name (DNS and WHOIS)
‡ These indexes are used only if the HTTP request is redirected to another page with a different URL.

3 IMPLEMENTATION

We built a software to demonstrate the proposed method. The software is written in Python, about 1,200 lines of code, and SQLite is used for the analysis database.

3.1 Suspicious Domain Names Generation and Information Gathering

At first, the system receives a domain name of the target brand and generates a list of suspicious domain names using the domain names generation algorithm. Then, for each domain name in the list, it collects the following ever-changing information and stores in the analysis database for later analysis.

- PhishTank entry
- WHOIS information
- DNS information
- HTTP response information and Web page screenshot

If a suspicious domain name is found in PhishTank, it is already judged and is not a zero-hour phishing site. WHOIS is a protocol used for querying databases that store the registered users of an Internet resource, such as a domain name and an IP address block. The indexes J2, J4, and J5 are checked using WHOIS. The HTML text contains important information to judge: the indexes J1 and J7–J10. The HTML text and Web page screenshot of the legitimate brand site are gathered beforehand for later comparison with suspicious sites: J16 and J17.

In case of a non-existing site, an HTTP request is eventually timed out. However, PhishTank entry and WHOIS information may exist because the site was shut down before or it will be built in future.

3.2 Judgment and Reporting

The system calculates the suspicious score for each existing site using the indexes. The values of the indexes and the suspicious score are stored in the database. Finally, the system creates a report to notify the judgment result, particularly the high and medium suspects to the human administrator.

For the indexes J16 and J17, i.e. textual and visual similarities between the suspicious site and the legitimate site, we are under consideration of algorithms, such as Histogram, ORB, and AKAZE [7, 18, 20]. The visual image of HTML is created by using Selenium [4] and Firefox [12].

4 EXPERIMENTAL RESULT

In this section, we show the preliminary experimental result. The experiment was conducted from December 1, 2018 to December 18, 2018 for 12 brands which are the top most popular phishing targets registered with PhishTank. In the experiment, we simplified the method: only the domain name deformation rules B1–B4 and the TLD replacement were used to generate suspicious domain names, and the judgment scoring indexes J1 and J2 were used.

4.1 System Execution Result

Table 5 shows the result in descending order of the number of phishing sites registered with PhishTank on December 15, 2018. The system generated 11,718 suspicious domain names for 12 brands using the domain names generation algorithm. Among them, 2,069 (17.7%) sites existed. Then, the system calculated the suspicious scores of them, and finally selected and reported 206 (1.8%) sites as phishing.

Table 5: Experimental Result (The TLD .com is common among the brands.)

Brand	PhishTank entries	Suspects (generated by system)	Existing sites		Phishing (judged by system)		Phishing (judged by human)
Facebook	1,338	480	112	(23.3%)	29	(6.0%)	0
PayPal	1,320	379	125	(33.0%)	20	(5.3%)	0
MicrosoftOnline	387	4,591	611	(13.3%)	2	(0.04%)	0
eBay	257	216	91	(42.1%)	11	(5.1%)	1 (ebey.ru)
MyEtherWallet	151	2,631	371	(14.1%)	6	(0.2%)	1 (myetherwallet.ru) †
Binance	111	371	60	(16.2%)	7	(1.9%)	1 (binamce.ru)
Google	82	333	105	(31.5%)	53	(15.9%)	2 (goegle.net, goggle.com.br)
Dropbox	46	502	91	(18.1%)	9	(1.8%)	0
Amazon	34	245	91	(37.1%)	34	(13.9%)	1 (amozon.com.br)
Yahoo	20	302	76	(25.2%)	12	(4.0%)	0
BankOfAmerica	18	1,198	239	(19.9%)	2	(0.2%)	0
LinkedIn	18	470	97	(20.6%)	21	(4.5%)	0
total	3,782	11,718	2,069	(17.7%)	206	(1.8%)	6 (0.05%)

† As it turned out, the site is not for phishing.

4.2 Accuracy

After the execution of the method by the system, we checked the selected sites by the human eye and decided six sites as highly suspicious zero-hour phishing sites. These sites are shown in Table 6. We can see the domain names generation rule B3, replacing characters with look-alike ones, is effective. In addition, all these sites are matched by the scoring index J1: brand name is included in HTML text. Also, there are high visual similarities.

We reported this result to the specialized agencies: PhishTank and Council of Anti-Phishing Japan. Finally, two sites were judged as phishing: ebey.ru and goegle.net. The HTML images of these two sites are shown in Figure 2. They look just like the legitimate sites, respectively. Other three sites were closed before the final judgment. Each of them also looked exactly like the original site. As it turned out, one site, myetherwallet.ru, is not for phishing. Since the lifespan of phishing sites is very short, we believe that the three closed sites were very likely phishing. As a result, it is safe to say that we could detect five zero-hour phishing sites disguising as eBay, Binance, Google, and Amazon.

The upshot is that five among 206 sites were certainly phishing sites; the accuracy rate is 2.4%. Let us remember that these phishing sites are *zero-hour* ones. That is, no one has noticed them yet until our system detected.

Table 6: Detected Zero-Hour Phishing Sites

Brand	Domain name	Domain names generation rule	Scoring index
eBay	ebey.ru	B3, TLD	J1
MyEtherWallet	myetherwallet.ru †	B0, TLD	J1
Binance	binamce.ru	B3, TLD	J1
Google	goegle.net	B3, TLD	J1
Google	goggle.com.br	B3, TLD	J1
Amazon	amazon.com.br	B3, TLD	J1

† As it turned out, the site is not for phishing.

5 RELATED WORK

There are roughly two approaches to phishing detection: user training and software. The former approach has limitations against cunning tricks and difficult to be an effective method [14]. Here, we introduce software methods and compare them with our proposed one. There are four kinds of software methods: *blacklist, heuristics, image similarity, and machine learning* [9]. We incorporate some of the proposed techniques into our judgment scoring indexes except machine learning.

A blacklist is a list of URLs, IP addresses, and/or DNS information of known phishing sites. This kind of method checks if a list contains target sites. An example implementation is the Google Safe Browsing API v4 [5]. This API is used by the major Web browsers, including Chrome, Safari, and Firefox. The advantage of the blacklist method is low false positive rate. However, it takes time to propagate: as 47%–83% of phishing sites appeared on blacklists 12 hours from the initial test [15].

Heuristics methods focus on features of phishing sites. For example, there are many dots in the URL, the URL scheme is not https but http [1], the page rank is low [16], and the Web page includes illegal links [10]. Another method uses the TF-IDF (term frequency/inverse document frequency) algorithm for the Web page text [19]. The disadvantage is that the false positive rate is generally higher than blacklists.

Visual similarity methods are based on the content presentation, i.e. what users see on Web browsers. As a result of comparing the Web page image of a suspicious site with the legitimate one, if the similarity is high, it is judged as phishing [7, 20]. The characteristics of this kind of method are inherent in phishing detection. That is, it is potentially more accurate than other types of method. The disadvantage is the need for adequate computation resources.

(a) ebey.ru

(b) goegle.net

Figure 2: Proactively detected zero-hour phishing sites

As recent advances of machine learning algorithms and systems, they are utilized for phishing detection [11, 18]. A classification model is created by extracting features from the datasets of whitelists and blacklists. An example implementation is CheckPhish [8]. Enormous and carefully selected datasets are indispensable for this type of method.

6 CONCLUSION

In this paper, we discussed how to detect zero-hour phishing sites in a proactive way. The proposed method is composed of the suspicious domain names generation and the judgment algorithms. The former makes a specific prediction about likely phishing domain names and the latter tests them. That is, unknown phishing sites are detected by hypothesis and test cycles without receiving phishing mails or clicking malicious URLs. As a result, it becomes detectable zero-hour phishing sites to shut down as early as possible. Early countermeasures by specialized agencies and mitigation of user damage are expected.

We built a system and conducted preliminary experiment to evaluate the proposed method. Consequently, we could successfully detected several zero-hour phishing sites disguising as major brands like eBay, Google, and Amazon. Therefore, the method is proved to be effective in achieving the goal at a certain level.

In future work, the method will be improved with the introduction of collaborative filtering technique. The suspicious domain name generation algorithm is currently based on heuristics. We are aiming to make automatic predictions about phishing domain names of a brand by collecting the prior occurrences, e.g. from PhishTank.

REFERENCES

[1] Debra L Cook, Vijay K Gurbani, and Michael Daniluk. 2008. Phishwish: a stateless phishing filter using minimal rules. In *International Conference on Financial Cryptography and Data Security*. Springer, 182–186.
[2] M. Duerst and M. Suignard. 2005. Internationalized Resource Identifiers (IRIs), RFC 3987. https://www.rfc-editor.org/rfc/rfc3987.txt
[3] Evgeniy Gabrilovich and Alex Gontmakher. 2002. The Homograph Attack. *CACM* 45, 2 (feb 2002), 128. https://doi.org/10.1145/503124.503156
[4] GitHub. 2019. Python bindings for Selenium. Retrieved May 20, 2019 from https://github.com/SeleniumHQ/selenium/
[5] Google. 2019. Safe Browsing APIs (v4). Retrieved May 20, 2019 from https://developers.google.com/safe-browsing/v4/
[6] Anti-Phishing Working Group. 2018. Phishing Activity Trends Reports. Retrieved May 20, 2019 from https://www.antiphishing.org/resources/apwg-reports/
[7] Masanori Hara, Akira Yamada, and Yutaka Miyake. 2009. Visual similarity-based phishing detection without victim site information. In *2009 IEEE Symposium on Computational Intelligence in Cyber Security*. IEEE, 30–36.
[8] RedMarlin Inc. 2019. CheckPhish: AI Powered Real-time Phishing Detection. Retrieved May 20, 2019 from https://checkphish.ai/
[9] Mahmoud Khonji, Youssef Iraqi, and Andrew Jones. 2013. Phishing detection: a literature survey. *IEEE Communications Surveys & Tutorials* 15, 4 (2013), 2091–2121. https://doi.org/10.1109/SURV.2013.032213.00009
[10] Nakahiro Kitayama and Takashi Imaizumi. 2015. Phishing site detection based on links in a web page. In *Proceddings of the Internet and Operation Technology Symposium 2015*. 22. http://id.nii.ac.jp/1001/00145976/
[11] Gang Liu, Bite Qiu, and Liu Wenyin. 2010. Automatic detection of phishing target from phishing webpage. In *2010 20th International Conference on Pattern Recognition*. IEEE, 4153–4156.
[12] Mozilla. 2019. Firefox. Retrieved May 20, 2019 from https://www.mozilla.org/ja/firefox/
[13] OpenDNS. 2019. PhishTank. Retrieved May 20, 2019 from https://www.phishtank.com/
[14] Steve Sheng, Mandy Holbrook, Ponnurangam Kumaraguru, Lorrie Faith Cranor, and Julie Downs. 2010. 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 (CHI '10)*. ACM, New York, NY, USA, 373–382. https://doi.org/10.1145/1753326.1753383
[15] Steve Sheng, Brad Wardman, Gary Warner, Lorrie Cranor, Jason Hong, and Chengshan Zhang. 2009. An Empirical Analysis of Phishing Blacklists. (jul 2009). https://doi.org/10.1184/R1/6469805.v1
[16] A Naga Venkata Sunil and Anjali Sardana. 2012. A PageRank based detection technique for phishing web sites. In *2012 IEEE Symposium on Computers & Informatics (ISCI)*. IEEE, 58–63.
[17] Wikipedia. 2019. IDN homograph attack. Retrieved May 20, 2019 from https://en.wikipedia.org/wiki/IDN_homograph_attack
[18] Haijun Zhang, Gang Liu, Tommy WS Chow, and Wenyin Liu. 2011. Textual and visual content-based anti-phishing: a Bayesian approach. *IEEE Transactions on Neural Networks* 22, 10 (2011), 1532–1546.
[19] Yue Zhang, Jason I. Hong, and Lorrie F. Cranor. 2007. Cantina: A Content-based Approach to Detecting Phishing Web Sites. In *Proceedings of the 16th International Conference on World Wide Web (WWW '07)*. ACM, New York, NY, USA, 639–648. https://doi.org/10.1145/1242572.1242659
[20] Yu Zhou, Yongzheng Zhang, Jun Xiao, Yipeng Wang, and Weiyao Lin. 2014. Visual similarity based anti-phishing with the combination of local and global features. In *2014 IEEE 13th International Conference on Trust, Security and Privacy in Computing and Communications*. IEEE, 189–196.