Internationalized Domain Name Homograph Attacks

CSE 227: Computer Security - Spring 2017
University of California San Diego

Chen Lai[*] chl588@ucsd.edu
Zhongrong Jian[*] zhjian@ucsd.edu
J. Sidrach[*] jsidrach@ucsd.edu

Abstract

TODO

1 Introduction

Domain names were originally designed to only support ASCII characters. Internationalized Domain Names (IDNs) were proposed back in December 1996 by Martin Dürst[1] for the purpose of letting non-English speaking people use Internet without additional restrictions. This extension involves representing Unicode characters in ASCII using Punycode, so that they could be then rendered back into their Unicode representation. Homograph letters (different letters whose representation is almost, if not, the same), however, present a potential security vulnerability. For example, Cyrillic letter "a" can look identical to Latin letter "a" depending on the font. In other languages, like Chinese, there exists many homograph letters between traditional Chinese and simplified Chinese. Malicious attackers could then register a domain where one of the letters is actually Cyrillic but whose representation matched a Latin one. Users could be linked to this newly registered malicious page, and they may not have any visual indication (at least without further interaction) that the page is not the one they think they are visiting.

In this paper, we analyze the potential impact of this type of attack, commonly refered to as IDN Homograph Attack. Section 2 explains past and present policies of major browsers and top-level domain (TLD) registrants in an attempt to prevent the attack. Section 3 explores relevant work in the literature related to homograph attacks. Section 4 describes the methodology of data collection and analysis adopted in this project. Section 5 comments the different results obtained. Section 6 addresses the ethical concerns regarding the data collection. Section 7 presents the conclusions obtained from this project.

2 Background

TODO - Browsers policy - TLDs policy

Currently, most popular browsers will show the hostname in Unicode depends on the language setting to defend IDN attack. For example, Google Chrome[2] will display IDN in Unicode only if all characters of the domain names belong to only one language and this language has to be the user's preferred language. 1 show the screenshots of the original url of `www.google.com` and its homograph.

2.1 Browsers Policy

The major browsers have their own IDN policies to defend against IDN homograph attack. Google Chrome, starting with Chrome 51, will display punycode if certain tests fail on the input url. These

[*] Author names are ordered lexicographically, by last name.

[1] `https://tools.ietf.org/html/draft-duerst-dns-i18n-00`

[2] `https://www.google.com/chrome`

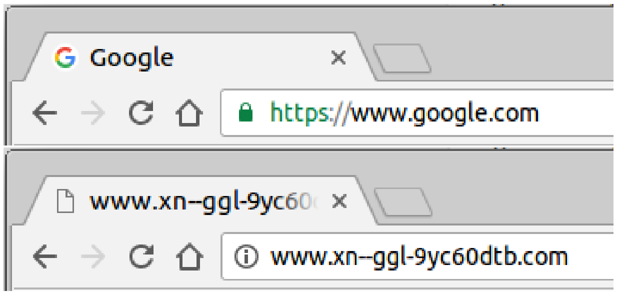

Figure 1: Google(top) and its homograph(bottom) on Chrome

tests include but not limited to converting to unicode, less than one numbering systems involved, no invisible characters, etc. Internet Explorer[3] displays urls in IDN form if every component contains only characters from the languages configured in *Internet Options*. Firefox[4] uses a script mixing detection algorithm based on *”Moderately Restrictive” profile of Unicode Technical Report 39* and displays unicode when url consists of any single script, any single script plus Latin, or it's in the whitelist of other combinations. Opera[5] shows IDN only for whitelisted TLDs. Safari[6] has a whitelist of scripts that do not contain confusable characters and only shows the IDN form for the whitelisted scripts.

2.2 TLDs Policy

3 Related Work

TODO - Brief analysis of previous papers on the same topic

Many relevant studies have been conducted to show the vulnerability of IDN and provide possible solution to mitigate IDN homograph attack.

In 2001, Evgeniy Gabrilovich and Alex Gontmakher has proved the feasibility of such attack based on the vulnerability of Unicode in *The Homograph Attack* [2].

In 2006, Vicktor Krammer proposed a defense mechanism based on address bar highlighting/indication technique and better UI of the browsers [3]. His method of defense mainly target at regular users who do not neccessarily have relevant knowledge and through the better rendering of address along with explicit notification or alert on phishing url users would be aware of malicious website. This method check each character in an url seperately, highlighting digits or characters of various scripts in different background color so that users will be aware of the difference. It also suggest using larger font size in the address bar to help distinguish possible homograph characters from the Latin letter and alerting users when multiscript is detected by means of scripts block[7]. His proposal was innovative in his year but not entirely accurate because characters from the same script may be in different blocks while characters from different scripts sometimes lie in the same block.

Manurer and Höfer believed that safe guard based on browsers' blacklist is far from enough to protect users from sophisticated phishing attacks [1]. They proposed an advanced method based on url components and spell checking functionality of search engine. In this method, an url would be divided into several subterms, namely, base name[8], subdomain[9], path domain[10], brand name[11]. Their algorithm is based on the following characteristics on different subterms of url:

- Base name: Phisher cannot register the original domain but can only register mispelling or homograph domain name because the original one is often registered ahead by the company.
- Subdomain: Phisher can prepend subdomain to the primary domain being attacked to fool users into thinking that they are on the real domain. One example would be

[3] https://support.microsoft.com/en-us/help/17621/internet-explorer-downloads

[4] https://www.mozilla.org/en-US/firefox/new/

[5] http://www.opera.com

[6] https://www.apple.com/safari

[7] http://www.unicode.org/reports/tr24/

[8] The primary domain name as registered at the registrar

[9] A part of the primary domain

[10] A subfolder of the url

[11] A brand name that occurs in a url

http://us.battle.net.loginaccoutbattle.net/login/en/login.html.

- Path domain: Phisher sometimes place a second domain as a subfolder of the url path, usually right after the primary domain name, for instance, http://piasel.altervista.org/www.paypal.com/new/paypal/intl/update.
- Brand name: There are cases that only the brand name is inserted somewhere in the URL to fool users.

Each of this components will be sent to search engine as a term for spelling check and in the cases that spelling suggestion is returned the url will be counted as suspicious. By splitting a url into these subterms, this method cover almost all possible cases of fake url or domain name. However, this mechanism is based on the search engine for return spelling suggestion and as a result, it does not perform well when it is used as sole detection.

4 Methodology

Our data collection involved two primary sources. The first one was a snapshot of the `.com` and `.net` domain zone, from now on referenced as *.com snapshot*. This snapshot was provided by Verisign[12], and it is dated on 2017/05/01. It contains, for the most part, the name servers of all `.com` and `.net` domains. For the purposes of this project, only the `.com` domains were considered.

The second data source was the *Alexa Top 1 million sites ranking*[13], from now on referenced as *top domains snapshot*. It contains the most popular one million web sites (as ranked by Alexa), regardless whether they are `.com` domains or not. It was also retrieved on 2017/05/01.

In an effort to help with the reproducibility and replicability of this project, the original data and processing code used in this project has been made available in a public repository[14].

4.1 Data Processing

We used the *.com snapshot* to obtain all the internationalized domains of the `.com` domain zone. All internationalized domain names are represented using Punycode, so they start with the `xn--` prefix. Using this information we first filtered the *.com snapshot* to match only the `.com` domains that start with `xn--`. Since the *.com snapshot* contains name server records, a single domain may have more than one entry in the file. Only the domain name is relevant for this project, so the rest of the columns of the *.com snapshot* were discarded. We sorted the domains in lexicographical order, removing duplicates in the process.

We used the *top domains snapshot* as a source for the canonical (non-international) domain name of a website, based on its ranking. As we were only interested in non-internationalized domain names, we discarded the ones that start with the prefix `xn--`. We also removed sub-domains, since we only had top-level domain names in the *.com snapshot*.

4.2 Clustering

The underlying assumptions behind the clustering process are that homograph attacks are more likely to target popular domains, and that the million domains from the *top domains snapshot* contains most of the popular domains. To detect potential internationalized domain name homograph attacks, we clustered the internationalized domain names in the *.com snapshot*. The representative of each cluster is a homograph domain name from the *top domains snapshot*, which we called canonical domain name. Only clusters with at least one internationalized homograph domain name were output.

The detection of homograph domain names was done using an algorithm to check if two strings are confusable. This algorithm is described in the *Unicode Technical Standard #39*[15]. The approach followed has some caveats, that could be addressed in future work. For instance, it does not detect a homograph of `www.google.com`, where the first "." has

[12] https://www.verisign.com/en_US/channel-resources/domain-registry-products/zone-file/index.xhtml

[13] https://www.alexa.com/topsites

[14] https://github.com/jsidrach/idn-homograph-attack

[15] http://www.unicode.org/reports/tr39/#Confusable_Detection

been replaced by a similar looking Unicode character. More sophisticated homographs can also be generated by using Unicode characters similar to "/". It is also worth noting that only domain names from the `.com` zone were considered, but similar studies could be done to other top-level domains following the same procedure. All things considered, we still think this approach is a valuable first step.

Another type of clustering was also performed. We grouped the homograph domain names by its registrant organization, and ranked each organization by the number of homograph domain names to their name. This could shed light on which registrant organizations are allowing homograph domain names, or even on which individuals and companies are doing it the most.

4.3 Manual Classification

The last part of the data processing was the classification of the homograph domain names. This classification was done manually, to be able to differentiate between scam and unrelated websites. To help with the classification and speed up the process, we queried the *WHOIS* records[16] of every homograph domain name. Some domains had expired at the time of the classification, and were subsequently deleted from the output file.

The two high level categories that were defined are *Canonical* and *Third Party*. The first one, *Canonical*, was employed when the domain is registered by the same organization as its canonical homograph domain name. The second one, *Third Party*, was used when the domain is registered by a different organization than its canonical homograph domain name. Additionally, a more detailed classification was made:

- *Canonical - Parking*: domain was registered but it was not accessible via HTTP.
- *Canonical - Redirect*: domain redirected (HTTP Status Code 301/307/308) to its canonical homograph domain name.
- *Third Party - Redirect to Canonical:* domain redirected (HTTP Status Code 301/307/308) to its canonical homograph domain name.
- *Third Party - Unrelated*: domain resolved, but the contents of the website were totally unrelated to its canonical homograph domain name.
- *Third Party - Parking*: domain was registered, but it was not accessible via HTTP, or when accessed, a default domain parking web page was displayed.
- *Third Party - Scam*: domain resolved, and the website displayed was a clear attempt (similar color, logos, etc.) to make users think they were visiting the canonical homograph domain name.

5 Results

We obtained 458731 domain names from the *top domains snapshot* that belonged to the `.com` zone. From the *.com snapshot*, we identified a total of 1044301 internationalized domains. However, only a low percentage of the internationalized domain names (3.68%) had a matching homograph canonical domain name in the *top domains snapshot*. Even if we take into account that our homograph detection algorithm is rather limited (as described before), these results still indicate that the majority of internationalized domain names are not being used in homograph attacks. An overview of these clustering results can be found in Table 1.

Table 2 shows the number of internationalized domain name homographs for the top ten `.com` domains. It is worth noting that most of the homograph domains of *google.com* are actually registered by Google, possibly as a defensive measure. Another surprising result of the homograph clustering is that 81.87% of the groups have just one homograph internationalized domain name. This suggests that this kind of attack may not be targeted only to top websites, but rather to any domain where valuable information could be obtained from the users by a scam.

The manual classification sheds some light on how these homograph internationalized domain names are currently being used. For the most

[16] `https://whois.icann.org`

Domains	#	%
Canonical domain names	*458731*	*8.31%*
With IDN homographs	825	6.04%
Without IDN homographs	457906	2.27%
Internationalized Domain Names	*1045400*	*91.69%*
With canonical homograph	1099	3.68%
Without canonical homograph	1044301	2.74%

Table 1: Overview of the clustering results.

Domain	# of IDN homographs
google.com	24
youtube.com	3
facebook.com	9
baidu.com	3
yahoo.com	4
reddit.com	1
qq.com	2
taobao.com	1
live.com	1
vk.com	6

Table 2: Top ten .com domains in the Alexa ranking with IDN homographs.

Status	#	%
Canonical	*88*	*8.31%*
Parking	64	6.04%
Redirect	24	2.27%
Third Party	*971*	*91.69%*
Redirect to Canonical	39	3.68%
Unrelated	29	2.74%
Parking	872	82.34%
Scam	31	2.93%

Table 3: Breakdown of the manually classified homograph IDNs.

part (82.34%), the domains are registered by a third party, and not being used actively (parking). One possible explanation for this is that the domain was bought in an attempt to be re-sold later at a higher price to the owner of the matching canonical homograph domain. A detailed breakdown of this classification can be found in Table 3.

Table 4 contains the top ten registrants with the most homograph internationalized domain names. Most of these registrants are actually domain privacy companies, so it is impossible to know if the domains belong to the same person or not. There are also several individuals with a high number of homograph internationalized domain names. It is also worth mentioning that some of these individuals seem to be targeting specific industries. For instance, a single person has registered homographs of “audi”, “citroen” and “diesel”.

6 Ethical Considerations

The domain information used in this project is publicly available, and was voluntarily provided by the registrants of the domains when they registered them. The full `.com` domain zone snapshot will not be publicly disclosed, as its access requires a paid subscription. The retrieval of the *WHOIS* information was performed only for the homograph internationalized domain names, which were about one thousand. It was performed with a significant delay between petitions (3 seconds) in order to avoid saturating the servers.

Registrant organization	Registrant email	# of homograph IDNs
Domains By Proxy, LLC	–	89
Super Privacy Service c/o Dynadot	privacy@dynadot.com	23
Domain Registries Foundation	–	22
Duong Thien	thiendv@outlook.com	18
Syngenuity Limited	manager@syngenuity.com	12
Helpnet: Brand Development & Sales	help@strongestbrands.com	12
ONUNO L.L.C.	corucas@gmail.com	11
Privacy Protection Service INC d/b/a	contact@privacyprotect.org	10
Hubertus Henz	hu_h5@yahoo.de	9
wuyu	wy65535@126.com	7

Table 4: Top ten registrants with the most homograph IDNs.

7 Conclusions

TODO - Conclusions of our work - Possible future work - TLDs

Acknowledgments

We would like to thank Louis DeKoven and Stefan Savage for their help and support throughout this project.

References

[1] *Sophisticated Phishers Make More Spelling Mistakes: Using URL Similarity against Phishing* (Heidelberg, Berlin, 2012), vol. 7672 of *Lecture Notes in Computer Science*, Springer.

[2] Gabrilovich, Evgeniy. Gontmakher, A. The homograph attack. *Communications of the ACM 45*, 2 (2002), 128.

[3] Krammer, V. Phishing defense against idn address spoofing attacks. *PST '06*, 32 (2006).