Bitsquatting: Exploiting Bit-flips for Fun, or Profit?

Nick Nikiforakis, Steven Van Acker, Wannes Meert†, Lieven Desmet, Frank Piessens, Wouter Joosen

iMinds-DistriNet / †DTAI, KU Leuven, 3001 Leuven, Belgium
firstname.lastname@cs.kuleuven.be

ABSTRACT

Over the last fifteen years, several types of attacks against domain names and the companies relying on them have been observed. The well-known cybersquatting of domain names gave way to typosquatting, the abuse of a user's mistakes when typing a URL in her browser's address bar. Recently, a new attack against domain names surfaced, namely *bitsquatting*. In bitsquatting, an attacker leverages random bit-errors occurring in the memory of commodity computers and smartphones, to redirect Internet traffic to attacker-controlled domains.

In this paper, we report on a large-scale experiment, measuring the adoption of bitsquatting by the domain-squatting community through the tracking of registrations of bitsquatting domains targeting popular web sites over a 9-month period. We show how new bitsquatting domains are registered daily and how attackers are trying to monetize their domains through the use of ads, abuse of affiliate programs and even malware installations. Lastly, given the discovered prevalence of bitsquatting, we review possible defense measures that companies, software developers and Internet Service Providers can use to protect against it.

Categories and Subject Descriptors

K.6.5 [**Security and Protection**]: Unauthorized access; H.3.5 [**Online Information Services**]: Web-based services; K.4.4 [**Electronic Commerce**]: Security

Keywords

domain name; cybersquatting; bitsquatting; affiliate abuse

1. INTRODUCTION

The Domain Name System plays a crucial role in the world wide web. It transparently converts domain names, i.e., hierarchical user-memorable strings of text, to routable, machine-friendly IP addresses. Users are instructed to trust the domain names shown in their browsers' address bars and to always consult them before divulging sensitive information, making them indicators of the familiarity and legitimacy of any given web site. As with many popular technologies, their ubiquitous nature has made them an attractive target for malicious individuals seeking to exploit users.

WWW 2013, May 13–17, 2013, Rio de Janeiro, Brazil.
ACM 978-1-4503-2035-1/13/05.

In the early days of the web, people would register domain names associated with known companies and trademarks and later sell them back to their rightful owners at a much higher price. This practice was named *cybersquatting*, and it is well-known that cybersquatting pioneers made large profits from buying domains early and selling them when the demand was high [13].

When the web grew in popularity and large companies had already bought the appropriate domains for their business, the cybersquatters based their model on a new type of squatting, namely *typosquatting*. Typosquatting is based on *type-in navigation*, which is the act of a user manually typing a URL in her browser's address bar instead of relying on a hyperlink in an existing site. In the process of typing the URL of a familiar web site, a user may accidentally mistype a character in the desired domain, e.g., `paypap.com` instead of `paypal.com`, and have her browser request the page without realizing her mistake. Typosquatters started registering such mistypes of popular authoritative domains and offered them for sale. In the mean-time, the domains were used for displaying ads (even of competing companies), and in some cases, conduct phishing and drive-by download attacks [9]. Even today, the act of typosquatting is so popular that entire companies have been formed, who offer "domain-parking services" and automate the display of relevant ads on a typosquatting domain.

Popular legitimate companies whose domains were typosquatted, in an effort to protect their customers and trademarks, buy common mistypes of their sites and redirect the visiting users back to their main authoritative domains. For instance, the domain `microspft.com` is owned by Microsoft and redirects users to `microsoft.com`. Unfortunately, this action exacerbates typosquatting since it drives typosquatters to register even more similar domains in hope that they will be able to sell them to the company for profit.

In 2011, Dinaburg presented a new type of cybersquatting which he named, *bitsquatting* [6]. In bitsquatting, a cybersquatter registers a domain name which has a character that differs for one-bit from the same character in the targeted authoritative domain. Dinaburg postulated that hardware errors could cause a random bit error, specifically a bit-flip, in the bytes of memory storing a domain name and thus route requests towards a different domain, even if the user typed the correct domain. To test this theory, Dinaburg registered 30 bitsquatting domains that were targeting popular authoritative domains, e.g., `mic2osoft.com`, a bitsquatting domain for `microsoft.com`. Over a period of eight months, Dinaburg's monitors recorded more than 52,000 requests,

originating from all types of operating systems and browsers, even the ones of hand-held gaming devices.

In this paper, we study the influence of Dinaburg's findings on the domain-squatting community. While it would certainly be interesting for researchers to independently verify Dinaburg's claims, we chose not to focus on whether bitsquatting happens but on whether cyber-squatters are *convinced* that it does. Following Dinaburg's report, we construct a crawler for bitsquatting domains which, given a list of authoritative domains, automatically computes all possible bitsquatting domains that are one-bit different from the binary representation of the original domain. For each valid bitsquatting domain, the crawler attempts to resolve its IP address and if it is successful, it then visits and records the HTML code of the bitsquatting domain's main page.

Using our crawler, we track the registration of bitsquatting domains targeting the Alexa top 500 domains for nine months, and discover ample evidence which suggest that bitsquatting is now the newest addition in the arsenal of domain-squatters. In a nine-month period, we recorded a total of 5,366 unique bitsquatting domains, showing a 46% increase from the first day of our experiment. We perform a series of automated and manual experiments on the corpus of the downloaded pages of the bitsquatting domains and discover that, while the majority of them are parked and serving ads, others are abusing affiliate programs of the authoritative sites, launching drive-by download attacks to unsuspecting visitors and attempting to trick users into installing fake antivirus programs [4] and other rogue software.

The main contributions of this paper are the following:

- We present the first large-scale analysis of bitsquatting, covering the Alexa top 500 sites over a nine-month time span
- We provide detailed statistics of the population of discovered domains and categorize the domains according to their usage and their abuse
- We review possible ways of defending against bitsquatting ranging from hardware-based solutions to damage-control and solutions based on legislation

Organization.

The rest of this paper is organized as follows. In Section 2, we briefly define bitsquatting and summarize Dinaburg's findings. In Section 3, we describe our experiment and present our methodology and results for the discovery and categorization of each discovered bitsquatting domain. In Section 4, we provide some details about bitsquatting domains clustering around specific popular web sites, followed by a discussion of possible defenses in Section 5. In Section 6, we review the related work and we conclude in Section 7.

2. BITSQUATTING

In this section, we describe how bitsquatting works and introduce the terminology used in the rest of this paper. We also summarize Dinaburg's empirical validation [6], showing the plausibility of conducting a real-life bitsquatting attack.

2.1 Definition

In July 2011, Dinaburg presented for the first time the notion of bitsquatting [6], i.e., the abuse of random bit-related errors in the memory of computers, in order to drive traffic to attacker-controlled destinations. Corruption of bits can occur due to faulty hardware, memory present in devices operating outside of the expected temperature range (like smartphones and tablets that are commonly operated outdoors) or even cosmic rays.

1^{st}	2^{nd}	3^{rd}	...	10^{th}	Domain name
1110000	1100001	1111001	...	1101101	paypal.com
1111000	...	...	...	1101101	xaypal.com
1110100	...	...	...	1101101	taypal.com
1110010	...	...	...	1101101	raypal.com
1110001	...	...	...	1101101	qaypal.com
1100000	...	...	...	1101101	0aypal.com

Table 1: All possible and domain-name compatible bit-flips on the first-character byte of `paypal.com`

While bit-errors (specifically bit-flips) are infrequent on the memory of any given machine, the total amount of RAM available to networked computers and smartphones today is substantial. Moreover, according to Dinaburg, the majority of commodity desktop PCs, laptops and smartphones do not utilize Error-Correcting Code memory (ECC RAM) which could identify and correct erroneous bit-flips. Using conservative assumptions, the researcher calculated the worldwide hourly rate of errors, in devices with non-ECC RAM to 614,400. Even though the majority of these random bit-flips will be of no use to a remote attacker, there is data in memory that could lead to exploitable scenarios. More precisely, the data that could be of use to a remote attacker, is data related to URLs and their resolved IP addresses. This data can be corrupted both at the client and the server-side as well as in-transit. Here we present a few possible scenarios:

- **Cached HTML in server memory** Whenever a web page is requested from a web server, the hardware of the remote server places the page into the server's cache so as to avoid disk I/O in subsequent identical requests. If the random bit-flip occurs in the memory that holds a URL, then the errors will be propagated to clients requesting that specific page.
- **Caches in DNS servers** When a recursive DNS server resolves an unknown domain, bit-flips that happen in the rest of the resolving infrastucture can be populated and stored in the server's cache. These errors are more disastrous than the previous case, since now, all correct requests for a domain name may receive an erroneous cached response.
- **Received HTML on the client** Similarly to web servers, a web page cached in a user's browser can be a victim of bitsquatting, if a bit-flip occurs in URLs of links and remotely-included objects, such as scripts, images, and Cascading Style Sheets.
- **Router memory** Any networking devices between a user and a server are also susceptible to random bit-errors. Thus, bit-flips can be introduced in a page by the routing infrastructure between the client and the server, both in the actual content of the packets relayed as well as the routing decisions.

In all of the above cases, an undetected bit-error in the domain name can cause a user's browser and network-utilizing

software to connect to a domain that is one-bit different from the intended, authoritative domain. An attacker who registers these bitsquatting domains, can serve ads, conduct phishing attacks, launch browser exploits or even attempt to steal the cookie-stored credentials of the intended domain in the cases where the bit-flip occured in the DNS infrastructure.

Consider the case of a random bit-error occurring on the first byte of the memory storing the authoritative domain `paypal.com`, as shown in Table 1. Several observations can be made based on this example. First, not all bit-flips result in characters that are allowed to be part of domain names. Thus, even if a bit-flip takes place in the memory holding a domain name, it may result in an invalid domain and thus not resolve to an IP address. Second, some of the bit-flips result in neighboring characters and thus could be the result of an accidental mistype, (like `0aypal.com`). At the same time, other characters are "far-away" from the original characters, essentially ruling out mistypes. We explore the overlap of bitsquatting and typosquatting in Section 3.2.3. Lastly, there is always a chance that the bit-flip will result in a legitimate domain, belonging to another party. In our example with `paypal.com`, `raypal.com` is the home page of "Ray Palla", a radio-broadcaster.

In principle, bit-flips can also occur in memory holding IP addresses. While these errors could also divert traffic to attacker-controlled servers, the acquisition of a specific IP address is significantly more complicated than the registration of a bitsquatting domain.

2.2 Empirical validation

In order to discover whether bitsquatting is a real issue, Dinaburg registered 30 domains that were bitsquats of popular domains, such as `mic2osoft.com` (targeting Microsoft), `fbbdn.com` (targeting Facebook's content delivery network) and `do5bleclick.net` (targeting DoubleClick, Google's Ad management platform). In a period of over eight months, his bitsquatting domains received a total of 52,317 requests from 12,949 unique IP addresses with an average of 59 unique IP addresses per day. According to Dinaburg, requests were received from all over the world, by all popular operating systems and browsers, as well as smartphones and gaming consoles with networking capabilities, showing that all systems are potentially vulnerable to a bitsquatting attack. Additionally, Dinaburg found evidence of requests that were definitely not user-initiated, such as automatic update requests from "Windows Update", which could only be generated by misbehaving hardware.

Overall, his study showed that bitsquatting is a real possibility and that companies should protect themselves by e.g., pro-actively registering all bitsquatting domains in the same way as they already do with typosquatting domains [17]. As shown in later sections, attackers are convinced that bitsquatting is a new way to profit, as evidenced by the constant rise of registered bitsquatting domains since Dinaburg's presentation in 2011.

3. ANALYSIS

In this section, we first describe our methodology for gathering data about bitsquatting domains and then provide a detailed analysis of the population of the discovered bitsquatting domains. We study the overlap of bitsquatting with typosquatting and, using a combination of automatic and manual analysis, we categorize the discovered domains based on their content and purpose.

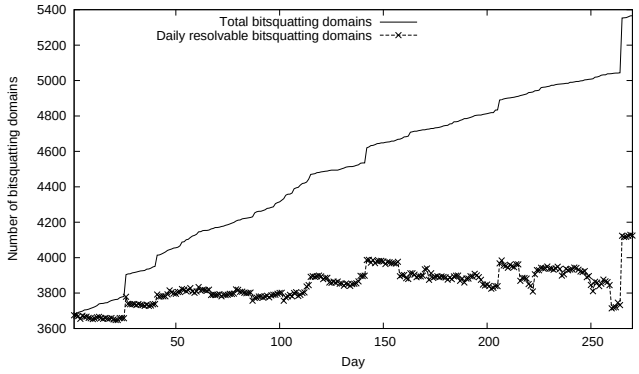

Figure 1: Daily counts of discovered bitsquatting domains

3.1 Experiment

To identify the prevalence of bitsquatting and its evolution over time, we constructed a fully automated crawler capable of discovering and recording bitsquatting domains. For each authoritative domain in a given list, the crawler first computes all possible one-bit text permutations of that domain that adhere to the allowed syntax for domain names. More precisely, a bitsquatting result is considered an allowed domain, if it only contains dots, dashes and alphanumeric characters. For every resulting bitsquatting domain, the crawler attempts to resolve the domain's IP address, and if the resolution is successful, it then requests and records the main page of the site corresponding to that domain. This process is repeated daily, in order to discover new bitsquatting domains and track the disappearing of old ones.

Our crawler was supplied with the list of the Alexa top 500 domains and allowed to execute for 270 days, starting from August 14, 2011.

3.2 Results

3.2.1 Overall growth

In the period of 270 days, we discovered a total of 5,366 different bitsquatting domains targeting 491 out of the Alexa top 500 domains. Moreover, the total number of bitsquatting domains shows a 46% increase from the starting date of our experiment. Figure 1 shows the daily growth of bitsquatting domains over that period. For any given day, the solid line represents all the bitsquatting domains found till that day. The graph shows an obviously increasing trend, which means that as days go by, more and more bitsquatting domains are purchased and made available online. The dotted-line in the same figure, shows the daily number of bitsquatting domains that were resolving to an IP address. The slope of this line is obviously smaller than the slope of the solid line. In addition, there are days where the number of resolving domains is smaller than earlier days showing that, while bitsquatting domains are registered daily, many of them are, willingly or forcefully, taken down. We believe that these domains are taken down after legal action by the authoritative domains who are being bitsquatted. Given, however, the low cost of *.com* domains, this doesn't stop

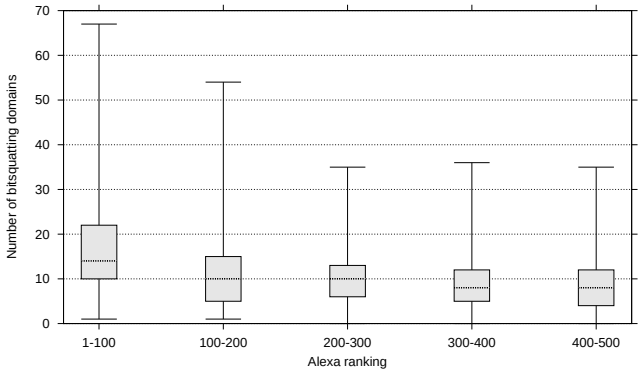

Figure 2: Number of bitsquatting domains per legitimate domain, grouped by Alexa rank

attackers from merely registering new ones, when their old domains become unavailable.

3.2.2 Targeting frequency

Figure 2 shows the number of times, each of the Alexa top 500 domains was targeted by bitsquatters. We use a "box-and-whisker" plot to map the data in quartiles. The graph can be read as follows: The edge of the lower whisker of a box, represents the minimum number of recorded bitsquatting domains for any given authoritative domain, within a specific rank, whereas the edge of the higher whisker represents the maximum number. The dotted line in each box, represents the median number of bitsquatting domains, whereas the box itself is comprised by the median numbers of the groups of data below and above the central median. For example, all sites ranking from one to one hundred were targeted by bitsquatters at least once and at most 67 times. The median number of bitsquatting attacks for all domains of that rank is 14. The small height of all gray boxes in relation to the range of their whiskers, as well as the positions of their median values show that, even though some web sites are attacked much more than the rest, the majority of sites within the Alexa top 500 receive roughly the same number of attacks. From an attacker's point of view, this can be interpreted as follows: most authoritative domains within the Alexa top 500 are equally important and thus most are targeted the same number of times.

3.2.3 Bitsquatting vs. Typosquatting

Before we explore the usage of the discovered bitsquatting domains, we want to focus on the overlap of bitsquatting with typosquatting. Given that typosquatting is known and practiced for over ten years, the question rises of why the domains discovered in our experiment were registered. Were the prospective domain-name owners registering them with typosquatting in mind, or were they considering the newly proposed bitsquatting?

A single bit-flip in a valid DNS character could be interpreted as a typo depending on the keyboard layout used. In fact, the characters resulting from most typos on any keyboard are identical with the characters resulting from a single bit-flip. Of the 38 possible characters (a-z, 0-9, dot and dash) that can be present in a valid domain-name, the binary representation of about 28[1] characters has a Hamming distance of 1 to the binary representation of another character in the valid DNS character set.

[1]28 for QWERTY and QWERTZ layouts, 27 for AZERTY

#Domains	QWERTY	AZERTY	QWERTZ	Typo-squatting?
1,301	yes	yes	yes	
6	yes	no	yes	
42	yes	no	no	Possibly yes
118	no	yes	no	
45	no	no	yes	
3,854	no	no	no	Definitely no

Table 2: Number of bitsquatting domains in the experiment that could be confused with typosquatting domains. The last line shows that 3,854 domains can not be a typo-domain according to the given keyboard layouts.

We analyzed the bitsquatting domain names in our experiment to determine whether these domains could possibly be typosquatting domains according to any of the popular keyboard-layouts, i.e., QWERTY, AZERTY or QWERTZ. We consider a domain to be a typosquatting domain when it has a "fat-finger" distance of one, from the targeted authoritative domain [14]. The results of this analysis are presented in Table 2 and show that 3,854 or 71.8% of the bitsquatting domains are not typosquatting domains. This indicates that these domains were registered specifically with bitsquatting in mind.

To further support our claim that these domains are registered with bitsquatting in mind we hypothesize that registrations for bitsquatting domains saw a sudden increase when the work of Dinaburg appeared. From the 5,366 discovered bitsquatting domains, we isolated the ones that, given a QWERTY keyboard layout, were not within a "fat-finger"' distance of one of the original domain. We limited ourselves to the QWERTY layout since, as shown in Table 2, this layout could be the most responsible for a domain being both a bitsquatting as well as a typosquatting domain. For these domains, we queried their registration dates, which we plot in Figure 3. To prove our hypothesis, we build a linear regression model describing the variation in registrations over time up to the coining of the term *bitsquatting* (no variation $p < 10^{-19}$ and $R = 0.81$). After the coining of the term, we see a sudden increase of registrations which are significantly different from the current trend ($p < 10^{-8}$) and indicates that something has abruptly changed the trend established over multiple years. Intuitively, one can see that while this type of mistyped domains were always registered, the registrations spiked in the second-half of 2011, which is when Dinaburg presented his work at BlackHat [6]. Thus it is reasonable to associate the notion of bitsquatting with the sudden increase in registrations of domains, not commonly associated with typosquatting.

3.2.4 Parked domains

Prior research by Wang et al. [17] has shown that most typosquatting domains are pointing or redirecting their traffic to domain-parking agencies. Domain-parking agencies are Internet advertising companies which specialize in the monetization of domains with no real content. The *modus operandi* of these agencies is the following: A user registers a domain name and forwards all of the received traffic to the domain-parking agency. The agency, using information

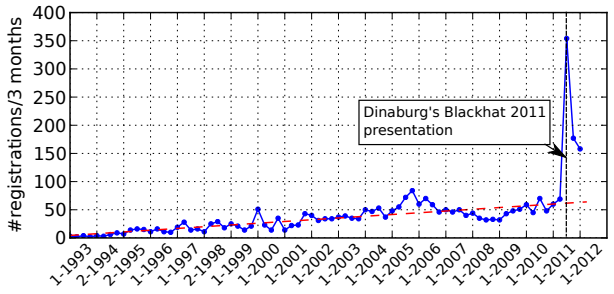

Figure 3: Registrations of bitsquatting domains that are further than a fat-finger distance of one, from the attacked domain

from both the owner of the domain, as well as the keywords present in the domain name, serves relevant ads to visiting users. Finally, the owner of the domain receives a commission for every click on the displayed ads.

Since domain-parking is prevalent among typosquatters, our hypothesis was that a similar trend would appear for bitsquatting domains. To test this hypothesis, we analyzed the data from all 5,366 bitsquatting domains for evidence of utilizing a domain-parking agency at any point during the 270 days that we were monitoring them.

The redirection of traffic from a domain name towards a domain-parking agency can be done using DNS entries, HTTP status codes, HTML META-refresh tags and client-side scripting languages, like JavaScript. In the first case, the domain owner creates a DNS record which resolves to an IP address controlled by the domain-parking agency. In the case of HTTP redirects, the domain owner needs to setup a web server that issues HTTP 301/302 status messages to forward a visiting user's browser to another website. These status messages are handled by the browser and do not render any information on the page, making the redirection transparent for the user. In the last two cases, a domain owner can setup a web server with a web page containing an HTML META-refresh header or a JavaScript-based redirection. Browsers will then render the page before being redirected to the domain-parking agency.

For redirection through DNS records, our detection method inspects the reverse DNS entry of the IP address to which the bitsquatting domain resolves. For the other three ways of redirecting traffic, our detection method inspects the hostname of the URL being redirected to. If a bitsquatting domain has a reverse DNS entry or redirects to a URL belonging to any known domain-parking agencies, it is flagged as being a "parked domain".

Our list of parking-domain agencies, shown in Table 3 comes from Wang et al. [17]. To account for less known agencies, we also use the occurrence of the word "park" as an indication that the domain is a parking domain, since the word is not frequently used in popular non-domain-parking websites. For instance, in the top 10,000 Alexa domains, there are only ten domains that use the words "park" without being domain parkers. In addition to analysis of the reverse DNS and redirected-to URLs, we also searched the downloaded HTML pages for domains that typically only occur in links embedded on domain-parking agency websites. We obtained these keywords, shown in the second row of

Domain-name-level detection
information.com, domainsponsor.com oingo.com, sedoparking.com qsrch.com, netster.com hitfarm.com
HTML-level detection
perfectnames.com, domainpool.com siliconalleydomains.com, fabulousdomains.com googlesyndication.com/apps/domainpark memorabledomains.co.uk, trafficz.com revenue.net

Table 3: Domains names utilized for the detection of domain-name parking agencies

Parking methodology	Count
Reverse DNS	1,409
HTTP 302 redirection	108
HTML META-refresh redirection	54
HTML code	1,211
Total parked domains	2,782 (51.8%)

Table 4: Parked domains discovered by each set of heuristics

Table 3, by preliminary experimentation and analysis of our bitsquatting HTML corpus.

The domain-name-level and HTML-level heuristics for the detection of domains utilizing domain-parking agencies, were used to automatically scan all 5,366 bitsquatting domains and the results are shown in Table 4. The results show that the majority of the discovered domains were indeed trying to capitalize on visiting users through the use of domain parking. At the same time, we discovered that there were some bitsquatting domains that were flagged as a parking domain by our domain-name-level heuristics but not by the HTML-level ones. Although these domains were correctly classified as parking domains, the lack of detection at the HTML-level, meant that our set of heuristics was incomplete, which in turn prompted us to take a closer look at the unclassified data (See Section 3.2.7).

3.2.5 *Self-Redirects*

As we mentioned in earlier sections, various companies, in an effort to protect their brands and customers from various cybersquatting attacks, register mistypes of their domains. Thus, when a user visits the site corresponding to a mistyped domain, the company will redirect her to the appropriate authoritative domain, usually using an HTTP 302 message. This way, the user ends up on the correct page and also sees the corrected URL (resulting from the redirect) on her browser's address bar.

From the 5,366 bitsquatting domains discovered in our experiment, we recorded 311 domains (5.7%) which redirected, for at least one day, the visiting user back to the correct authoritative domain. We manually inspected the WHOIS records of each bitsquatting domain name and compared the available information, e.g., the name and email address of technical contact and name-servers, to the information listed in the WHOIS records of the corresponding authoritative domain.

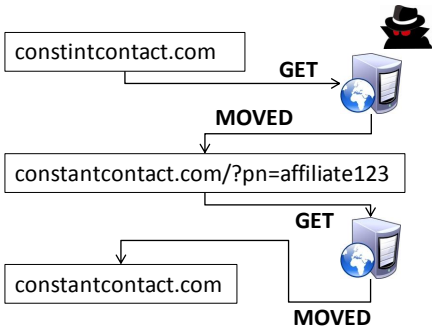

Figure 4: Abuse of bitsquatting in affiliate programs

For 211 bitsquatting domains, we were able to verify that they either belonged to the companies owning the corresponding authoritative domains or they were owned and managed by companies which specialize in brand and trademark protection. By studying the traffic generated upon the visit of the pages of those domains, we saw that the brand- and trade-protecting companies were usually first registering the fact that a user visited the specific bitsquatting domain and then redirected the user back to the appropriate authoritative domain.

From the 100 remaining bitsquatting domains which redirected the user back to the appropriate authoritative domain, we were able to verify that 58 (18.7%) were abusing affiliate programs of the authoritative domains. Affiliate programs are offered by various online companies which pay a commission to their affiliates, for every customer brought to their site, who bought their products or services. These programs usually operate with a unique affiliate identifier embedded in a link, which affiliates are expected to place on their web sites. In the case of bitsquatting, however, the attackers were using the bitsquatting domains to redirect users back to the appropriate authoritative domains with the addition of their affiliate identifiers in the new URLs. Figure 4 shows an example of actual misuse discovered in our data set. When a user requests the bitsquatting domain `constintcontact.com`, the attacker's web server redirects the user's browser to the affiliate page of `constantcontact.com` using the attacker's specific affiliate identifier (anonymized as `affiliate123`). The legitimate web server of `constantcontact.com`, registers the affiliate's identifier and redirects the user to the main page of the site. At the end of this process, the user is presented with the main page of `constantcontact.com` without knowing that she has been an unwilling part of an affiliate scheme. The authoritative domains that were targeted by bitsquatting to perform affiliate fraud, were companies offering web hosting, adult content, services, online shopping and travel-booking. Moore et al. [14] have found instances of similar abuse in typosquatting domains.

The remaining 42 bitsquatting domains were redirecting the user to the correct authoritative site and not exploiting the visitor in any obvious way. We theorize, that the owners of these domains fall in the following three categories. First, the domains may be owned by the company owning the corresponding authoritative domain which for some reason lists different details in the WHOIS records. Second, the bitsquatting domains may be registered by researchers who are attempting to recreate Dinaburg's findings. Lastly, the domains may be owned by domain-squatters who have not yet decided on the best way of monetizing their visitors, and forward the traffic back to the original site in an attempt to temporarily avoid unnecessary attention.

3.2.6 *Observed bitsquatting experiments*

The gathered data also carries evidence of ongoing bitsquatting experiments from third parties. We have recorded a total of 61 bitsquatting domains from 8 authoritative domains that announce that they are part of bitsquatting experiments. These domains were automatically discovered by searching for they keywords "bit," "squatting," and "experiment" in the HTML code of the web pages of all discovered bitsquatting domains.

These experiments are most likely conducted by researchers trying to verify Dinaburg's work [6]. We assume that in these cases, the researchers have no intent of attacking visitors, since attackers experimenting with bitsquatting would have no reason to explicitly announce their work. Examples domains are: `iozilla.org` and `wozdpress.com`

3.2.7 *Breakdown of domain usage*

Figure 5 shows a breakdown of all 5,366 bitsquatting domains in our experiment, by category. After removing 2,782 known parked domains (51.8%), 211 domains that were clearly owned by the companies owning the corresponding authoritative domains (3.9%), 112 domains that were never associated with a web server (2.1%), and 61 domains that were part of other bitsquatting experiments (1.1%), 2,200 domains (41.0%) remain that could not easily be categorized using automated means.

From these 2,200 uncategorized domains, we selected a 10% random sample for a thorough manual analysis. For each domain in the sample, we rendered its page for various days from our logs, inspected its source and whenever necessary checked its WHOIS records and its presence in Google's database of known malicious sites. Our manual inspection resulted in the following categorization:

Legitimately owned (40.0%): These were domains resolving to legitimate web sites that were either not related to the original authoritative domain or were domains with a different TLD of the same company. As demonstrated in Section 2, a random bit-flip in a domain name may result in a different legitimate domain, which is owned by a third-party who has no intent of attacking the authoritative domain, such as `androyd.com` and `raypal.com`. There were some cases, however, where the legitimate third-party web site was offering products and services of the same type as the authoritative domain, making it unclear whether it is a double coincidence or whether it is a competitor who is trying to assume the identity of the original authoritative site.

For the latter case, we discovered that some large companies, e.g. Google, own domains that when bitsquatted resolve to new domains, which still belong to the same company but under a different TLD, e.g. a bitsquat of `google.com.vn` (Google's site in Vietnam), is `google.com.tn` (Google's site in Tunisia).

Parked (15.4%): These were domains that were serving the same purpose as the ones described in Section 3.2.4 which were not discovered by our set of heuristics.

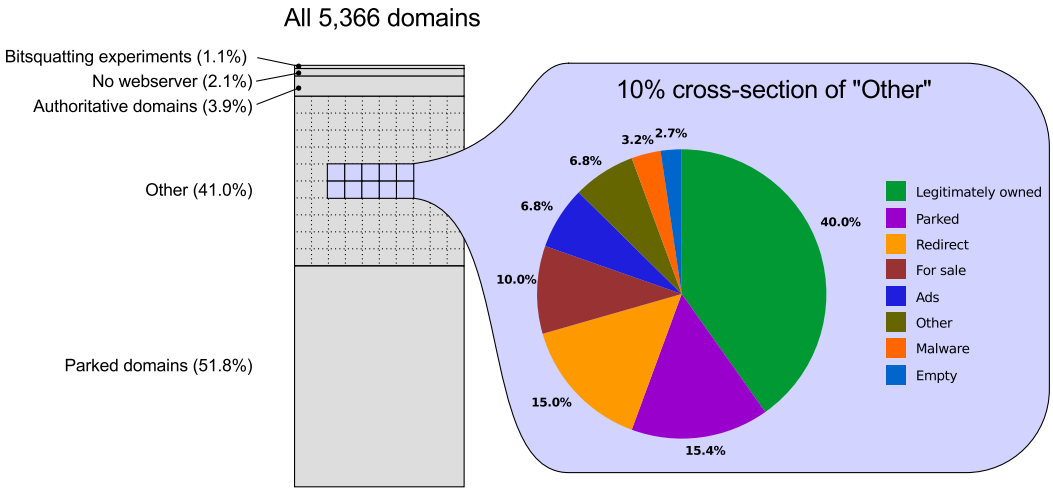

Figure 5: Analysis of bitsquatting domains by category

Among them, we discovered some special cases, such as bitsquatting domains ending in `.co.ph`. This extension belongs to the Official Domain Registry in the Philippines which resolves all non-existant domains to their own domain-registration web site.

Redirects (15.0%): In this category, the web sites were redirecting the user either to a completely different web site, e.g. that of a competing company, or were redirecting back to the authoritative domains while performing affiliate abuse, as described in Section 3.2.5.

For sale (10.0%): In these cases, the owners of domains were clearly offering their domains for sale and providing means of contacting them.

Ads (6.8%): 6.8% of the sampled domains were showing ads, but were not affiliated to a domain-parking agency. In some cases, the ads were static, specifically targetting the users of the corresponding authoritative domain, revealing the bitsquatter's intent of focusing on specific products and companies.

Search/Under Construction (6.8%) & Empty (2.7%): The domains of this category were generally providing non-useful content, being either empty, or showing an "Under Construction" message. Lastly, some of them were "fronts" for search engines, which merely forwarded a user's query to a popular search engine.

Malware (3.2%): Among the sampled domains, 3.2% of them were serving malware, either through the direct inclusion of a malicious script from a remote host or indirectly through the advertising network with which they collaborated. These script-providing hosts were automatically identified by our web browser, due to their presence in Google's Safe Browsing database.

Overall, our manual analysis, combined with the results of the previous sections leads to the following two observations: First, care must be taken when attempting to characterize a bitsquatting domain, since it may owned by a legitimate third-party. Second, the manual analysis verified that owners of bitsquatting domains are trying to capitalize on visiting users using either adverstising and for-sale listings or in some cases utilizing more intrusive approaches, such as the installation of malware. More precisely, by extrapolating the capitalizing-categories to the entire population of uncategorized results (50.4% of 2,200 uncategorized bitsquatting domains) and including the parking domains from Section 3.2.4, we can conclude that over 73% of the entire set of discovered bitsquatting domains belong to domain-squatters who attempt to profit by exploiting erroneous bit-flips.

4. CASE STUDIES

In this section, we briefly describe two instances of bitsquatting attacks, clustered around specific domains in the list of Alexa top 500 domains.

huffingtonpost.com.

"The Huffington Post", is a popular online newspaper that currently ranks in the top 100 Alexa domains. The newspaper has an unusually long domain name (14 characters excluding the suffix), which provides more bytes of characters that an attacker can squat. In fact, `huffingtonpost.com` is the host which received the maximum number of bitsquatting attacks, of the 1-100 ranking category, in Figure 2. On the 14th August 2011, when we started our experiment, `huffingtonpost.com` had 18 bitsquatting domains. This number remained the same till the 8th of September, when overnight, 49 new bitsquatting domains were registered. By manually examining these domains, we found out that for their majority, they were all providing the same page.

As Figure 6 shows, each page was alerting the user that she is there because her hardware was faulty (referring to bit-errors that were responsible for bringing the user to the bitsquatting site) and even warned the user that a malicious individual could have used this opportunity to steal the user's credentials. Subsequently, the owner of the bitsquatting domain, advised the user to buy new hardware from Amazon. More precisely, the bitsquatter was suggesting some Apple products and when clicked, the user would be redirected to Amazon with a specific affiliate identifier, so that if the user did buy a new laptop or smartphone, the bitsquatter would get a commission. Note that this attack instance is different from the affiliate abuse described

Figure 6: Bitsquatting domain for huffingtonpost.com

in Section 3.2.5, in that `huffingtonpost.com` has no affiliate program of its own and thus the attacker needs to "explain" the transition to `amazon.com`.

All 49 bitsquatting domains were available till the 29th of April 2012, giving the attacker over seven months to capitalize on visiting users. In addition, we recorded identical pages on bitsquatting domains of `zynga.com` (a popular game producer), `nytimes.com` (the New York Times) and `reddit.com`.

microsoft.com.

In the period of 270 days, we recorded a total of 40 different bitsquatting domains for `microsoft.com`. While the majority of them were parked or "for sale" domains, we also discovered more intrusive examples. `micposoft.com` is a domain which used the logo of the Microsoft Corporation and their usual blue-and-white color scheme. The site was supposedly offering multiple downloads, such as a password recovery utility, Internet Explorer 9 and Windows 7, all of which were pointing to the same executable. We downloaded the executable and submitted it to VirusTotal, an online service that scans user-submitted files against the signature databases of popular antivirus software. The executable was flagged as a "packed malware" by 2.3% of the utilized antivirus engines.

Five other domains, e.g., `microskft.com` and `microsogt.com`, were redirecting the unsuspecting user to the domain `errorfix.com`. That site was offering an "Advanced Registry Repair tool", which was flagged as a fake antivirus by 28.6% of VirusTotal's antivirus engines. Lastly, `migrosoft.com`, was offering products,training and services, abusing the similarity and name of their trademark with Microsoft's.

5. DEFENSES AGAINST BITSQUATTING

In the previous sections, we presented ample evidence showing that cybersquatters are actively bitsquatting popular Internet sites and attempt to monetize, in a variety of ways, the visits of unsuspecting users. In this section, we briefly describe some possible solutions for the protection of users and companies against bitsquatting.

5.1 Hardware-based

Bitsquatting occurs because of hardware problems either on the client-side, the server-side or any of the network infrastructure in between. The most obvious solution therefore, is to address the problem at its root. As Dinaburg suggested, data stored in hardware should include data integrity information to ensure that the data has not suffered unexpected modifications. Such data integrity validation could be accomplished by using ECC memory and CRC checks. Unfortunately, this approach will only ensure that local data is not corrupted. Routers on the network for example, will correctly store and forward any data they receive without corruption, but the corruption might already have occurred. To stop data-corruption in a networked environment, all parties must use hardware-based data integrity validation in order to be effective. Thus, even if all major ISPs and hosting companies would be willing to invest in hardware with error-correction capabilities, a complete migration would require a significant span of time.

5.2 Software-based

Another way to avoid random corruption of critical data, is to validate data integrity more frequently in the software. If the data exchanged between client and server includes data integrity information, then the data integrity can be verified at either end, ensuring that there was no corruption along the way. One option for ensuring data integrity on the DNS level is by using DNS Security Extensions (DNSSEC), which add data integrity information to DNS queries. However, as with all client-server protocols, this approach requires that both the client and the used DNS infrastructure support DNSSEC. While modern operating systems ship with built-in support for DNSSEC, the deployment of the security extensions in the DNS infrastructure is still not complete due to unforeseen obstacles [7, 16]. Another option is to use Transport Layer Security (TLS) or Secure Socket Layer (SSL), to ensure that users at least get a warning about being connected to the wrong endpoint, in case DNS traffic has been corrupted.

5.3 Incentive-removal

In Section 3.2.4, we showed that more than 50% of all registered bitsquatting domains are used to show ads, through the use of dedicated domain-parking agencies. This means, that for their majority, bitsquatters use a relatively simple, non-technical and non-intrusive approach to monetize their newly-purchased domains. Thus, even if there are thousands of individuals purchasing bitsquatting domains, they all eventually cluster to a relatively small number of domain-parking agencies. If legal control would be applied at these companies, i.e., to be forced to deny their services to domains that are obviously bitsquatting domains, then bitsquatters could no longer utilize them. It is worth pointing out that there is already legislation in-place which legally protects companies from cybersquatters and could be straightforwardly extended to cover bitsquatting [1].

If bitsquatters can no longer rely on ads, the only safe alternative for making a profit would be to sell the bitsquatting domain to the company owning the corresponding authoritative domain. While this is still an option, a collective boycott from large companies towards cybersquatters would leave them with useless non-profiting domains. Bitsquatters could of course try to monetize their domains through mal-

ware installations, but this assumes significantly more legal risk than the simple hosting of ads.

5.4 Damage-control

A more immediate way for a company to protect its trademark and users, is to accept that data corruption can occur and prevent its exploitation by rogue parties, through the pre-registration of all possible bitsquatting domains when registering the master, authoritative domain. This fix has a substantial cost overhead, as the following example shows:

The most common domain-name length among the top one million Alexa domains, is 9 characters, not counting the top-level domain (TLD); the most common top-level domain is *.com*. Consider a company wishing to register `mycompany.com`, a 9-character domain name under the *.com* TLD, and all the bitsquatting variations of this domain-name to be safe from bitsquatters. In this case, there are 42 *.com* domain names that would need to be registered, including the authoritative `mycompany.com` domain.

For some domains which are not under the *.com* top-level domain, there could be a need to register more domains under a different top-level domain authority. For instance, to register all bit-squatted variations of `mycompany.cn`, requires the registration of all domains in the *.an*, *.bn*, *.cf*, *.cl*, *.co*, *.gn*, *.kn* and *.sn* top-level domains, since a random bit-error can also occur in the TLD part of a domain name. Unfortunately, some of the resulting TLDs may be very expensive or subject to local regulations.

At the same time, Dinaburg pointed out that bitsquatting attacks can be practically exploited only against the companies owning the most popular domains, since these are the ones which get resolved the most and thus have the most chance of a random corruption. These companies are large enough to be able to afford the registration and maintenance of additional domains, especially when it comes to protecting their online identity.

6. RELATED WORK

To the best of our knowledge, our work is the first one that studies the adoption of bitsquatting by the domain-squatting community. Bitsquatting however, is only the latest instantiation of a series of attacks against the Domain Name System and the web sites relying on it. Thus, in this section we review prior domain-squatting attacks and relevant surveys.

6.1 Cybersquatting

Cybersquatting refers to the act of registering domains that are trademarks belonging to other persons and companies. Cybersquatting was popular at the dawn of the world wide web, when there were long-existing brick-and-mortar companies that did not yet have a web presence. Many opportunists registered their trademarks as domain names before them, so that they would sell the domain back to the company for profit [11]. Occasionally, the cybersquatters would host offensive or mocking content on the cybersquatting domains so that they would force the company to buy the domain from them as soon as possible [8].

Today, this type of domain-squatting is not as popular since companies usually register all appropriate domains, well before the company and its trademarks become popular. There are still cases however, where cybersquatters speculate the name of future products and services and register them, before the company marketing the product or service, does [2]. Coull et al. [3] have studied this phenomenon together with other domain registration abuses, such as *domain name tasting* and *domain-name front running*.

6.2 Typosquatting

Cybersquatting later evolved into *typosquatting*, i.e., the act of registering domains that are mistypes of popular authoritative domains, with the intention of capturing the traffic of users that make mistakes while typing a URL in their browsers' address bar. Such mistakes include missing-dot typos, character-omission typos and character-permutation typos. This practice can be traced back to over 13 years, since the 1999 Anticybersquatting Consumer Protection Act (ACPA) already mentions URLs that are "sufficiently similar to a trademark of a person or entity" [1]. In 2003, Edelman reported on 8,800 mistyped and cybersquatting domains that served sexually-explicit content, which he postulated were registered by the same individual [8].

Wang et al. [17] described a system for automatically discovering and analyzing typosquatting by simulating typing errors. The researchers also brought attention to the fact that the majority of the discovered typosquatting domains were pointing to domain-parking agencies, which were used to automatically serve ads related to the mistyped domain name. Banerjee et al. [2] identified that typosquatting extends to the abuse of domain suffixes, such as registering a typosquatting *.org* domain, for an authoritative *.com*.

Moore and Edelman perform a similar experiment to discover typosquatting domains in 2010 [14] and estimated that, at the time, at least 938,000 typosquatting domains targeted the top 3,264 *.com* sites. Interestingly, the authors point out that large advertising networks such as Google Ads, willingly cooperate with typosquatters by showing ads on the mistyped domains and should thus be held equally responsible for the damage against the authoritative domains that are being attacked. Apart from serving ads, there have also been documented cases of typosquatting domains used to serve malware [9]. Nikiforakis et al. [15] recently showed that typosquatting can also occur in remote script inclusions, where developers mistype the domains of remote code providers and thus make their sites susceptible to malicious script injections.

6.3 Homograph attacks

In a domain-homograph attack, an attacker takes advantage of the perceived visual similarity between two or more letters, in order to trick the user into believing that she is interacting with a specific authoritative web site while she is interacting with the attacker's site. This confusion may lead up to the user willingly submitting her credentials or other sensitive information. The main difference between these attacks and the aforementioned domain-squatting attacks, is that the homographed domains are usually spread-out through spam emails and social networks, instead of relying on user mistakes, since their construction cannot usually be achieved by the mistype of a letter for a neighboring one.

Gabrilovich and Gontmakher were the first to bring attention to the possible use of characters from non-Latin character-sets that look like Latin characters and could be substituted to confuse the user of the nature of a given

[2]Parked domain with ads - `www.iphone6.com`

domain [10]. For instance, an attacker could register `paypal.com` using the Cyrillic letter р (lower case "r", Unicode U+0440), which looks almost identical to the Latin letter "p".

Dhamija et al. [5], study the reasons which make phishing work, and make special mention of "visually deceptive text", i.e., domains that substitute characters with look-alikes within the same character-set, such as `paypa1.com` (last letter is the number "one" instead of the letter "l") and `bankofvvest.com` (two "v"s instead of a "w").

Holgers et al. [12] performed a large-scale study of homograph attacks by gathering popular domains and searching for homographed ones by substituting up to three characters of each domain, with their confusable counterparts. They discovered a total of 399 homographed domains, targeting 299 authoritative domains, from a corpus of over 3,000 domains. The majority of the discovered homographed domains were used to display ads to the visiting users. Others were listed for sale and some were even parodies of the authoritative domains that they mimicked. These results suggest that, while homography is used to construct confusable domains, the population of homographed domains is several orders of magnitude less than typosquatting and not exploited as much as it could be.

7. CONCLUSION

The importance of domains has made them an attractive target for malicious individuals. As the web expands, domain names can only become more popular and thus attacks against them are likely to become more frequent and more severe. Even though today, search engines greatly assist users in discovering web sites, domain names are still the de facto symbol of familiarity of any given web page appearing in a user's browser. Bitsquatting is the latest instantiation of attacks against domain names, but differs from its predecessors in that it relies on hardware failure rather than human error.

In this paper, we explored the impact of bitsquatting on the domain-squatting community and showed that domain-squatters have embraced it as the latest way of parasitically profiting on popular web sites. Bitsquatters were found to employ all the known ways of domain-squatters as a way of profiting: parked domains, affiliate abuse and malware installations. We hope that this study, can serve as a reference point for the dangers of bitsquatting and the need for appropriate reaction from companies that wish to protect themselves and their customers.

Acknowledgments: The authors acknowledge the support of EURid, the European Registry of Internet Domain Names. This research was performed with the financial support of the Prevention against Crime Programme of the European Union (B-CCENTRE), the Research Fund KU Leuven and the EU FP7 project NESSoS.

8. REFERENCES

[1] Anticybersquatting Consumer Protection Act (ACPA). `http://www.patents.com/acpa.htm`, November 1999.

[2] Banerjee, A., Barman, D., Faloutsos, M., and Bhuyan, L. N. Cyber-fraud is one typo away. In *Proceedings of the 27th Conference on Computer Communications, IEEE INFOCOM* (2008).

[3] Coull, S. E., White, A. M., Yen, T.-f., Monrose, F., and Reiter, M. K. Understanding domain registration abuses. In *Proceedings of the 25th International Information Security Conference (IFIP SEC)* (2010).

[4] Cova, M., Leita, C., Thonnard, O., Keromytis, A. D., and Dacier, M. An analysis of rogue AV campaigns. In *Proceedings of the 13th international conference on Recent Advances in Intrusion Detection (RAID)* (2010).

[5] Dhamija, R., Tygar, J. D., and Hearst, M. Why phishing works. In *Proceedings of the SIGCHI conference on Human Factors in computing systems* (2006), CHI '06, ACM.

[6] Dinaburg, A. Bitsquatting: DNS Hijacking without Exploitation. In *Proceedings of BlackHat Security* (July 2011).

[7] EURId Insight: Overview of DNSSEC deployment worldwide, October 2010.

[8] Edelman, B. Large-scale registration of domains with typographical errors, September 2003.

[9] F-Secure. W32/Googkle. `http://www.f-secure.com/v-descs/googkle.shtml`.

[10] Gabrilovich, E., and Gontmakher, A. The homograph attack. *Communucations of the ACM 45*, 2 (Feb. 2002), 128.

[11] Golinveaux, J. What's in a domain name: Is cybersquatting trademark dilution? In *University of San Francisco Law Review 33 U.S.F. L. Rev. (1998-1999).*

[12] Holgers, T., Watson, D. E., and Gribble, S. D. Cutting through the confusion: a measurement study of homograph attacks. In *Proceedings of the annual conference on USENIX '06 Annual Technical Conference* (Berkeley, CA, USA, 2006), ATEC '06, USENIX Association.

[13] Kesmodel, D. *The Domain Game: How People Get Rich from Internet Domain Names.* Xlibris Corporation, 2008.

[14] Moore, T., and Edelman, B. Measuring the perpetrators and funders of typosquatting. In *Financial Cryptography and Data Security* (2010), vol. 6052, pp. 175–191.

[15] Nikiforakis, N., Invernizzi, L., Kapravelos, A., Van Acker, S., Joosen, W., Kruegel, C., Piessens, F., and Vigna, G. You Are What You Include: Large-scale Evaluation of Remote JavaScript Inclusions. In *Proceedings of the ACM Conference on Computer and Communications Security (CCS)* (2012).

[16] Osterweil, E., Ryan, M., Massey, D., and Zhang, L. Quantifying the operational status of the dnssec deployment. In *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement* (New York, NY, USA, 2008), IMC '08, ACM, pp. 231–242.

[17] Wang, Y.-M., Beck, D., Wang, J., Verbowski, C., and Daniels, B. Strider typo-patrol: discovery and analysis of systematic typo-squatting. In *Proceedings of the 2nd conference on Steps to Reducing Unwanted Traffic on the Internet - Volume 2* (Berkeley, CA, USA, 2006), SRUTI'06, USENIX Association, pp. 5–5.