PaDAWaNS

Proactive Domain Abuse Warning and Notification System

by

T. L. M. Brands

to obtain the degree of Master of Science
at the Delft University of Technology,
to be defended publicly on Tuesday January 15, 2019 at 11:00 AM.

Student number: 4247132
Project duration: Sept 1, 2017 – January 15, 2019
Thesis committee:

Dr. C. Doerr,	TU Delft, supervisor
Dr. ir. J. C. A. van der Lubbe,	TU Delft, chair
Dr. D. Tax,	TU Delft
Drs. M. Wullink,	SIDN Labs

An electronic version of this thesis is available at http://repository.tudelft.nl/.

TU Delft

Abstract

The counterfeit market is rapidly expanding into the online realm. Large amounts of fraudulent webshops advertise luxury clothing and fashion accessories, but ship counterfeit products to their customers. Apart from customers, brand owners and domain registries experience a negative impact caused by these fake webshops. Current countermeasures are slow and of a reactive nature, leaving a large enough window of opportunity for criminals to make a profit. This thesis introduces a proactive mitigation approach that can be deployed at domain registries. By predicting whether a newly registered domain will be used to sell counterfeit merchandise, preventive countermeasures can often be taken in advance, minimizing the criminals' window of opportunity and profits. These predictions are made by training a detection model using both registrant information and infrastructure measurements of the registered domains. To evaluate the prediction system, new domain registrations are classified for a period of 6 months. Registrations classified as malicious are then monitored for signs of abuse. Overall, the system is able to detect malicious registrations with reasonable precision. Additionally, the body of abusive domain registrations created during this thesis project is analyzed to gain insights into the methods used to host counterfeit webshops, which can be used as a starting point for future research.

Preface

The past year and a half have been a wild ride. Work on this thesis has lead me past mind-numbing drudgery to exciting new insights and from stressful all-nighters to moments of relaxation, although the latter were mostly imposed by friends and family. As such I am writing this preface with conflicted feelings. At the one hand, the continuous extensions of deadlines through setbacks and health issues make the finish line a sight for sore eyes. On the other hand, although I am proud of the results of my work, I feel like there is still so much left to research, improve and explore. A similar feeling creeps up on me with the realization that the finalization of this thesis will also bring an end to my time at the Delft University of Technology. Some of the people I have met on campus, as well as some of the professors that taught me, have made a lasting impression on me. Through them, I engaged with topics and fields that I would never have on my own.

First and foremost I would like to thank Christian Doerr (TU Delft) and Maarten Wullink (SIDN), my thesis supervisors, for their guidance and support. Being able to discuss problems, both in and outside of the thesis project, proved to be invaluable. Looking back, my time at SIDN has been a great experience. I have had the privilege to be a part of the SIDN Labs team, where some of the brightest people I know work on making the Internet a safer, better place. I want to thank Giovane Moura and Moritz Müller, along with the entire team for their friendship, hospitality and insights.

Finally, I would like to thank my friends, family and especially my mother for their continuous support during this project. Thank you all for pointing out my own sense and nonsense throughout this project, and reminding that there is a world outside of my studies.

Contents

List of Figures

List of Tables

1

Introduction

With the ever-increasing connectivity and accessibility of the Internet, the rise of both new applications and threats often go hand in hand. A prime example hereof is e-commerce: with more and more consumers favoring webshops over brick and mortar stores, the anonymity of the Internet presents a field of opportunity for fraud and scams. This potential for abuse is realized with the rise of fraudulent webshops. Fake, or fraudulent webshops, are shops set up with the intention to scam people: sold goods are often counterfeits or from knock-off brands, or are not delivered at all. Furthermore, a customer's data can be used for credit card and identity fraud. These shops often offer designer clothes and shoes or fashion accessories such as purses and sunglasses from popular fashion and sports brands (e.g. Adidas, Gucci, Ray-ban). To unsuspecting shoppers, a fraudulent shop can easily come across as legitimate: the shops have a professional design, often display various 'certificates' that supposedly vouch for their legitimacy, and it is often possible to create an account.

A growing number of people become victim of fake shops, with 3% of online shoppers in the Netherlands indicating they have been the victim of fraud: promised goods were never delivered, credit card data stolen or identity stolen [5]. Fake webshops are a growing problem. The Dutch police has taken action against 438 fake webshops in 2017, a steep increase from the 35 take-downs in total throughout 2016 [51]. Unfortunately, these reactive actions are far from enough. According to Dataprovider.com, roughly 10% of the 85.000 webshops in the Netherlands in 2017 were likely fraudulent [22]. On a more global scale, a joint investigation by intellectual property crime institutes of Europol, the US, and 27 EU member states resulted in the take-down of over 20.000 domains in 2017 alone. Previous iterations of this recurring investigation in 2014, 2015, and 2016 combined resulted in the take-down of 7776 domains [23], indicating that the problem of fraudulent webshops is on the rise across the globe.

Aside from unsuspecting shoppers being scammed out of their money, brand owners are also the victim of fraudulent webshops. Global financial damages of international trade in counterfeit and pirated goods was estimated to be $461 billion in 2013, and is expected to rise to $991 billion dollar in 2022 [2]. The Directorate-General for Taxation and Customs Union of the European Union reported that counterfeit fashion accessories and clothes are leading the charts on both the amount of seizure procedures and retail value of seized goods, indicating that the sales of these goods is a significant part of the international trade of counterfeit goods [4]. As fraudulent shops act as a point of sale for elaborate counterfeit production chains, and the vast majority of these shops is offering clothing and fashion accessories, these numbers show that fraudulent webshops are part of a large international problem that has a big financial impact.

Apart from duped buyers and brand owners, fraudulent shops negatively affect another party: Top Level Domain registries. A Top Level Domain (TLD) is the last part of an Internet domain name. All Internet domain names fall under a Top Level Domain: www.tudelft.nl belongs to the .nl TLD, example.com to the .com TLD. TLD registries maintain the domain names registered within their respective TLDs. Their core business model is selling domain names. A Top Level Domain is a registry's most valued trademark, abusive domains within their TLD can cause reputational damage to their core busi-

ness. TLDs with a large amount of malicious domains are more likely to become distrusted by the general public, leading to a loss of traffic and, consequently, sales of domain names. If a TLD gets overrun by domain abuse, new businesses will avoid this TLD, favoring a TLD with a better reputation. While a large amount of TLDs exist, most people will likely only be familiar with the more well known and established TLDs such. .com, .net, .org and national TLDs such as .uk, .de and .nl. More recently introduced TLDs such as .xyz or .top are less known to the public, and domains using these TLDs are often perceived to be less trustworthy by Internet users [13]. This makes well established TLDs attractive to criminals setting up counterfeit webshops.

Another factor that attracts these malicious actors is the presence of residual trust. A domain name running a legitimate service builds up trust over time. When a domain name that has built up some trust is abandoned, its reputation and trust remains. This is known as residual trust: trust left behind by the previous owner of a domain. For fraudulent webshops (and malicious websites in general), residual trust is a desired property of a domain name: it lowers the chance at being detected by reputation-based security services. As there are, by volume, more domains with residual trust within established TLDs then within new TLDs, established TLDs are more prone to the abuse of residual trust.

SIDN is a TLD registry with .nl as its core business model. With over 5.8 million registered domains[1], SIDN is a critical part of digital infrastructure in the Netherlands. In order to keep innovating and improve their operations SIDN has a research department named SIDN Labs[2]. SIDN Labs actively researches a wide range of topics aiming to understand and improve the current state of the Internet and its Domain Name System, as well as the security of the .nl top level domain. Recent work done by SIDN Labs explored the resilience of the Domain Name System [36, 42] and the abuse of domain names in relatively new TLDs [35]. Preventing abuse of .nl domains by fraudulent webshops is a topic of interest for both SIDN's core business and SIDN Labs' research mission. The research presented in this thesis was done on behalf of SIDN Labs and Delft University of Technology.

1.1. State of the art

Fraudulent webshops are causing damage to both their customers and the brand owners of the counterfeit goods being sold. Furthermore, they are a threat to TLD registries. There are many countermeasures in place to combat consumer fraud and counterfeit trading. Enhanced inspections at customs, anti-fraud operations, and international treaties to crack down on illegal trade are all valid measures against the global problem of counterfeit trading and thus, fraudulent webshops. The focus of this thesis is not on these kind of international policies, but rather on technical measures at TLD registries to prevent the operation of fraudulent webshops. From this perspective, two methods are commonly used to fight these webshops: reputation tools and domain take-downs.

Reputation based tools are a common way to protect customers against fraudulent webshops (and other threats stemming from abusive domains). Most of them come in the form of browser plug-ins or DNS blacklists that block domains based on their reputation. Domain reputation is a score based on whether the domain is (or has been) associated with malware, fraud or other illegal or suspicious activities. Google Safe Browsing, Microsoft SmartScreen and McAfee WebAdvisor all use a form of domain reputation to determine whether a domain should be blocked. Once a domain's reputations drops below a certain threshold, it gets a negative reputation and it is blocked for the end user. While this is a decent solution for consumers, it requires the consumer to install a piece of protective software, meaning that unaware consumers are still at risk. Furthermore brand owners and TLD registries are not helped much by this solution. Some consumers are willing to take the risk of ordering counterfeit goods for low prices, causing financial loss for the brand owners. For TLD registries, requiring third party solutions to safely navigate their TLDs is an undesirable situation, as this reflects poorly on the safety of the TLD. In fact, having a large amount of domains with bad reputations in a TLD can be damaging for a TLD registry.

As stated earlier, multiple organizations try to combat the proliferation of fake webshops by seizing

[1] http://stats.sidnlabs.nl/#/registration
[2] https://www.sidnlabs.nl/over-sidnlabs?language_id=2

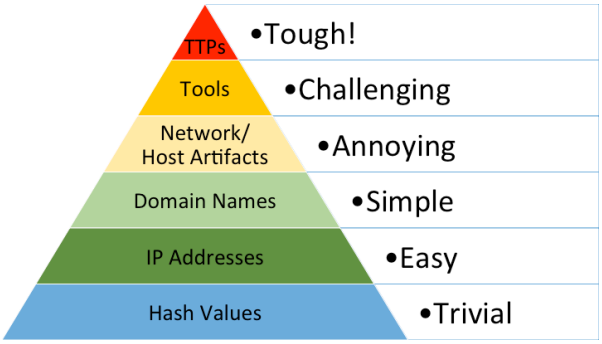


Figure 1.1: The pyramid of pain, as published by David Bianco [10].

their domains. Most of these organization react to reports of malicious shops from duped consumers. Then, an organization such as the police or a consumers rights group will investigate to determine whether a reported shop is actually fraudulent. Once it is established that a shop is indeed malicious, an abuse report or take-down request is send to a registrar or registry, asking to suspend the domain name. This request has to go through some bureaucratic processes before finally being approved and carried out. The domain is then seized and the deceptive shop is no longer available via the confiscated domain.

The problem with these measures is that they are reactive: the offending webshop has already been online for a period of time, and has likely already made victims before its domain was seized. Furthermore, blacklists and take-downs are a temporary solution. A malicious actor only needs to register a new domain name and possibly change from hosting provider in order to have a new, unblocked infrastructure ready to use. These reactive countermeasures lead to a game of whack-a-mole, with the authorities seizing domains and the groups behind the fraudulent webshops opening new shops ad infinitum. To break this endless circle, a different approach is required. Instead of waiting for consumers to report that they have been duped, one could actively scan the Internet for counterfeit shops in order to seize them sooner. This way the lifetime of a fraudulent webshop is shortened, reducing the amount of victims it can make. This is, in fact, the method currently used by SIDN.

1.2. Novelty

Of course, counterfeit traders can simply start making more fake webshops, exhausting the resources needed to impose countermeasures and carry on. Here lies the biggest problem for efficient protection against deceitful webshops: setting up new webshops is incredibly easy. Domain name registration is a short process with little financial cost. A large part of the process can easily be automated. If 100 fraudulent shops are taken down, 200 new shops can be set up within a week.

From this perspective it seems reasonable that, instead of trying to minimize the lifetime of a webshop, it might be more worthwhile to make creating a new fraudulent webshop harder for the counterfeiters. In order to make it harder for these actors to set up a new shops, this research aims to create a profiling system capable of detecting the methods and behavior of these actors. A system capable of detecting a domain registration as being made by a malicious actor would be very useful. Such a system could be used to shut down a domain moments after registration, before the actual abuse has occurred. It would make it harder (and thus more expensive) to successfully exploit domains at a large scale, as one would have to change many aspects of their observable behavior after every registration to remain undetected.

This focus on increasing the effort required from an adversary to reach his goal is not a new one. Within the security community, this idea is also known as the “pyramid of pain” [10] and was introduced as a classification method for attack indicators to defend against advanced persistent threats (APTs): criminal groups/organizations that use sophisticated attacks to achieve their goals. As can bee seen in figure 1.1, this pyramid consists of multiple levels containing assets of an APT. The lower region of the pyramid contains indicators that are relatively easy to change, such as hashes and IP addresses.

Each step up in the pyramid contains assets that are harder to change than the level below; Domain names and network artifacts are already harder to change than hashes, being able to detect an APT's tools forces them to create or purchase new tools. The top of the pyramid is for Tactics, Techniques and Procedures: These are indicators of an adversary's methods and behavior. Just as being able to detect an adversary's tools requires a change of tools, being able to detect an adversary on the TTP level requires a change of behavior. As behavioral change is often very hard for humans, TTPs are seen as the top of the pyramid of pain: once an adversary's behavior is detected, the adversary has to make a choice: fundamentally change the modus operandi or give up. The underlying thought is that focusing on detection of the lower levels (easy to change) only provides short term mitigation, while detection of the higher levels (hard to change) can provide long-term mitigation.

While not exactly an APT, the theory of the pyramid of pain is very much applicable to malicious webshops. For example, a single IP address would be on the lowest level, as it is trivial to change the domain name of a fake webshop. The methods used to register domain names for these shops are higher up in the pyramid, as these will cost considerably more effort to replace. This research aims to apply this principle to TLD registries by creating a classifier that is capable of detecting malicious actors using features that correspond to the higher levels of the pyramid of pain. This thesis will show the performance of such an approach and a detailed overview of possible features and their use in protecting a TLD infrastructure.

1.3. Research question

This research attempts to enable TLD registries to leverage their data to stop malicious actors effectively, preventing them from repeating their process over and over. The proposed approach to solve this problem is to extract complex behavioral features from the registration data and DNS logs of a TLD registry. Complex behavioral features are features that are not used just to detect the "external symptoms" of malicious domains (e.g. scanning domains for their content to detect fake webshops), but instead are used to describe behavior: the process used by actors creating these malicious domains (e.g. request patterns from recursive resolvers, registrars used). These behavioral features can in turn be used to create a system capable of flagging domain names for suspected abuse. By trying to profile the behavior of actors instead of the domains they create, it is likely that malicious domains are detected quicker than conventional detection methods. Moreover, the costs of starting new campaigns will increase for malicious actors. Instead of simply changing a domain name or hosting provider, the actor now has to change every part of his method that is observable as a behavioral feature to avoid detection.

In light of the above, the research questions can be defined as:

Research question

1. What data is available at Top Level Domain registries that can be used to profile their registrants?
2. What kind of repeated behavioral patterns are exhibited by registrants of fraudulent webshops?
3. What accuracy can be achieved by machine learning solutions trained on the behavior of previous fake webshop registrations?

The thesis will start with an overview of related work in section 2, in which prior research on leveraging data of TLD registries and DNS servers is presented. Then, the datasets available at SIDN are discussed in section 3, along with the processes used to obtain an initial ground truth. Thereafter, the designed detection system is introduced in section 4, along with chosen features and machine learning algorithms. Results of the evaluation of the detection system, as well as the results of an analysis into the registration methods deployed by fraudulent webshop operators are presented in section 5. Finally, this thesis is summarized and concluded in section 6.

2

Related work

This chapter will present an overview of research relevant to the research questions posed in 1.3. The chapter consists of two sections: the first containing background and related work regarding online counterfeit trading, the second detailing different methods used to classify malicious domain names.

2.1. Fraudulent webshops

This section gives an overview of the current body of knowledge on fake webshops and the actors involved. First, insight is given into the motivations of both the actors operating the webshops as well as their victims. Then a summary of work done to counter or frustrate operating fake webshops is presented. Finally, this section concludes with the solution proposed in this thesis and its relation to the existing work.

2.1.1. Why operate a fraudulent webshop?

The goal of running a fake webshop is to make money. There are different methods to generate revenue by operating a fake webshop. The most obvious one, of course, is to sell counterfeit products. The demand for cheap brand clothing and accessories will always exist, and unknowing consumers form an easy target. These consumers will order goods, expecting to receive a high quality product, but end up with a cheap knock-off instead, if anything at all. This scenario where the consumer is tricked into believing he is buying actual brand produce is called the deceptive market. The scenario where the webshop states upfront that the goods are counterfeit is called the non-deceptive market. Not all consumers value whether their purchases are counterfeit, as long as they look the part.

Pushing counterfeit produce does not always have to be the main source of revenue for a fake webshop. A different method to make money by running a malicious webshop is to gain personal and financial data of consumers. This data can then be used to commit identity or credit card fraud, or can be sold to others who wish to do so. Another method to make money through fake webshops is transaction laundering. In this scenario, the goal is not to sell goods nor steal data. Instead, money is transfered through webshops to obfuscate the true intention of a transaction. This way money flows related to illegal practices such as gambling or the import of illegal goods can go unnoticed by authorities. Of course, these motivations are not mutually exclusive, and can easily be combined. While this thesis focus is on webshops set up to sell counterfeit goods, these alternative motives should be taken into account.

In order to have a good understanding of the problem of fake webshops, it is important to look at this problem from multiple perspectives, including that of the perpetrators. While it is obvious that the main drive for selling counterfeit products is money, understanding how these actors view and justify their actions can lead to useful insights. Studies examining and interviewing vendors of counterfeit goods, both in the streets and on social networks, show that some vendors on the non-deceptive market are not aware of any wrongdoing [52]. Others argue that counterfeiting goods is not considered morally wrong in their culture and is socially accepted [28]. Those who do perceive their actions as immoral or

illegal argue that they are just trying to make a living. They further point to the lack of local regulation and enforcement that made their current occupation an attractive way to do so. Interestingly, vendors using social media or websites to move their produce repeatedly state the relative ease of marketing their goods as a reason for their current career choice. Counterfeited brands often publish high value marketing materials (i.e. images and descriptions of products) that can easily be re-purposed to promote counterfeit stores [52].

However, these vendors are only the proverbial tip of the iceberg of counterfeit trading. While the product outlets are the most visible part of the fake goods phenomenon, it is part of a large and often shady supply chain. Production of counterfeit goods is done by workers who labor under poor conditions. Their employers are often illicit businesses that have little regard for health and safety norms [65]. The produced goods are sold online and on underground markets by vendors and the profit ends up at the organization facilitating this supply chain. These organizations are often part of large organized crime syndicates, who have deployed counterfeit trading pipelines as a method to finance their activities [46]. Terrorist organizations have been seen using this method of generating income as well, with the sale of counterfeit luxury goods such as sneakers being tied to the funding of terrorist operations [61].

2.1.2. Why order goods from a fake webshop?

As with all goods that are being sold, counterfeit goods are subject to the rules of supply and demand. Without customers, the problem of fraudulent webshops would not exist. Of course, many (if not most) customers of fake webshops are unaware of the fact that they are ordering counterfeited goods. These consumers fall victim to the deceptive counterfeit market and feel cheated when their purchase turns out to be a fake product. In 2016, damages reported by customers in the Netherlands who received fake or no products amounted to half a million euros [25]. It is expected that only a fraction of the victims report the fraud, and that the actual damages are at least twice as high.

However, research has shown that there also is a genuine demand for counterfeit products. People consciously purchasing these products are often attracted to the high utility perceived in these goods. That is, purchasing fake luxury goods at low prices, while still reaping the benefits of the genuine product (e.g. high social status) [50]. Another important factor is the perceived risk from buying counterfeit goods: the risk of being caught by authorities (financial risk) and peers (social risk) [63]. These factors, in combination with lesser influences affect the consumers willingness to purchase counterfeit goods [9].

2.1.3. How to stop fake webshops?

The issue of the sale of counterfeit goods through the Internet has been an issue for quite some time. Nonetheless, relatively little research has been done into technical countermeasures. A possible reason for this is that the problem is only more recently becoming a widespread issue. Another explanation could be that counterfeit trading is often seen as a legal 'grey area', offending domains can be harder to take down [44] than other forms of domain name abuse (e.g. malware, phishing), making such forms of abuse a more appealing topic. SIDN, the .nl registry, reported in 2016 that the increasing number of fraudulent webshops was becoming a problem [59].

In years prior, the academic community was interested in how it could design methods to differentiate between genuine and counterfeit products for sale on the Internet [53]. In 2014, research by Stroppa et al. contained case studies on Facebook ads leading to counterfeit webshops [60]. These webshops would have domain names similar to the brands they tried to counterfeit (e.g. Louisvuittona.tld) and were abusing online fora, blogs, spam email and Facebook advertisements to promote their domains. The website would use images from the counterfeited brand's official site to appear as a legitimate vendor. Many of these tactics are still being used by fake webshops today.

In the same year Wang et al. described black hat search engine optimization (SEO) campaigns related to webshops for counterfeit luxury products [71]. They described the use of 'cloaking': domains that serve targeted keywords when visited by a search engine crawler, but will serve a webshop to normal visitors. Wang et al. crawled web search results for certain luxury clothing brands and trained a classifier on the content of the returned websites. This classifier was used to uncover 52 SEO cam-

paigns, which were then subjected to domain seizures to study the effect on the campaigns. Domain seizures were found to have some effect, but SEO campaigns were quick to deploy new domains, at a quicker rate than domain take downs were taking place.

Wadleigh et al. also investigated domains based on search results for luxury brands. They too created a classifier that used content-based features and enriched their data with public WHOIS records. The study showed similar results, noting that brands that enforced active domain take down policies were successful in lowering the rate of search result pollution [70]. Wadleigh later published additional research into compromised sites being used for black hat SEO for counterfeit goods, finding that plug-ins for content management systems with a large user base were exploited to compromise domains [69]. In 2017, Carpineto and Romano improved on the work of Wadleigh and developed RI.SI.CO., a more extensive classifier with more content-based features capable of detecting fake webshops within search results [15].

Wimmer and Yoon proposed a system to compute the likelihood of an online product being a counterfeit by utilizing natural language processing of user reviews on amazon [72]. Their solution acts as a decision support system implemented as a browser plug-in. When a user comes across a product, the plug-in gathers reviews and performs natural language processing and computes a counterfeit score, which is shown to the user. The user can then use this metric to make when deciding on whether or not to buy the product. While this is a good way to protect consumers it comes with several shortcomings. First of all, the solution requires the existence of honest user reviews. Many fake webshops come without user reviews (which should be red flag to an observant consumer) and those that do often fake them, nullifying added value of this solution. Furthermore, presenting a solution as a browser plug-in requires users to explicitly opt-in to this safety measure. Just as with many opt-in security measures, those who chose to use them are likely to already be more aware of the risks it mitigates, while unaware users will be less likely to use the plug-in.

Instead of trying to thwart malicious webshops with domain take-downs and search engine demoting, Tang et al. studied the effect of intervening at the level of the payment processors [64]. As paying with cash is hardly ever an option in online sales, fraudulent webshop operators use third-party payment payment processors to process their transactions. These payment processors, in turn, need a bank that is willing to do business with them. The researchers made 424 test purchases at counterfeit webshops, reporting every purchase to Visa. While this resulted in the closure of some payment processors and enhanced vetting of new payment processors at some banks, other payment processors began filtering (declining) transactions from the researchers. A cat and mouse game ensued between the researchers, who were dodging the transaction filtering, and the so-called 'bullet-proof' payment processors, who were finding more sophisticated ways to identify the research group. Moreover, some banks seemed to have no problem taking on counterfeit merchants, most likely due to lax regulation in their countries. This research illustrates one of the more successful methods to actually frustrate counterfeit webshops. However, remaining ahead of the filters of sophisticated payment processors can be a costly practice.

During the work on this thesis, two relevant papers were published regarding online counterfeit trade detection. Both works focused on image processing on social media and image sharing platforms. These platforms are often used to advertise and sell counterfeit goods. Mangiatordi et al. created a multimedia analytics platform capable of scraping these platforms and extracting textual information and QR codes from images [40]. This textual information consists of email addresses, instructions on how to order goods and links to hidden services of the seller. The extracted information can then be used to investigate fraudulent webshops, either manually, or by mining the raw data extracted by the platform. Cheung et al. used a convolutional neural network (CNN) to discover merchants using images that likely come from the same source. Counterfeit vendors often use similar sources to promote their merchandise and apply cropping and watermarks to avoid discovery. The proposed framework utilizes a CNN to compute connection scores between merchants on social media and marks merchants with strong connections to known counterfeit sellers as suspicious [17]. While these solutions focus on social media, it is likely that they could be used to identify counterfeit shops with a dedicated website, given enough resources.

Concurrent to this thesis project, Cox and Haanen, together with SIDN, developed a proof of concept detection tool for fake webshops [20]. The tool extracts content-based features from a website's source page, utilizing both HTML attributes and natural language processing. These features are fed to an AdaBoost classifier, which has been trained on a set of known fake webshops and benign domains. While this method shows promise, experimental results indicate that further research is needed to make it viable, as the current accuracy is to poor to use the tool in production.

Most of the related work described here relies on the content of counterfeit webshops to detect them. This means that these solution can only identify a fake webshop once it is already online. This approach is of a reactive nature and experimental results so far have shown to at best lessen the proliferation of counterfeit stores on the Internet. The notable exception here is Tian et al. [64], who approached the issue from a strategic point of view, correctly identifying and targeting payment processing as a choking point for fraudulent merchants. This thesis aims to target another choking point: access to domain names. As described in previous work, counterfeit traders rely on their access to domain names [44, 59, 71] in order to sell their goods and perform SEO. Detecting these domain names once they are abused is possible, but combating counterfeit traders using this method proves challenging and ineffective, as new domains will be used to replace the seized domains. [70, 71]. By recognizing registrations made by counterfeit traders at the TLD registry level, it is possible to block access to fresh domain name registrations needed for their operations. By only utilizing hosting infrastructure and domain name registration details, counterfeit registrations can be identified before the domain name has been abused. This way, instead of treating the problem by taking down counterfeit webshops once detected, this research aims to prevent counterfeiters from setting up shop in the first place. After all, an ounce of prevention is worth a pound of cure.

2.2. Forms of domain abuse

There are two ways in which abusive domains come into existence: domains that are specifically registered for abuse and domains that were compromised and abused. Fraudulent webshops seem to almost exclusively favor the former method. Domains registered with malicious intent are often abused within the first few days of registration. The solution to mitigating this form of abuse is to detect them at registration time, essentially predicting future abuse [44]. There have been few attempts to predict abuse of compromised domains, as this is a much more difficult problem. Domain abuse is predominantly a means to financial gain to criminals. This section aims to give an overview of the different abuse cases prevalent in the .nl zone and how they generate revenue for criminals.

2.2.1. Spam

Spam email is likely the oldest form of domain abuse. A spam email is an unsolicited email sent with the goal to create revenue for the sender. Spam emails are emails send out to massive lists of email addresses. While the majority of sent spam emails are ignored or filtered, the small percentage of recipients that fall for the spam email is enough to generate revenue for the abuser. Spam comes in many forms. Spam methods range from offering fake or counterfeit goods [38] to encouraging readers to invest in cheap financial stocks as part of a pump and dump scheme [27].

2.2.2. Phishing

Phishing is the practice of misleading people in order to make them perform a certain action in the attacker's advantage. A common phishing attempt is to try to steal sensitive information from a victim. An attacker usually replicates a login page of an company or institute, hoping that an unsuspecting victim will mistake it for the real site, and enter his credentials. The acquired credentials can then be used to compromise the user's account. Other popular phishing targets are credit card numbers and social security numbers. Phishing can also be used to trick a victim into installing malware on his system. This is often done by sending an email with a malicious attachment, and convincing the user to open it. Another approach is to trick a victim into downloading and executing malware by posing it as a software upgrade. Email is often used to start phishing attacks, but phishing domains have also been seen using social media, advertisements and search engine optimization to lure victims to their site.

Phishing can generate revenue in multiple ways. Retrieved information such as credentials and credit card information can be sold on online black markets. Phishers tricking victims to install malware get a compensation per install [14]. More sophisticated and targeted phishing (spear-phishing) can be used to get a foothold in an organization's network, or to obtain sensitive information that can be used to blackmail victims.

2.2.3. Malware infrastructure

Most of the abuse not yet covered in this section can be categorized as malware infrastructure. These domains are used to facilitate the operations of malware. One example of this would be payload distribution. Malicious files attached to phishing mails discussed in section 2.2.2 seldom contain the actual malware itself. Instead, they are "droppers": small programs that download the malware from a distribution domain. Another form of malware infrastructure are command and control (CnC) servers. These servers are used by criminals to send instructions to infected systems or to let infected machines upload sensitive data.

2.3. Domain name classification

This thesis aims to detect domain name registrations from counterfeit traders. To gain insight in how one might perform this task, this section will provide an overview of the current state of automated domain name analysis using DNS and registration data. Most of the recent studies in this field applied machine learning to detect abused domain names. First, a summary of commonly used data sources is given. Afterwards, the features used to classify domain names in these studies will be explored. Finally, an overview of used machine learning methods, both supervised and unsupervised, will be given.

2.3.1. Datasets and measurements

Almost all research on domain analysis uses some form of infrastructural data. Most only resolve domain names to their IP address, others also resolve IP addresses of nameservers. Beyond this shared dataset, there are three categories of data that are used in varying degree.

Traffic observation

Different approaches to domain classification use various sources to obtain the required datasets. Many studies used large recursive DNS servers as their main source of data [7, 11, 49, 57]. From this point, one can observe queries from end-users (clients) to the DNS servers. As the stub resolvers used by the clients usually use little to no caching, every request made by the client becomes part of the data. This makes datasets obtained from these servers relatively fine grained. Others have focused on higher DNS levels. Antonakakakis et al created a method for large DNS operators [8]. They designed features specifically geared to work for DNS servers higher in the DNS hierarchy (e.g. TLD authoritative servers). These features were designed taking into account that caching at the lower levels will affect the data at the higher levels.

Registration databases

To retrieve newly registered domains, some researchers made use of DNS zone files from various top level domain operators [30]. Others went further and enriched this data with registration data containing name and contact information of registrants [24, 62, 67]. This information is either gained from public WHOIS databases, or from direct access to a TLD registry. The combination of zone files, together with registration information from public WHOIS services or registrar databases provide a solid base of data for research on registration-based domain name classification.

Ground truth

Almost all domain name classification research needs some form of ground truth to properly train classifiers and evaluate their results. Consequently, the validity of the methods used to derive a ground truth directly impact the validity of a study's results. The majority of existing works use a combination of blacklists as basis for their ground truth of malicious domains. Examples of popular blacklists include

Spamhaus[1], URIBL[2] and hpHosts[3]. In addition to public and commercial blacklists, some researchers operated spam-traps and honeypots or run known malicious executables in a sandbox to extract contacted domains [7, 19, 30, 56].

Apart from a list of known malicious domains (blacklist), a list of known benign domains (whitelist) is often required as well. Some approaches simply consider all domains not on their blacklist as benign [18, 24, 45] and try to verify this claim manually. Others use a site reputation services such as Norton Safe Web[4] and McAfee Site Advisor[5]. One of the most popular sources for known benign domains are Alexa toplists[6]. Many researchers reason that most popular sites on the web are very unlikely to be malicious, especially when they are listed consecutively over a larger period of time [7, 8, 11, 19, 49, 57].

Besides using these sources, many researchers also use manual labor to evaluate domains. The goal of these studies is often to improve upon existing detection methods, and thus to find malicious domains that have so far remained undetected by blacklists. To ensure the validity of the manual validation of researchers, a subset is often checked by an independent expert in order to determine how reliable the researcher's manual validation was [24, 45, 49]. Vissers et al use a different method to prove that newly discovered malicious domains are correctly labeled as such. Their solutions intentionally leaves out some independent features while training their classifier. To validate the classifier's result, they check these features for similarities between blacklisted domains and domains classified as malicious to claim their classifier is likely correct [68].

2.3.2. Features

The aim of this section is to give an overview of features used in prior research. These features have been divided over three categories: lexical, request-based and infrastructural.

Lexical features

Lexical features are features derived from lexical properties of domain names. These features are relatively easy to obtain, since in most cases only the domain name in question is necessary. Lexical features are an often used to detect domains created by Domain Generation Algorithms (DGAs). These domains are often easily spotted because, unlike benign domains, they do not resemble a recognizable word, brand, name or sentence. Moreover, this also holds true for some non-DGA malicious domains, as criminals often do not need an human-friendly domain name for their practices. Because of this, features that distinguish human-readable names from random or nonsensical strings are considered good for detecting these abusive domains. In an effort to make this distinction, Ma et al [39] tokenized every symbol in domain names, essentially creating a "bag of words" model. Variations on this idea were used by various researchers, who did the same with bigrams and trigrams [7, 19, 30, 74]. Antonakakis et al also used the average domain name length and its standard deviation to classify clusters of related domains [7]. Hao used the domain length as well, and also computed the ratio of the domain name length to the longest English word contained within it, a feature first proposed by Bilge et al [11, 30]. Shi et al also used domain length, and proposed the longest consecutive repetition of a single character and the entropy of a domain name as additional lexical features [57].

Another lexical approach is to look for similarities between domain names. Some criminals make use of typo-squatted domains, either to increase the authenticity of their phishing page or simply to perform a drive-by download attempt on unlucky visitors. Both the detection and occurrence of typo-squat domains has been a popular research subject [6, 67]. Szurdi et al generated a set of possible typo-squat domains for the .com zone and found that, out of those that were actually registered, half of the typo-squat domains were malicious [62]. Similarity can also be used in favor of the defense: Hao et al computed distances between new domains and known malicious domains, as well as the distance between new domains suspected to be registered in a single batch [30].

[1] https://www.spamhaus.org/dbl/
[2] http://uribl.com/
[3] https://www.hosts-file.net/
[4] https://safeweb.norton.com/
[5] https://www.siteadvisor.com/
[6] https://www.alexa.com/topsites

DNS request features

DNS request features are features that can be extracted from observed DNS queries. Most DNS request features resolve around patterns in DNS queries over time. While these features prove useful when trying to detect abuse as it happens, there are no known solutions that employ them to predict abuse. Hao et al showed that malicious domains are queried more than 40 times as much as legitimate domains in the first days after creation [29]. Moura et al used this property to develop nDEWS, a system that uses the DNS request volume and the diversity of request source IP, source country and source autonomous system to detect abuse of newly registered domains [43]. Requester diversity was also used by Antonakakis et al for Kopis, albeit in a more extensive way, using the mean, standard variation and variance of the BGP prefixes, country codes and autonomous systems of the requesters [8]. Other studies worked on the assumption that hosts that query a known malicious domain name are likely to query more bad domains. They considered such hosts infected, and used co-occurrence of domain names queried by infected hosts to discover both new malicious domains and infected hosts [26, 37, 56].

Infrastructural features

Infrastructural features are based on the infrastructure behind a domain name(e.g. IP address, autonomous systems, name servers). While a lot of infrastructure is shared between multiple (unrelated) domains, it turns out that a significant amount of abuse originates from distinct parts of the Internet's infrastructure [41]. These features are popular in proposals geared towards abuse prediction, as they allow to leverage this non-uniform distribution between domain abuse and infrastructure. Hao et al did this by using a domain's authoritative nameservers and their corresponding IP addresses and autonomous systems as features for their classifier [30]. Chiba et al focused on IP address spaces and octets and bitstrings extracted from IP addresses to use as feature vectors [18]. Other studies specifically started out with known malicious domains, and looked whether an observed domain's IP addresses and BGP prefixes coincide with those of bad domains [7, 11].

Another approach is to watch for changes in infrastructure over time. To prevent getting caught by these statistical models, criminals deploy the use of compromised hosts, using these systems as a network of proxies to increase resilience for their own services. Holz et al used a domain's distinct IP addresses, nameservers and ASNs in a specified time window as features to detect these so called Fast-Flux domains [31]. Others used similar features for his passive analysis solution, adding (among others) reverse DNS requests and the amount of distinct country codes as additional features [48, 49]. Felegyhazi et al proposed that Fast-Flux domains were more likely to use relatively young nameservers that are responsible for their own resolution. They further observed known bad domains for changes to a new nameserver, and searched for domains that changed to this nameserver at the same time to uncover new malicious domains [24].

Another method used by criminals to increase availability of their servers is to set low TTL (time-to-live) values for their records, in combination with a round robin DNS setup. The low TTL ensures that the record will only be cached for a short time, causing more frequent requests. When using round-robin DNS, the nameserver will rotate over multiple IP addresses, meaning that if one IP becomes unavailable a new one is quickly supplied, minimizing outages. For this reason, various studies use the observed TTL of a domain as a source for features [11, 49, 57].

Other features

Some researchers looked outside the scope of conventional features described in the above categories. Soska et al proposed to predict what benign sites will be compromised within a year. To this end, content-based features were used (e.g. CMS name and version) by scraping websites and parsing their HTML. These features were combined with information extracted from the Alexa Web Information Service (AWIS) [1] and yielded a 66% true positive rate and a 17% false positive rate [58]. Chiba et al uses the occurrence and absence of various domain name lists (e.g. Alexa and hp-hosts) over time to compute temporal variation patterns (TVPs). The research demonstrated that TVPs of domains can improve previous solutions such as those created by Antonakakis et al by combining TVP features with the features of the existing solutions to improve performance [19].

2.3.3. Machine learning

A multitude of different machine learning techniques have been used to classify domains. This section first introduces decision trees, a approach popular throughout many studies. Afterwards, several recent approaches observed in domain name classification studies are discussed.

Decision trees

A popular classification method seen in many studies is the use of decision trees [11, 45, 49, 58]. Decision trees are trees that have an attribute evaluation at every branch and classes at its leafs. Items are classified by starting at the top of the tree and moving through the tree until a leaf has been reached. Decision trees are popular because they are easy to interpret by humans. Its decisions can be simply explained by taking an item through its evaluation path, resulting in a boolean logic statement. Decision trees are relatively simple models and can have difficulties recognizing complex relationships (e.g. XOR). The trees tend to over-fit and can be non-robust, leading to large differences in results when small changes are introduced to the data. To compensate for this shortcomings, some researchers used random forests [8, 19]. A random forest uses a consensus vote from multiple decision trees constructed from a random subset of features, which improves its accuracy.

Several recent approaches

Hao et al used a Convex Polytope Machine (CPM) to classify domains. CPM uses multiple weighted linear Support Vector Machine (SVM) -like sub-classifiers and uses the highest scoring sub-classifier to classify observations. CPM achieves comparable accuracy to SVM, while achieving faster computation times up to 5 orders of magnitude [3]. Hao et al compared the two methods specifically for their classification problem and reported that CPM outperformed SVM on both accuracy and efficiency [30].

Shi et al deployed Extreme Learning Machine (ELM) for their classifier. ELM is a machine learning technique based on Single-hidden Layer Feedforward neural Networks (SLFNs). Its main distinctive feature is that it assigns hidden nodes and input weights randomly, as opposed to tuning them like conventional SLFNs. This property causes the resulting SLFN to essentially become a linear system, resulting in training times that are several orders of magnitude faster. Despite not tuning the hidden nodes, ELM achieves accuracy comparable to conventional SLFNs approaches [32, 33]. Shi et al compared ELM to Logistic Regression (LR), Classification and Regression Tree (CaRT), Back Propagation Neural Network (BPNN) and SVM. The results showed that ELM has significantly shorter training times compared to BPNN and SVM while preserving comparable accuracy. Compared to LR and CART, ELM achieves higher accuracy, albeit at the cost of a higher computation time [57].

Visser et al proposed complete-linkage agglomerative clustering to perform automatic campaign analysis on a corpus of malicious domains [68]. Agglomerative clustering merges the two closest clusters or observations on every iteration. This means that only observations with a similar feature vectors will be clustered, and outliers are unlikely to become part of a cluster. The stop criterion was the maximum V-measure; a cluster evaluation measure based on homogeneity and completeness [55]. It should be noted that V-measure requires the ground truth of the clustered data, meaning that it cannot be used on unlabeled data. However, intelligent stopping criteria that do not require ground truth could be used to perform the clustering on new, unlabeled observations.

3

Available data and measurements

As the operator of the .nl registry, SIDN has access to some unique datasets related to the .nl domainspace. This chapter discusses the datasets used to classify domain name registrations and their potential to be leveraged to identify fraudulent webshops. First, a registry's role within the Domain Name System is discussed, along with the datasets that a registry obtains in this role. Afterwards an additional dataset is introduced that will be used to enrich the data available at SIDN. Finally, the construction of a ground truth is described in section 3.3.

3.1. The Domain Name System

On the Internet, machines connect with each other using numeric addresses known as IP addresses. The amount of possible addresses is vast: approximately 4.3 billion (2^{32}) possibilities exist for IPv4, which by now have all been assigned. The next generation of IP addresses, IPv6, consists 2^{128} possible addresses, ensuring possibilities for future growth on the Internet. Remembering the correct IP for required services or locations on the Internet would be a rather cumbersome task for most humans. The need to give locations a human readable name, that reflects the nature or intended use of the system and the services hosted on an address, eventually lead to the creation of the Domain Name System (DNS).

DNS is a system that manages information related to domain names. Different types of information can be accessed by sending the appropriate query for a domain name. The most prominent function of DNS is translating domain names into the actual IP address of a server. It can be thought of as the Yellow pages of the Internet, helping people connect to the servers they need.

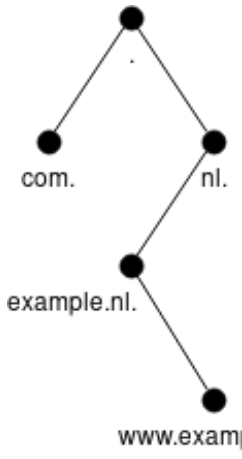


Figure 3.1: Different layers of hierarchy within a domain name.

Domain names adhere to a certain tree-like hierarchy that is present in the Domain Name System. As can be seen in figure 3.1, a domain name consists of a concatenation of labels, separated by a full stop ('.'). The order of the labels indicate their place in the hierarchy, with the most-left and most-right label of a domain name corresponding to respectively the lowest and highest level of the DNS hierarchy. The highest level of the hierarchy is shared among all domain names and is called the root label, denoted as a dot ('.'). Directly below the root (and specified at the right end of a domain name) are the top level domains (TLDs) such as *.com* and *.nl*. From here on, each level further down is called a subdomain of the higher level domain it falls under. Equivalently, each label in a domain name indicates a subdomain of the label after it (excluding the TLDs).

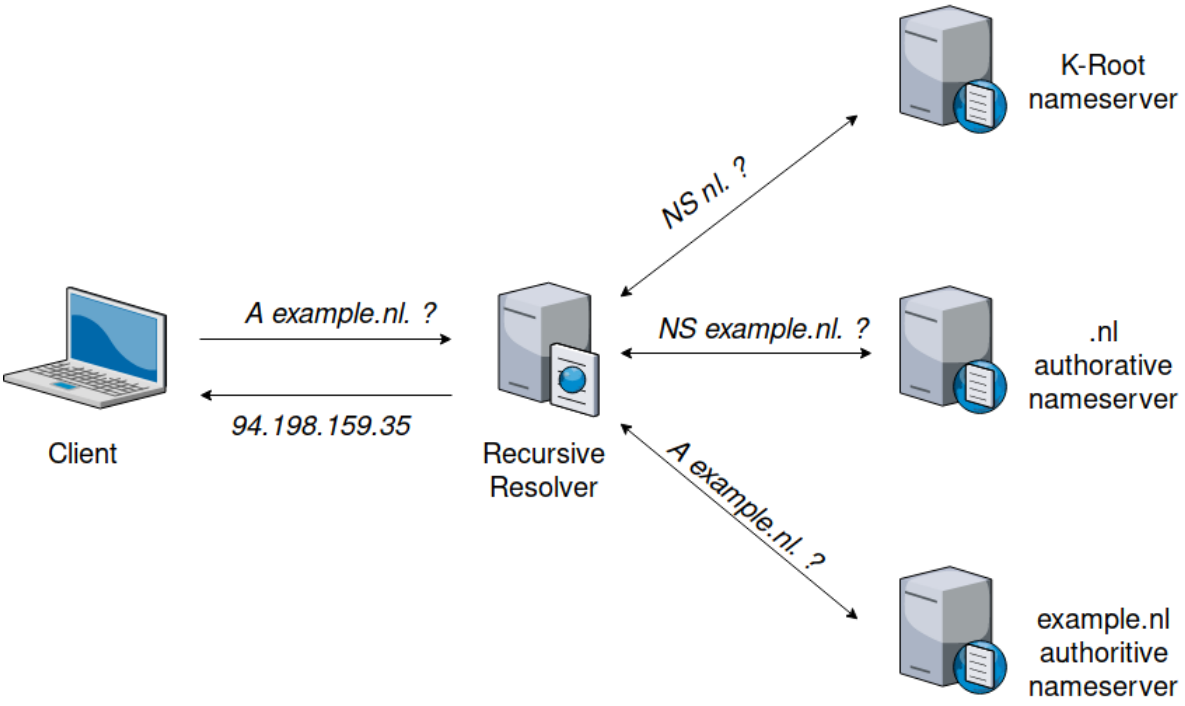


Figure 3.2: Resolution of a domain name.

3.1.1. From domain name to IP address

The Domain Name System consists of a hierarchical network of name servers, containing addresses of domain names for which they are authoritative. These name servers can be queried using the eponymous DNS protocol. The process of finding the IP address belonging to a domain name is called domain name resolution, or DNS resolution. To resolve a domain name, one recursively queries name servers until the name server authoritative for the domain name replies with an address. An example of DNS resolution is shown in figure 3.2. As can be seen, a domain name is resolved along the hierarchy of the DNS system. When a client needs to resolve a domain name, in this case *example.nl*, the following steps are taken:

1. Assuming the client does not have the address belonging to the domain name in its cache, a query is send to a DNS resolver. Which DNS resolver is queried depends on the client's configuration. In many cases, the queried DNS resolver belongs to the client's Internet Service Provider unless a specific resolver has been configured. Usually, a DNS resolver is a recursive resolver, meaning that the resolver will recursively resolve the domain and only return the final answer containing the address.

2. After receiving the request for *example.nl*, the recursive resolver starts to resolve the domain name by querying the root for the authoritative name server for *.nl*. This, of course, is under the assumption that the recursive resolver does not have the domain name's address in its cache. Otherwise, the recursive resolver will simply return the cached address.

3. Once the recursive resolver has received the location of the *.nl* authoritative name server, it queries this name server where to find an address for *example.nl*. The *.nl* authoritative name server responds with the address of the authoritative name server of the second-level domain *example.nl*.

4. The DNS resolver queries this name server for the address of *example.nl*, which it retrieves from the name server.

5. Finally, the resolver replies to the client with the address belonging to *example.nl*, enabling the client to connect to the system belonging to the domain name.

SIDN is responsible for the authoritative name servers of the *.nl* zone, as is depicted in figure 3.2. This means SIDN has the names and addresses of authoritative nameservers for all domains in the .nl zone. This data can be used to detect abusive domain name registrations, as it is possible that a malicious actor utilizes the same name server for multiple malicious domains.

3.1.2. Domain name registration

A domain name is a human-readable identification string used to indicate a location on the Internet. As such, it is overseen by ICANN, the Internet Corporation for Assigned Names and Numbers. ICANN maintains the root zone, meaning that ICANN maintains the root name servers containing the addresses of the authoritative TLD nameservers. The right to add TLDs to the root zone is reserved to ICANN. The right to add second-level domains to a TLD zone is delegated to the registry of the TLD. SIDN is the registry for .nl, meaning it is the only party allowed to add domains to the .nl zone. However, SIDN does not offer direct domain registration to the public. Instead, it allows intermediary companies to register .nl domain names for their customers. These intermediary companies are called registrars and their customers, the entities actually registering domains, are called registrants. A registrant orders a domain from a registrar, which in turn registers the domain name at SIDN on the registrant's behalf.

To register a domain, a registrant has to disclose some information to the registrar in order to construct a valid WHOIS record [21]. This information is passed to SIDN, where it is stored in their contact information database. This database contains personal data (i.e. names, addresses, phone numbers and email addresses) for every registrant that has registered a domain name. This data, detailing the registrant's personal data can be used to both spot repeat offenders, and discover patterns within the data of abusive registrants. Fraudulent registrants are of course aware of this, and use a myriad of faked registration details, limiting the reuse of each fictional registrant to only a couple of registrations. Most registrant details are easily faked as a registrant's name and address are not validated nor verified by many registrars. Phone numbers take some effort to fake, as they are often validated, meaning that malicious actors have to come up with phone numbers that actually exist. Still, phone numbers are not verified, so malicious actors do not have to actually own the telephone number used for their registration request. The most interesting piece of registrant information is the email address, as according to ICANN rules, this address has to be verified by the registrar within 14 days[1]. Not only does a criminal have to supply an existing address, he also needs to have ownership of this address, meaning the address can't be faked.

Once a domain has been registered, it is owned by the registrant, meaning the registrant is allowed create DNS records for the domain name. However, domain name registrations for most TLDs are not permanent, but have an expiration date. A registrant has to renew a registration before it ends, essentially 'renting' the domain name from the registry. If a domain name is not renewed on time, or deleted by the holder, it is put into quarantine for 40 days. A domain name put into quarantine is no longer usable: the DNS records are removed from the authoritative nameservers of the .nl zone. A quarantined domain can not be registered by anyone except for the previous registrant, offering a final chance to keep control over the domain name before it is available for registration to the public.

The moment of registration, expiration and release from quarantine of a domain name are available at SIDN and could be useful to detect malicious registrations. As prior research was able to detect business day-like patterns in malicious registrations [68], analyzing the moment of registration might uncover useful patterns. Moreover, since fraudulent merchants are actively trying to boost their visibility in order to get more traffic [71], it is likely that they also try to 'drop-catch' domains: registering domains just after they come out of quarantine, in order to profit from the domain's residual trust and search engine ranking. This behavior could be used as an indicator of malicious registrations if it is also exhibited by malicious webshop registrations.

3.2. Infrastructure

It is not unlikely that the groups running fraudulent webshops are using the same infrastructure for many of their websites. As such, the hosting and registration infrastructure utilized by counterfeit traders might

[1] https://www.icann.org/resources/pages/approved-with-specs-2013-09-17-en#whois-accuracy

prove useful to detect new malicious registrations. The first and most visible part of infrastructure used by counterfeit traders is the choice of registrar. As all .nl registrations are made through a registrar, the registrar used for every domain registration is known at SIDN. Moreover, as described in the previous section, SIDN has data on authoritative name servers for every registered domain name. An interesting addition to this dataset would be the IP addresses of the servers used to host the fraudulent domains. While SIDN does not have access to such a dataset, SIDN Labs is part of a research project that does: OpenINTEL.

OpenINTEL is an active DNS measurement platform, measuring 50% of the global DNS namespace [66]. It is a joint project between the University of Twente, Surfnet, and SIDN Labs. OpenINTEL measures every second-level domain name of a select set of top level domains on a daily basis. An OpenINTEL measurement consists of sending a fixed set of different DNS queries (e.g. A, MX, TXT) to a domain. All replies to these queries are then stored to form a historical database of DNS responses. Fortunately, the .nl zone falls within the scope of the OpenINTEL measurements, making it a suitable dataset for this research.

In order to get more data on the hosting infrastructure of registered domain names, a mapping of IP addresses to .nl domain names is retrieved from the OpenINTEL dataset. The dataset of all queries made to the .nl zone contains a significant amount of duplicates: most domains are not moved to a new host on a daily or even monthly basis. As such, the dataset is filtered to only include changes in IP addresses over time. First, the OpenINTEL database is queried for all observed pairs of domain names and IP addresses that have not been observed the day before, creating a dataset of (domain name, IP, timestamp) tuples. As the OpenINTEL dataset does not track domain registrations, the domains in the retrieved dataset are linked to their respective registrations by looking up the active registration at the time of observation. Afterwards, duplicate (registration, IP address) pairs are discarded, keeping only the first sighting of such a duplicate group. This procedure results in a set of IP address changes observed for a registered domain name during its lifespan. This same method is used to map the autonomous systems where a domain is hosted to domain name registrations.

3.3. Ground truth

Classification problems require a training set: a subset of the data for which labels are assigned indicating the classes they belong to. These labels are often called the ground truth. The remainder of this chapter will discuss the methods used to create a ground truth for the fraudulent webshop classification problem. This ground truth is used by the detection tool discussed in section 4. First, a description of fraudulent webshops is given, detailing characteristic properties that can be used to create a body of confirmed fake webshops. Afterwards, a method of creating a set of domains that are guaranteed not to be related to fraudulent webshops is described.

3.3.1. Known fraudulent webshops

In order to gain insight in the suitability of features to recognize fraudulent webshops, a body of confirmed fraudulent webshops is required. The only method of indisputably proving a webshop to be fraudulent is to order some goods, and authenticate the returned goods, if any. This, of course, is a lengthy and poorly scalable approach to create a body of known fraudulent shops. An alternative approach is to judge a webshop by its content. Fraudulent webshops have certain properties that distinguish them from their legitimate peers. One of such telltale signs is the domain name being used. In order to boost the search ranking of a webshop, dishonest merchants register expired domain names in order to exploit the residual trust of the domain. The search and reputation scores left by the previous domain owner are leveraged to gain better visibility in search engines. As a result, fake webshops often have unrelated or nonsensical names. For example, domain names formerly belonging to bakeries, barbers or consultancy firms are now being used to sell sneakers, jewelry or luxury coats. On the other side of the spectrum, some fraudulent shops have been observed using overly specific domain names, geared to a very specific product line (e.g. adidasstansmith.nl). This should be considered a red flag, as most major brands will only sell through their own website, or through established resellers with their own name and domain.

Figure 3.3: Multiple 'certificates' without any value.

Another red flag are the prices advertised on a webshop. Fraudulent webshops often mark all of their goods at a discount ranging between 40% to 80% below the market price. While such discounts can occasionally be encountered on legitimate webshops, it is rarely the case that a webshop has its entire collection on sale. The old adage "If it seems too good to be true, it probably is." seems very applicable to the world of online shopping.

An indicator of a legitimate webshop would be one or more certifications of a webshop certification program. These certification programs usually ask a fee to certify an online store and periodically check in to see if a webshop is up to its standards. Furthermore, many of these certification programs offer mediation between customer and webshop in the case of a dispute. This makes these programs undesirable to counterfeit traders, who favor to simply put a fake certification on their webstore. These fake certifications are easily spotted, as valid certificates link back to a page of its issuing organization. Of course, fake certificates do not have such a page to link to, so these are simple images. In fact, a large amount of fake webshops has a myriad of meaningless certifications crammed into a single image, such as figure 3.3, likely in the hope to lull potential victims into a false sense of security. Furthermore, legitimate webshops are required to share contact and registration information under Dutch regulation[2]. A webshop has to publish an email address for customer contact, the registered business name of the company behind the webshop and the corresponding registration number of the Dutch chamber of commerce. Of course, fake webshops do not have this information, and often do not even bother to put up an email address.

Yet another indicator is the absence of HTTPS certificates. The majority of webshops in the .nl zone deploy HTTPS[3], as this is required to pass most basic webshop certifications in the Netherlands. However, webshops (and perhaps websites in general) without HTTPS should be seen as suspicious or, at the very least, negligent.

A final red flag comes in the form of grammatical and translation mistakes. Most fraudulent webshops in the .nl zone are targeted at a Dutch audience. As such, counterfeit traders use automated translators (e.g. Google translate) to create the text for their webshops. This results in grammatical mistakes and erroneous sentence structures that are uncommon for anyone with a basic level of understanding of the Dutch language. Some words that have multiple Dutch translations depending on context or intention are often mistranslated as well. In many cases, these mistakes are clearly visible on the home page, in the fake shop's privacy and return policies and their product descriptions.

Using these indicators it is possible to identify fraudulent webshops by simply inspecting their website. While this method is not as indisputable as ordering actual goods, it more scalable, as it takes someone with a bit of training around 15 seconds to classify a webshop. Of course, manually inspecting all 5.8 million domains in the .nl zone is not feasible, so an automated way of finding fraudulent webshops is required. Luckily, SIDN was already doing some work on content-based webshop detection, using their Dmap [73] tool to scan the entirety of the .nl zone. This detection worked by comparing HTML headers to a list of keywords known to be often used by fraudulent webshops. This lists consisted mostly of luxury clothing and accessories and their brands, and broad categories (e.g. men, women and children). Furthermore, the list was complemented with Dutch words often used to attract

[2] https://www.acm.nl/nl/onderwerpen/verkoop-aan-consumenten/verkopen-aan-consumenten/uw-bedrijf-vindbaar-en-aanspreekbaar

[3] https://www.sidn.nl/a/internet-security/ssl-now-essential-for-every-website-and-webshop?language_id=2

customers such as 'cheap', 'sale' and 'low prices'. If a certain amount of words matched those on the list, the website would be marked as a fake webshop. Unfortunately, this method was not fail-proof, as manual verification of a sample of the results indicated an estimated 5% of false positives, based on manual inspection of a subset of the data. Moreover, these detection scans were not done on a steady interval, creating an incomplete overview of the amount of fraudulent webshops over time.

For the sake of the research presented in this thesis, it was necessary to validate the domains found by the Dmap scans. To do so, a simple content-based tool, from hereon called the validation tool, is created to detect fraudulent webshops. This tool searches a web page's HTML for certain strings and regular expressions that often occurred in fraudulent webshops that had been manually validated. These strings could be certain sentences used in the majority of shops (e.g. 'new products for <name of month>') or certain HTML tags associated with content management systems often used by counterfeit webshops. As a large part of the marked domains had already been taken down, the required HTML content could often not be downloaded from the domain in question. To obtain HTML pages of unreachable domains, the tool queries the Internet Archive[4] and the Common Crawl[5] datasets for website captures of the domain during its lifetime. This tool was run over the set of domains marked by the Dmap scans, and a number of lists of suspicious domains posted by Consumers Associations and the Dutch Police. Domains for which no HTML source page could be retrieved were disregarded and not used as ground truth. In the end this process resulted in the creation of a list 4950 fraudulent webshops, which formed the basis of required body of known malicious webshops.

3.3.2. Benign domains

Apart from a body of fraudulent webshops, a body of benign domains is required in order to do proper classification. This benign dataset would ideally be an average representation of all domain registrations that are not used to sell counterfeit merchandise. Unlike fake webshops, there was no previously compiled list of benign domains. To obtain a set of domain registrations that were definitely not fraudulent webshops, member lists from several webshop certification programs were combined. These certification programs usually ask a fee to certify an online store and periodically check in to see if a webshop is up to its standards. Furthermore, many of these certification programs offer mediation between customer and webshop in the case of a dispute. This makes these programs undesirable to counterfeit traders, who favor to simply put a fake certification on their webstore. Comparing the list of certified webshops against the previously constructed body of malicious shops showed that there was no overlap.

Unfortunately this set of domains was rather small. To increase the amount of benign domains, the lack of HTTPS used by fake webshops was leveraged. While manually validating malicious webshops, it was observed that rather few fraudulent webshops use HTTPS. Those that did, used self signed certificates, or free certificates from Let's Encrypt and Comodo. By querying Dmap for websites with certificates from other certificate authorities, a list of legitimate domains was constructed. While this method may not exactly yield an subset of benign domains that resembles a random sample of the population of registered domains, the dataset is free of fraudulent webshops, making it suitable to use for a classification task. Furthermore, this method can be used to yield a reasonable amount of new benign domain registrations every day, as opposed to webshop certification lists, who gain new members at a much slower rate.

3.4. Overview

This chapter presented an overview of the role of SIDN as a TLD registry within the Domain Name System. Furthermore, it gave an overview of the datasets available at SIDN and discussed briefly why the data within might be useful to detect malicious registrations. The available datasets have been summarized in table 3.1. Finally, a method to construct an initial ground truth was discussed, along with the introduction of a validation tool used to automatically identify fake webshops based on their HTML content. This tool was needed as the current method used by SIDN's DMAP platform yielded

[4] https://archive.org/about/
[5] https://commoncrawl.org/

false positives which would taint the ground truth. Section 4 will discuss the features extracted from the available datasets and the machine learning methods used to train a model on the ground truth.

Dataset	Contents
SIDN registration database	• Personal information of registrants • Registrar used for registration • Timestamps of domain name registration, expiration, and release from quarantine.
SIDN NS record database	• History of NS records for every domain name registration.
OpenINTEL	• Historic measurement of DNS records for all domains in the .nl zone.

Table 3.1: Available datasets at SIDN and their contents/usage.

4

Methodology

To be able to flag domain abuse in a proactive fashion, a detection system was designed to classify new domain registrations as either benign or malicious. The detection system is called PaDAWaNS: Proactive Domain Abuse Warning and Notification System. The system uses the available datasets discussed in the previous chapter. Domains classified as malicious by the system are observed to confirm the abuse, whereupon confirmed abusive registrations are added to the training data of the detection system, enabling the system to maintain an accurate prediction model. This chapter will start with a presentation of the features that are fed to the detection tool. Afterwards, an overall design of the classification tool, explaining the operation of the system in detail. Finally, section 4.3 will discuss the classifier used by the detection tool, as well as the two candidate machine learning algorithms used by the tool to perform the classification tasks.

4.1. Features and data collection

The detection tool relies on the data of previous registrations to decide whether new registrations exhibit the same pattern of abuse. This section sets apart exactly what features are being used by the tool's classifier. The collection process for the data used to create these features is described in section 3. The features can be split up into two categories: registrant features and infrastructure features. The former consists of features describing the personal information used to create a registrant identity, as well as the moment of registration. The latter category contains features that describe the infrastructure used to register and host the abused domain name. An overview of the features is given in table 4.1. This division of features is also used in the detection tools classifier, as is further explained in section 4.3. The remainder of this section will explain how the features are extracted from the data.

Registrant Features	**Infrastructure Features**
Registrant name	Registrar
Email user	IP address
Email Provider	24-bit subnet
Email TLD	Autonomous System
Phone number	Nameserver domain name
Phone prefix	
Registrant street name	
Registrant city	
Registrant country	
History	
Hour of registration	
Day of week	

Table 4.1: Features used by the detection tool.

Registrant features

Most of the registrant features are fairly straight forward and require little to no processing. The registrant's name, phone number, city and country can be taken from the data as is. The registrant's street name is acquired by stripping the street address field of any digits and punctuation. The digits are dropped to remove the house number from the address, whereas punctuation is removed to ensure abbreviations with and without punctuation result in the same street name. For example 'Example Str. 34' and 'Example Str 32' both result in the street name 'Example Str'. This is done because exploration of known abusive registrations has shown that street names were often reused, albeit with a different house number. The phone prefix feature consists of the country specific prefix of a phone number (e.g. +31 for the Netherlands). This prefix is extracted from the registrant's phone number. This feature can be useful when a malicious actor has a preference for phone numbers of a certain nationality, for example because those numbers are easy to obtain for said actor. The registrant's email address is split up in 3 features: the email user, the email provider and the top level domain (TLD) used for the address. For example, the email address 'user@provider.tld' would give the aforementioned features the respective values of 'user', 'provider', and 'tld'.

The history of a domain registration is a categorical feature based on previous usage of the registered domain name. The feature can have three possible values: drop-catch, reregistration, or new. Drop-catch is used to indicate that the domain was recently abandoned. That is, the registered domain name has only recently become available again for registration. Based on an exploratory study of the dataset, domains registered within a month of their release from quarantine are labeled as drop-catch registrations. Domains that are registered after this time are labeled as a reregistration, whereas domains registered for the first time are labeled as new.

Finally, to capture the preferred moment of registration of malicious actors, the hour of day and the day of the week are added as categorical features. While it is possible to represent the hour of day as a discrete feature, doing so would falsely represent the feature as a linear range of values. One could try to capture the cyclical nature of time by creating a sine and cosine feature based on the time of day. However, for this proof of concept, the time of day is treated as a categorical feature.

In fact, all features described so far are categorical features. Unfortunately, the machine learning algorithms discussed in section 4.3 are not able to directly handle categorical data. Thus, it is necessary to transform the featureset into a representation that can be used by the algorithms. This transformation is achieved through one-hot encoding. One-hot encoding transforms every unique value of a categorical feature into a binary feature of its own. The newly created binary feature has a value of 1 for observations that belong to the category it represents, and is 0 for all other observations. For example, the 'History' feature can have 3 values: Drop-catch, reregistration or new. One-hot encoding this feature will replace it with 3 binary features: History_Drop-catch, History_reregistration and History_new. Registrations for which the History feature had the value 'Drop-catch' will now have a value of 1 for the 'History_Drop-catch' feature, where other registrations will have a value of 0 for this feature.

Infrastructure features

The infrastructure features are selected to be able to properly profile the infrastructure used by malicious webshops. To this end, several features related to the hosting of a domain have been selected: nameserver name, IP address, 24-bit subnet, and the autonomous system. The registrar used to register a domain does not require any further processing and can be used directly as a feature. Choice of registrar is expected to be an important feature, as it is likely that malicious actors favor certain registrars over others. This preference might be due to low costs and/or a lacks response to abuse reports, maximizing the value of a registration. The authoritative nameserver used by fraudulent webshops can be used to learn whether or not the criminals behind them prefer certain nameservers. As the power to suspend a domain name remains at the registrar, it is expected that criminals will simply use the nameserver provided by the registrar. To represent the authoritative nameserver used to host a domain name, the second level domain name is extracted from the data. This means that a nameserver with the name 'ns1.example.nl' results in a nameserver feature with the value 'example.nl'. This is done because many nameserver operators use multiple subdomains to host multiple redundant servers. To prevent creating multiple values for domains hosted on essentially the same name server, the subdo-

main is disregarded.

It is expected that actors involved in domain name abuse reuse parts of their infrastructure. To capitalize on this behavior IP addresses are used as a feature, making it possible to detect malicious domains hosted at the same IP address. By only looking at full IP addresses it is only possible to detect direct reuse of addresses. This is useful when a malicious actor hosts multiple domains on a virtual private server, causing all domains to have the same IP address. However, it is also possible that an actor hosts domains using a shared hosting service at a hosting provider. In this case, it is likely that the abused domains do not share a single IP, but use multiple IP addresses part of the network of the hosting provider. To account for this, two coarser hosting features are introduced, the 24-bit subnet and the autonomous system to which the IP address of a domain belongs. The 24-bit subnet of an IP address is the network represented by the first 24 bits, or first 3 octets, of the IP address. For example, IP address 192.168.0.13 falls in the 24-bit subnet 192.168.0.0/24. In the case of the detection tool, the 24-bit subnet is obtained by removing the last octet from the IP address, resulting in '192.168.0' for the aforementioned example. The autonomous system (AS) of an IP address can be obtained by querying WHOIS databases for the relevant IP address. However, IP addresses can be reassigned to another AS over time: the moment of querying affects the returned AS. To find the AS the IP belonged to at the moment of observation, the AS is retrieved from the openINTEL dataset, the same source of information from which the IP address is fetched. The openINTEL project uses CAIDA's Prefix-to-AS dataset[1] as a source for their mapping of IP address to AS[2]. This dataset is actively maintained and updated, and enables openINTEL to have an accurate mapping of IP and AS over time.

Of course, infrastructure of a domain is not rigid: nameservers and host addresses can change over time. As such, there is often not a single value per registration for features such as nameserver and IP address. As the tool is meant to detect abusive registration in the period directly after registration, an easy way to deal with this problem would be to take the first observed value and discard all successive observations. However, this approach would likely yield bad results, as the first observed values would likely be related to the registrar. For example, many registrars initially add a NS record for newly registered domain names pointing to their own nameservers. Similarly, many registrars point the newly registered domain name to an IP hosting a standard welcome page, prompting the registrant to log in to configure their newly purchased domain. Using only the first observation for these features would thus result in having many features that all capture the same property of the infrastructure: the choice of registrar. One could try to avoid this scenario by maintaining lists of nameservers and IP addresses known to be used for the purpose just described. These lists can then be used to filter the observed infrastructure features, and only use the first observation not on such a list, if any exist. However, maintaining such lists is a labor intensive task and would scale badly. Instead of trying to limit the hosting features to a single value, these features are represented as a set of observed values. This means that initial hosting observations related to registrars are part of the featureset. However, these features are shared between both malicious and benign registrations of said registrar. A good machine learning algorithm will assign little to no decisive power to these values, as they are not characteristic for only the benign or malicious class. Certainly, machine learning algorithms used for this classification task need to be able to deal with some amount of noise.

Just as with the registrant features, the infrastructure features consists of categorical features. However, the hosting features described in this section differ from the rest of the features in that they can have multiple values (e.g. multiple observed IP addresses), instead of a single categorical value per observation. That is, these hosting features are represensted as a list of non-exclusive categorical features, whereas the other features are exclusive categorical features. Transforming these non-exclusive categorical features using one-hot encoding is still fairly trivial: A new binary feature is created for every unique value in the union of all values of a feature. This newly created binary feature is set to 1 for observations that have the values represented in its list, and 0 for all other observations. Through this transformation, the data is expanded into a set of binary features that can be represented as a sparse matrix of l rows and n columns, where l is the amount of observations and n the amount of features after the transformation.

[1] `https://www.caida.org/data/routing/routeviews-prefix2as.xml`
[2] As stated in `https://openintel.nl/background/dictionary/`

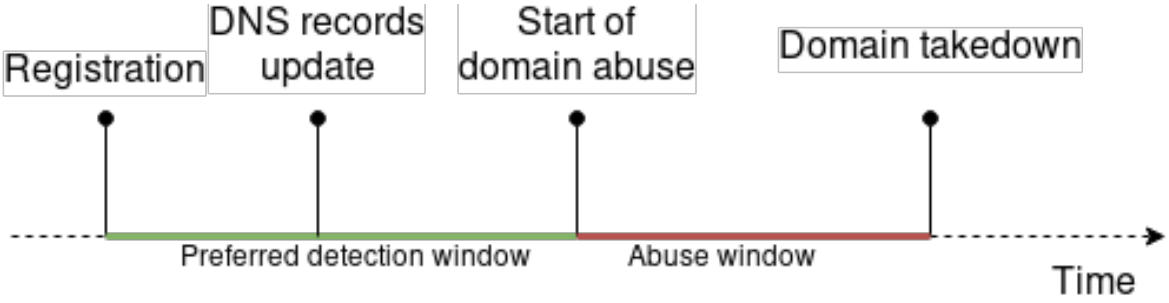


Figure 4.1: Timeline with important events of a abusive domain name registration.

4.2. PaDAWaNS design

To detect malicious registrations, a detection system has been designed. The aim of the detection system is to be able to detect abusive domain name registrations as soon as possible, ideally before the actual abuse takes place, as can be seen in figure 4.1. To achieve this goal, the detection system uses historical data from confirmed abuse cases in the past to classify new domain registrations as either malicious or benign.

The primary intended use of the system is to be run on a daily basis, as this is the interval at which most research datasets at SIDN are updated. In short, the system's daily operation consists of two phases: detection and validation. During the detection phase, the system trains a classifier on the ground truth obtained through the methods discussed in section 3.3, which is then used to detect abusive domain name registrations. Afterwards, during the validation phase, the the results of the detection are presented to the abuse team at SIDN, who investigate and take action against malicious domain names. At the same time, the results of the abuse team's investigation can be used to validate the labels assigned by the classifier, adding new data to the ground truth. This way, the detection tool will have new data to train on the next day, when this cycle is repeated. This section will first discuss the working of the detection system by describing the path a new domain name registration takes through the system. Afterwards, the different parameters used by the tool are discussed.

The tool divides the available data into three different pools: the training pool, the validation pool, and the classification pool. Figure 4.2 shows how a domain name registration might pass from one to the other. At the start of the detection system's cycle, all new domain registrations are added to the classification pool. Once a registration enters this pool, it is used for classification on a daily basis. A registration leaves the classification pool once it has been classified as malicious. Domains not classified as malicious remain in the classification pool for a limited amount of time, this period is called the classification period. This daily classification is necessary because not all data used by the detection tool will be available at the moment of registration. As described in section 4.1, the initial hosting data will often reflect a registrar's default parking page. To learn the actual infrastructure used by the registrant, one has to wait for an update of the DNS records (seen in figure 4.1). Registrations classified as malicious move from the classification pool to the validation pool. The length of the classification period thus represents the amount of time allocated to wait for additional information of the domain name's infrastructure.

A registration leaves the classification pool with either a malicious or a benign classification. It should be noted that a 'benign' classification is only benign in the sense that there is no relation between the body of known malicious registrations and the benign registration. Thus, a domain that is not related to fraudulent webshop registrations, but is part of another criminal operation (e.g. ransomware distribution) would still be classified as benign by the tool. Domains that leave the classification pool with a benign classification are discarded by the system. These domains were not flagged as malicious, and are no longer of interest. Registrations that receive malicious classification move to the validation pool, where they await validation.

During the validation phase, domains residing in the validation pool are checked for signs of abuse.

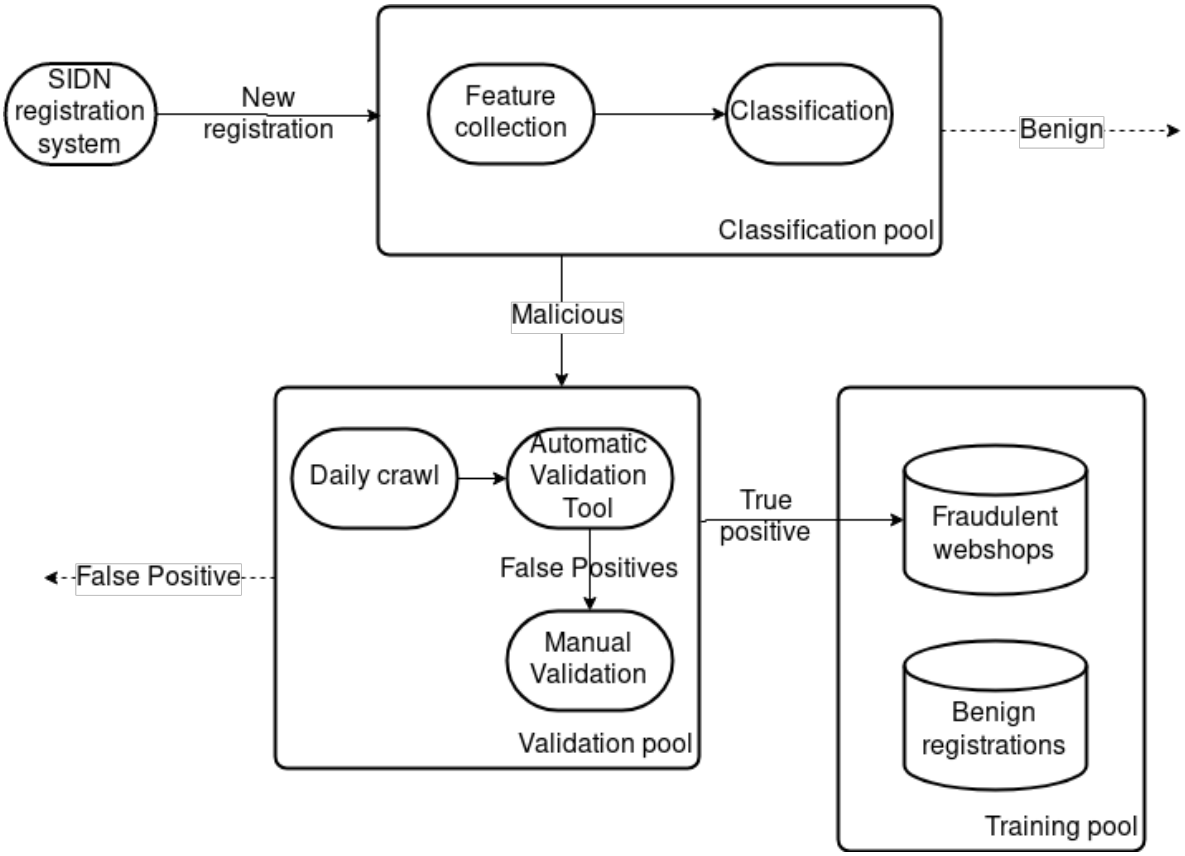


Figure 4.2: The flow of a domain name registration passing through the detection tool.

Assuming that a domain has been classified as malicious before the actual abuse has started, it is required to wait for actual signs of malice to emerge. Hence, a registration remains in the validation pool for a limited amount of time allocated to wait for such signs: the validation period. As checking domains by hand is a a tedious task, the validation tool introduced in section 3.3.1 is used to automate part of the work. All domains in the validation pool are crawled. The resulting HTML files obtained by these crawls are fed to the validation tool. Once the validation tool confirms the abuse of a domain, it is moved to the classification pool, ready to be used as training data. Domains regarded as a false positive by the validation tool can not immediately be regarded as such. As the validation tool simply checks for a static set of characteristic strings observed in the majority of fraudulent webshops, fraudulent webshops that deploy a new design might not be spotted at all. Furthermore, the fraudulent domains could deploy methods that obscure a website's content from a crawler. These methods have been encountered in the wild, as described in section 5.1. Thus, in order to ensure a proper validation, domains that are not confirmed with the validation tool will need to be checked manually. If both automatic and manual inspection observe no signs of abuse, the classification is regarded as a false positive and is disregarded by the tool.

New registrations can end up in the training pool as either malicious or benign. Malicious registrations have to be classified as malicious in the classification phase, where-after it has to be confirmed as such during the validation phase. It should be noted that registrations being classified as benign during either phase do not cause the registration to enter the training pool as benign. Instead, the ground truth gets its benign domains from the procedure described previously in section 3.3.2.

Sliding windows

The previous section described how registrations move from pool to pool through the detection system from the perspective of a registration. The validation and classification period were introduced as the time frame in which a registration is allowed to remain in their respective pools. Apart from these two periods, the system defines two additional periods: Cooldown and Training. Figure 4.3 shows how the periods relate to one another. The length of these periods are given to the detection tool as parameters. Every day, the system moves the end of the Validation and Classification period to the current date, causing these periods to mover over time as a sliding window. These periods are used to determine what data is used during the detection and validation phase.

The detection phase makes use of three sequential periods: Training, Cooldown and Classification. The classification period contains recent registrations that are being classified. The Training period defines which domains are being used during the training of the classifier. The optimal length of this period depends on the versatility of the actor behind the malicious registrations. Choosing a too high value will likely result into a model based on registration patterns that have been abandoned long since, whereas choosing a to low value will result into a model that does not fully capture the current registration strategies in play. Finally, the cooldown period serves as a buffer between the Classification and Training period. Domains registered in this period are not used by the detection system for training or testing, as as they have already been classified, but might not yet have been validated.

As can be seen, the validation period spans both the classification and cooldown period. As stated before, only domains that have been classified as malicious are subjected to the validation phase. That is, a registration will be subjected to the validation phase when:

1. the moment of registration lies within the validation period.
2. the registration has been classified as malicious.
3. the registration has not yet been confirmed to actually be malicious.

4.2.1. Simulated usage

To be able to properly evaluate the tool, it is necessary to simulate the usage of the tool. To facilitate this, the tool can be given a period to simulate the detection process. The tool will then initialize at a given start date at the beginning of the simulation period, and move its window over the data with a

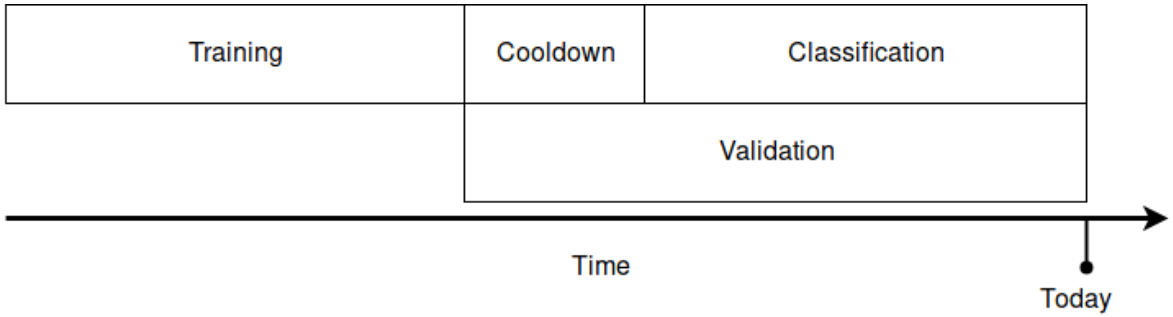


Figure 4.3: An overview of the different periods used by the detection system to select data.

given step size T_{step} on each iteration. This way, the tool can be evaluated on historic data of fake webshops. This step size does not necessarily have to be equal to a single day, because there is no need to wait for additional data. For example, moving the window with a T_{step} equal to the Testing period, each registration is classified exactly once. The results of various simulations and experiments using the tool are presented in section 5.

4.3. Classifier

The features described in section 4.1 are split into two categories: infrastructure and registrant. This dichotomy is also adopted in the design of the classifier used by the detection tool. The classifier consists of two sub-classifiers, infrastructure and registrant. A registration is only classified as malicious if it is deemed to be so by both sub-classifiers. The reasoning behind this design choice is that both sub-classifiers process different parts of domain registration behavior. The registration subclassifier will profile the typical registration details used for fraudulent registrations. This means that anyone registering using similar details (e.g. someone from a country that is often used to fake registration details) will likely be marked as malicious by this subclassifier. Similarly, the hosting infrastructure used by malicious webshops will likely also be used by perfectly legitimate domain names. Both subclassifiers will likely be biased due to the available ground truth, not fully representing a random sample of registrations. However, combining the two classifiers by requiring that both have to mark a registration as malicious will reduce the bias of the overall classifier.

4.3.1. Selection of learning methods

The problem of deciding whether a registration is malicious or not can be described as a binary decision problem. This problem is solved through supervised learning, using the established ground truth to obtain a model that can be used for classification. The machine learning algorithm used for this task needs to fulfill certain requirements. First of all, as the model has to be retrained often, a relatively short training time is required. Moreover, the available features are all nominal, meaning that the learning method must be able to correctly interpret this kind of data. In related studies, several approaches are used to classify domain names. The approaches most often used are decision trees, random forests and SVM.

A decision tree model is essentially a flow chart that can be used to classify new observations. Decision trees are learned by recursively splitting the training on a simple condition. Each split accepted by the algorithm leads to the largest reduction of impurity of the training set. The advantage of decision trees is that they are capable of directly handling nominal data, and have fast training and classification times. However, using a single decision tree is not ideal for the classification task at hand. First of all, they are inherently unstable due to the fact that relatively small variations in training data can drastically change the model of the learned tree. Moreover, learning an optimal decision tree is known to be an NP-complete problem. Because of this, real world implementations of decision tree algorithm use heuristics, which use a local optimum to find the optimal split. This in turn can lead to the generation of suboptimal decision trees.

To counter these problems and to create a more robust model, random forests are used. A random

forest is an ensemble learner that utilizes multiple decision trees. Randomness is introduced to create different decision trees, from which a majority vote is taken to classify new observations. Each decision tree is learned on a bootstrap sample of the training set. Furthermore, the algorithm does not use the full set of features to find an optimal split. Instead, a new random subset of the features is selected for each node of a tree. While this approach can lead to an increase of bias, it also reduces the variance of the model because it uses the average outcome of multiple trees. Generally, this reduction in variance leads to a better model, more than compensating for the increased bias [12].

Another proven machine learning approach is Support Vector Machines (SVM) [16]. An SVM models the testing data in an n-dimensional space and aims to find a n-1 hyperplane that will act as a decision boundary. This decision boundary is obtained by computing the hyperplane with the maximum minimum margin. That is, the biggest distance between the points closest to the decision boundary. Finding this optimal decision boundary is a computationally intensive task, causing high training times for large datasets.

A relatively new learning algorithm is the Convex Polytope Machine (CPM), introduced by Kantchelian et al [3]. Conceptually, a CPM model can be seen as a high-dimensional polytope, with the area within the polytope representing the negative class, and the area outside the polytope representing the positive class. Indeed, the polytope itself acts as a decision boundary. Just as a SVM, a CPM tries to find an optimal decision boundary in a n-dimensional space. However, instead of relying on a single hyperplane as a decision boundary, a CPM uses multiple hyperplanes that together form the model's polytope. Moreover, where a SVM tries to maximize the minimum margin of its decision boundary, a CPM tries to maximize the total margin of all points. The idea behind this strategy is that a model will have better generalizability when it has a high total margin. Computing a model using the maximum minimum margin can lead to models with a suboptimal total margin, leading to a suboptimal model. A CPM uses a stochastic gradient descent to approximate an optimal polytope, making it faster than SVMs. At the same time, CPM will likely perform better than SVM because of its focus on the total margin.

From these algorithms, both CPM and random forest were chosen to use for the classification task. CPM was chosen due to its advantages over SVM and the positive results achieved predicting malicious domain registrations in the past. [30]. Random forest was chosen due to the fact that it is a proven machine learning approach, and is capable of directly handling categorical data. For CPM, an implementation[3] by Kantchelian et al. [3] is used. For Random Forest, the `RandomForestClassifier` from scikit-learn [47] is used. The remainder of this section gives a more extensive description and mathematical definition of these two algorithms. Both approaches are evaluated in section 5.2.

4.3.2. Decision trees

Decision tree learning is an machine learning method that aims to learn a decision tree, conceptually a flow chart that consists of a series of forks. Each fork in the tree resembles a condition that determines what branch of the tree should be followed. At the endpoints of the tree, the so-called 'leaves', the tree gives a decision.

Mathematically, decision tree learning can be described as follows: Let the training data consisting of l observations with each n features be represented as an n-dimensional set of vectors $x_i \in \mathbb{R}^n$, where $i = 1, ..., l$. Additionally, let the training labels be represented as a vector $y \in \mathbb{R}^l$, where the label belonging to x_i corresponds with the i^{th} element of y. Furthermore, let m denote a node or 'branch point' and X^m the training data available at that node. A possible split θ consists of a tuple (j, t_m) where $j = 1, ..., n$ represents a feature and t_m represents a threshold. A split θ is also known as the branch condition, as X^m is divided into a group for which θ holds (X^m_{left}) and a group for which it does not (X^m_{right}). This division can be formalized as:

[3] `https://github.com/alkant/cpm`

$$X^m_{left}(\theta) = (x, y)|x_j <= t_m$$
$$X^m_{right}(\theta) = X^m \setminus X^m_{left}(\theta) \tag{4.1}$$

To find the most optimal θ at m, an impurity function $H()$ is needed. Two of these functions will be introduced shortly. In both these impurity functions, p_{mk} is defined as the proportion of observations x_i with label k in X^m:

$$p_{mk} = 1/N_m \sum_{x_i \in X^m} I(y_i = k) \tag{4.2}$$

A common impurity measure is Gini. For a node m, Gini computes the probability of an incorrect labeling if one was to assign a random observation $x_i \in X^m$ with a random label chosen from the available labels in m. This can be computed by taking the sum of probability p_{mk} of correctly labeling an observation with label k, and multiply it with the probability of incorrectly labeling the observation, $1 - p_{mk}$. Thus, Gini impurity can be defined as:

$$H(X_m) = \sum_k p_{mk}(1 - p_{mk}) \tag{4.3}$$

Another popular impurity function is information gain, which is also known as Cross-Entropy. It is based on Shanon's measure of information entropy and is defined as:

$$H(X_m) = -\sum_k p_{mk} \log(p_{mk}) \tag{4.4}$$

Both impurity functions have been widely deployed by the scientific community. A theoretical study comparing the two found that they are fairly comparable measures, only disagreeing with each other on 2% of all analyzed cases [54]. With this in mind, the detection tool uses the Gini impurity measure, as it is expected to have a better performance over Cross-Entropy, because of the the logarithmic computation required by the latter.

Having chosen a impurity function, the impurity of a split θ can be defined as the weighted impurity of the two resulting nodes:

$$G(X^m, \theta) = \frac{N^m_{left}}{N_m} H(X^m_{left}(\theta)) + \frac{N^m_{right}}{N_m} H(X^m_{right}(\theta)) \tag{4.5}$$

The optimal split can now be found by finding $\theta^* = \operatorname{argmin}_\theta G(X^m, \theta)$. By recursively applying this process at the root of the tree and all subsequently created nodes until a stop condition is met. Common stop conditions are a minimal amount of observations per node or a maximum depth of the tree.

Random forests

As stated before, a random forest is an ensemble learner that utilizes multiple decision trees. Randomness is introduced to create different decision trees, from which a majority vote is taken to classify new observations. Each decision tree is learned on a bootstrap sample of the training set. Furthermore, the algorithm does not use the full set of features to find the optimal split θ for node m in a decision tree. Instead, a new random subset of the features is selected for each node m.

4.3.3. Convex polytope machines

Convex Polytope Machine (CPM) is a relatively new learning algorithm introduced by Kantchelian et al [3]. CPM is a binary classification algorithm. As stated before, a CPM model can be seen as a high-dimensional polytope, with the area within the polytope representing the negative class, and the area outside the polytope representing the positive class. The polytope itself acts as the decision boundary.

When training a CPM model, the total margin between this boundary and the training data is maximized, to obtain a model with higher generalizability.

Generally, a high dimensional classification problem can be adequately solved with a linear classifier. This is because the chances of perfect linear separation are high for high dimensional datasets. A problem with this sentiment is that while this (near) perfect linear separation might exist, it might have a considerably small margin, leading to reduced generalizability. CPM aims to improve over existing machine learning methods by using an ensemble of SVM-like linear sub classifiers to maximize the classifier margin.

Just as decision trees, the training data can be represented as an n-dimensional set of vectors $x_i \in \mathbb{R}^n$, where $i = 1, \ldots, l$, with l and n corresponding to the amount of observations and features respectively. Again, the training labels are represented as a vector $y \in \mathbb{R}^l$. The model of a CPM can be described as a convex polytope, with each face k of the polytope representing one of the model's K linear subclassifiers. The faces of a polytope are bounding hyperplanes: supporting hyperplanes that intersect the polytope at its boundaries. A hyperplane can be represented as a linear equation of the form $w_1 x_1 + w_2 x_2 + \ldots + w_n x_n = b$, where b is a constant and there is at least one $w_i \neq 0$. A convex polytope can be represented as the set of its faces, that is, a system of linear inequalities:

$$\begin{matrix} w_{1,1}x_1 + w_{1,2}x_2 + \ldots + w_{1,n}x_n \leq b_1 \\ w_{2,1}x_1 + w_{2,2}x_2 + \ldots + w_{2,n}x_n \leq b_2 \\ \vdots \qquad\quad \vdots \qquad\qquad \vdots \\ w_{K,1}x_1 + w_{K,2}x_2 + \ldots + w_{K,n}x_n \leq b_K \end{matrix} \tag{4.6}$$

This system can be represented as a matrix inequality:

$$Wx_i \leq b \tag{4.7}$$

Where b is a vector column of m constants, $x_i \in R^n$ is an observation in the dataset, and matrix $W \in \mathbb{R}^{K \times d}$, with each row W_k corresponding to the k^{th} face of the polytope, or the k^{th} subclassifier. Furthermore, the value of $W_{k,i}$ represents the weight that sub-classifier k has assigned to feature i.

CPM is a binary classifier, and differentiates between an positive and a negative class. The labels used by CPM reflect this, as $y \in -1, 1$. An observation x_i gets classified as positive when it lies outside the polytope. As such, the classifier labels an observation x_n as positive when it fails to satisfy $Wx_i \leq b$. The decision function for a CPM model can thus be defined as:

$$f(x_i) = \max_{1 \leq k \leq K} W_k x_i - b_k > 0 \tag{4.8}$$

Indeed, CPM uses the highest score from its ensemble of sub-classifiers to assign a label to an observation. The sub-classifier giving the highest score for x_i is called the 'maximum sub-classifier' $z(x_i)$:

$$z(x_i) = \operatorname*{argmax}_{1 \leq k \leq K} (Wx_i)_k \tag{4.9}$$

CPM learns its model by trying to learn a model with a maximal margin on the training set. The margin δ_{W_k} of a single sub-classifier k can be calculated as:

$$\delta_{W_k} = \min_{1 \leq i \leq n} y_i W_k x_i \tag{4.10}$$

That is, the lowest margin of an observation for subclassifier k, or geometrically, the point with the lowest distance to the inner halfspace created by the k^{th} face of the polytope. However, the margin of a subclassifier k will only have an impact on the points for which it is the maximum sub-classifier, that is, observations x_i for which it holds that $z(x_i) = k$. Thus, the significant margin of k can be defined as:

$$\delta_{W_k}^S = \min_{i : z(x_i) = k} y_i W_k x_i \tag{4.11}$$

By only taking the significant margin of each sub-classifier in consideration, one can determine the total margin of the model:

$$\delta_W^S = \sum_{k=1}^{K} \min_{i:z(x_i)=k} y_i W_k x_i \tag{4.12}$$

CPM tries to learn an model with a maximum δ_W^S, maximizing the total margin of the polytope in respect to the given training data. It aims to find this model using a stochastic gradient descent (SGD) approach, with additional entropy constrains to prevent yielding suboptimal local optima. For a detailed explanation of the SGD method used, as well as a deeper explanation of the optimization problem solved by CPM, the reader is directed towards the original work of Kantchelian et al [3] and its accompanying appendix[34].

5 Results

This chapter presents the results of various experiments performed using the detection tool described in section 4.2, as well as an analysis of a large body of detected fraudulent webshops. First, section 5.1 discusses how the initial ground truth established in section 3.3.1 is expanded by running the detection tool over a large amount of registrations, as well as any difficulties encountered during the manual validation of registrations marked as malicious. Afterwards, this ground truth data is used to evaluate the performance of the detection system by letting it classify a set of known benign and malicious domains. A second evaluation is performed by simulating live usage of the detection tool to classify a semester's worth of domain names. Section 5.3 demonstrates that retraining on a regular interval has a negligible effect on the performance of the model, indicating that fraudulent webshop operators use the same methods of registration for a prolonged time.

These methods are further explored in section 5.4, where an overview of indicators of abuse is presented. Afterwards, in section 5.5 the body of detected fraudulent webshops is analyzed in an effort to discover different registration campaigns. Finally, section 5.6 concludes this chapter with a presentation of evidence suggesting that most fraudulent webshops can be attributed to an organization operating from China.

5.1. Using the detection tool to expand the ground truth

By running the tool over an extended period of time, starting around April 2016 and gradually moving the window towards the present, a large number of domains marked as malicious were returned by the tool. These marked domains were then validated using the validation tool introduced in section 3.3.1. If the marked domains indeed hosted fraudulent webshops, they were added to the ground truth as malicious domains. In many cases, the validation tool simply failed to validate a domain because the hosted website was no longer available and both the Internet Archive and the Common Crawl dataset did not have a snapshot of the domain in question. Another large amount of domains directed to a parking page of their registrar, stating that the domain was now in the possession of one of their customers. Because it is impossible to make a claim concerning the content of both unreachable and parked domains, they were regarded as false positives and not added to the ground truth.

Marked domains that were not validated by the validation tool were checked manually, to see whether this domain was truly not a fraudulent webshop. During the (manual) checking of fraudulent webshops, a number of suspicious domains were encountered that initially did not seem to host a fake webshop. When visiting these domains through a browser, these domains served either no content at all, an 'advertisement' page with information about luxury goods sold by many fake webshops, or an seemingly abandoned website. Upon closer inspection of these advertisement pages, it appeared that these advertisements were not so much geared towards humans, but to search engines. The advertisement pages only linked to other advertisement pages on the same domain, never advancing to an actual webshop. The texts were either in English or in poorly translated Dutch and contained little useful information, but instead seemed to be designed to contain as many keywords (e.g. cheap,

quality, luxury) as possible. This content is probably used to improve search engine rankings.

The abandoned websites were easily recognized by their broken links, outdated textual content and often antiquated design. Upon further investigation it became clear that these 'abandoned' pages once were hosted on their respective domains by a prior registrant. In other words, these pages were not so much abandoned as that they were stolen. It appeared that these pages were downloaded from the Internet Archive[1] and put back online on their former domain. The Internet Archive adds an archival and copyright disclaimer to every archived web resource. While this disclaimer was not encountered in the HTML source code, it was encountered in many CSS and javascript resources referenced by the HTML, proving that these pages were indeed taken from the Internet Archive. It is possible that by hosting the content of the previous registrant, the domain operators hope to extend the lifetime of the domain's residual trust.

Furthermore, a majority of these stolen pages had one or two links injected to an advertisement page, similar to the ones described earlier. While these pages still used the original templates and style of the prior registrant, the content was again geared towards search engine optimization. This seems to indicate a strategy in which the domain operator tries to both keep the residual trust of a domain while simultaneously optimizing its search ranking.

As this behavior seemed to be part of a search engine optimization (SEO) campaign, the SEO tags encountered were used in search queries in various search engines, with varying results. Some did not attain a noticeable ranking for the tags they tried to optimize, while others made it to the first page of results. However, when following a search engine link to one of the suspicious domains, the server did not return a stolen or advertisement page, but a fraudulent webshop. It became apparent that the domain operators had deployed methods to only serve a webshop under specific conditions. This obfuscation method is a well known SEO technique known as cloaking. Cloaking is a technique used to present different content based on some property of the request made, such as user agent or IP address. This technique is often deployed on a website to serve a page laden with SEO content to search engine crawlers, while serving its actual content to other visitors, effectively "cloaking" this content from the search engines. To achieve this cloaking effect, black SEO campaigns often use a doorway page. Such a doorway page often uses javascript or serverside scripts in order to determine what content will be served.

In the case of these suspicious SEO domains, the vast majority of doorways used a small piece of obfuscated javascript loaded by the page's HTML, often with an inconspicuous name such as 'header.js'. After deobfuscating this script, it became clear that the webshop is only shown to users that visit the site via a search engine. The script checks the referrer attribute and if the referring domain contains the string 'google', 'bing', 'yahoo', 'aol' or 'babylon', the user is redirected to the fake webshop, hosted on another domain.

```
1 if (window.document.referrer) {
2     var se = ['google', 'bing', 'yahoo', 'aol', 'babylon']
3     for (i = 0; i < se.length; i++) {
4         if (window.document.referrer.indexOf(se[i]) > 0) {
5             top.location.href = 'http://example.nl'
6         }
7     }
8 }
```

Listing 5.1: Deobfuscated functional equivalent of a cloaking script encountered on fake webshop domains.

For a small set of suspicious domains, such javascript was not found, indicating that server-side cloaking methods might be deployed. An investigation of these domains uncovered that all of them were targeting French words to boost their rankings. To test whether the cloaking was done by means of geoIP lookups, the domains were visited through a French proxy. Indeed, all of these domains now redirected to a fraudulent webshop in the .fr zone.

[1] https://wayback.archive.org/

	True negatives	False positives	False negatives	True positives	True Positive rate	False Positive rate	Precision	Accuracy
CPM	5097	1	317	4357	0.932249	0.000131	0.999848	0.967526
RF	5097	1	188	4486	0.959777	0.000196	0.999777	0.980659

Table 5.1: Average results of three detection simulations over the first 6 months of 2018, when only considering registrations that were part of the ground truth. Results for both CPM and RF classifiers are presented. The amount of positives and negatives have been rounded towards the nearest integer.

Domains for which it was possible to circumvent deployed evasion techniques were added to the ground truth as malicious domains. Domains that were believed to have no relation to a fraudulent webshop were considered to be a false positive. However, these domains were not added to the ground truth as benign domains. This was done primarily because it is possible that some of these domains used a cloaking technique that went unnoticed during manual inspection. Moreover, many domains for which no direct relation to fraudulent webshops could be made, still did exhibit signs of suspicious behavior. For example, for some domains it was easily proven that their content was stolen from the Internet Archive, but no direct link towards a fraudulent shop was observed. Other domains did host a webshop, but sold products that were unrelated to counterfeit luxury goods, such as male enhancement products, hormones and industrial components. While these domains seemed to have no direct relation to targeted 'fake' webshops, their legality remained questionable at best.

5.2. Detection performance

As described previously, the ground truth used by the detection tool comes from three different sources. The source for benign domains is Dmap, which is used to retrieve all domain registrations for which a non-free HTTPS certificate was used. The original source for domain names hosting fraudulent webshops also came from Dmap, scanning HTML headers for the presence of a predetermined list of words. The output of both sources was checked for incorrect labels using the validation tool written for this thesis project. Finally, the detection tool described in section 4.2 was used on the body of data to uncover more fake webshops, which were subsequently added to the ground truth. The combination of these three sources result in the ground truth used to evaluate the detection tool's performance. It should be noted that this ground truth is not necessarily a correct representation of the space of all domain name registrations. After all, both the benign and the malicious sets of ground truth data are selected from a subset of all registrations. A completely representative ground truth could be obtained by taking a random sample of registrations, and establishing a ground truth for this set. However, this would be rather costly, as such a sample would need to be of considerable size to contain enough malicious webshops. After all, the percentage of malicious webshops in all domain name registrations is estimated to be very small. Moreover, as discussed section 5.1, many domains are in a 'grey zone', for which it is unclear whether they are actually benign or malicious.

To show the performance of the classification tool, an experiment is performed. The tool is run in simulation mode for all registrations in the ground truth data in the first 6 months of 2018. This simulation is performed with both machine learning algorithms. The results of these simulations are shown in table 5.1. As can be seen, both learning algorithms have roughly the same performance: little to no false positives, and a small amount of false negatives.

Using the trained classification models, it is possible to see what features have a large impact on the decision process. To obtain the feature importance of a feature in a decision tree, one can compute the Gini importance of all features in the tree. As discussed in section 4.3.2, decision trees are computed by finding a split that minimizes an impurity metric for the two resulting nodes. The Gini importance of a feature is the mean decrease in node impurity that can be attributed to that feature. To compute the feature importance of a feature in a random forest model, one simply takes the Gini importance averaged over all trees in the ensemble. Table 5.2 shows the Gini importances averaged over all random forest models used for the experiment. As can be seen, the History, Country of residence and

Registration features	Importance	Infrastructure features	Importance
History	0.201589	Nameserver domain name	0.294041
Country of residence	0.195214	Registrar	0.201627
Phone prefix	0.186386	Country of hosting	0.178155
Mail provider	0.080033	Autonomous System	0.167396
Hour of day	0.057836	24-bit subnet	0.110017
Mail TLD	0.054154	IP address	0.048762
City	0.052774		
Mail user	0.043289		
Phone number	0.041826		
Street	0.040968		
Registrant name	0.040786		
Weekday	0.005144		

Table 5.2: Average Gini importance of the features for the registrant and infrastructure classifier.

phone prefix seem to be important features for the registration classifier. The choice of mail provider also seems important, but has a relatively low importance score. The infrastructure features all seem to contribute a fair share towards the classifiers decision model, with the exception of IP addresses, which is likely a too specific feature.

5.2.1. Running in the wild

Now, instead of only considering the established ground truth, this simulation is evaluated on all new registrations. The outcome of this evaluation is shown in table 5.4. As can be seen, the small amount of false positives changes considerably when the tool is used in the wild. However, it should be noted that this table represents the strict validation norms applied for the creation of the ground truth. A false positive in this table represents a domain for which it was not possible to obtain irrefutable proof of a fraudulent webshop. The false positives all have been investigated manually, the result of this investigation can be seen in table 5.3. The false positives can be divided into 5 categories, which are described below:

Category	Amount
Likely illegitimate shops	124
Suspicious	56
Known abusers	151
No content/information	275
WHOIS protection	33
Unrelated	86

Table 5.3: Results of investigating false positives from the CPM classifier.

Likely illegitimate shops As can be seen, webshops were encountered on a number of false positive domains. These webshops did not had direct similarities with the type of fake webshops that are the focus of this thesis. However, the content of these shops brought their legitimacy in question. For example, instead of advertising their fake products as the real thing, some of these domains openly admitted that they sold replicas of popular brands. While this informs the consumer, this practice is still considered illegal as it is considered to be a violation of intellectual property rights. Other webshops did not sell luxury goods, but sold male enhancement products, hormonal supplements or diet pills. None of these webshops gave clear information about the origin of their goods, nor the identity of their owners.

Suspicious As described in section 5.1, fraudulent webshops have been observed to reupload a domain name's previous content using the Internet Archive. Afterwards, these domains are injected with SEO tags and texts. A so-called doorway mechanism is used to differentiate between crawlers and a normal user, serving a fraudulent webshop to the latter group. A small amount of domain names did not host a webshop, but did serve websites stolen from the Internet Archive, often injected with text geared towards search engine optimization. It is possible that the doorway mechanism requires a specific condition to show a webshop, or that the doorway mechanism is broken or missing. Either way, the practice of hosting SEO injected content from a web archive is deemed suspicious on its own.

	Negative	False negative*	False positives	True positives	Precision
CPM	423751	292	725	4360	0.857424
RF	423315	159	1161	4493	0.794659

Table 5.4: Results of running the detection tool over the first 6 months of 2018. Note that the amount of False negatives represent a lower bound, as these are only the false negatives known from the ground truth.

Known abusers Some domains have not been observed to host any content during their lifetime, but were registered by registrants who have been observed to register multiple fraudulent webshops in the past. A possible explanation as to why the new registrations were not abused is that they have been suspended by their registrar. This theory is supported by the fact that all registrations by such a registrant were terminated around the same time, indicating a possible suspension.

No content/information Many false positive domains were never observed to host any content other than a default parking page. While many of these registrations were made with using contact details that closely match known abusive registrations, no direct relation to a fraudulent webshop can be made. Similarly, it is not possible to confidently say that these domains are completely unrelated.

Unrelated Finally, for a relatively small amount of registrations it became clear from the hosted content that there was no relation with fraudulent webshops.

5.3. Performance stability

To see how much the detection tool is dependent on regular retraining to detect fraudulent webshops, the detection tool is ran twice on the new registrations of the first 6 months of 2018. The first time, the parameters used for previous experiments are used: A training period of 6 weeks, cooldown period of 1 week and a classification period of 1 week, meaning that the classifier updates its model every week. The second run uses the same parameters as the first run. However, this time the model is not retrained on regular intervals. Instead, the entire 6 months' worth of registrations is classified using the model learned in the first cycle. The experiment is evaluated using the ground truth data. This experiment indicates at what rate registrants of fake webshops change their strategies. It is expected that an actor that changes strategies often will quickly evade detection when the classifier does not regularly retrain its model. Similarly, actors that keep reusing the same strategies are expected to still be detected fairly well. As is shown in this experiment, fraudulent webshop operators seem to fall in the latter category. Figure 5.1 displays the total amount of false positives and false negatives per week. As can be seen, the performance of the classification only seems to degrade after 5 months, indicating that fraudulent webshop operators keep using the same strategies observed at the start of the year. This is to be expected, as currently there is no real incentive for them to switch strategies. Current mitigation methods are reactive, and abusive registrants usually have enough time to make a profit from their domains before they are taken down. Overall, not retraining the model leads to a 3% decrease of the true positive ratio.

5.4. Exploring fake webshops

Running the detection tool using data for the period between April 2016 until the end of Juli 2018 uncovered a large amount of fraudulent webshops. More than 30.000 domain registrations have been identified as fraudulent webshops. This section explores the total set of malicious registrations by going over a number of statistical trends, discussing their relation to the detection tool and possible reasons for the existence of these trends. To comply with privacy and confidentiality regulations, registrar names have been pseudonimized and IP addresses have had their final two octets removed in the public version of this thesis.

5.4.1. Infrastructure

An overview of registrars used to register domain names for fraudulent webshops is given in table 5.5. As can be seen, more than 90% of all observed fake webshops are registered using one of five reg-

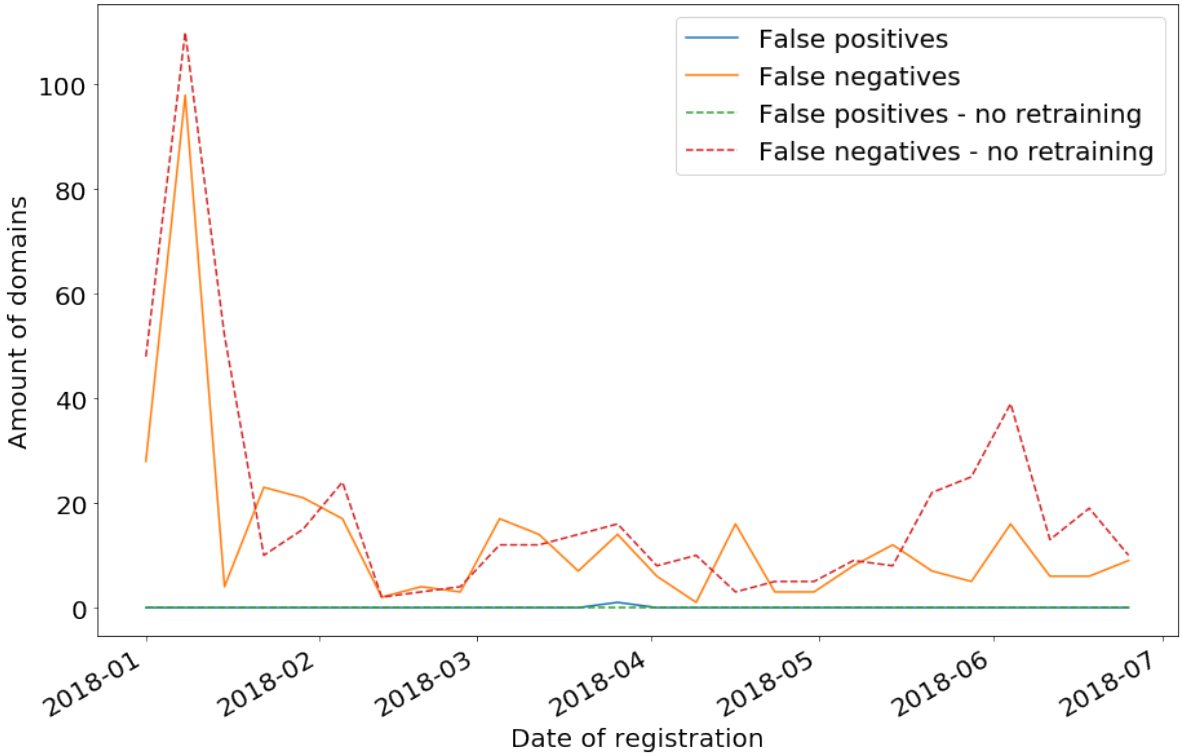


Figure 5.1: False positives and false negatives over time for classification with weekly retraining (continuous lines) and without any retraining (dashed lines).

istrars. This in line with prior research by SIDN, in which a relatively small group of registrars was observed to be responsible for the majority of malicious registrations [44]. At the same time, these registrars are only responsible for less than 10% of all registrations. It is likely that these registrars are used either due to cheap prizes, preferable payment options, slow response to abuse reports, or a combination of these perks. Some of these registrars are rather small, managing few registrations in the .nl zone. For one of these registrars, more then half of all registrations are fraudulent webshops. Other more established registrars have a considerable amount of registrations, allowing the malicious registrations to “hide’ themselves in the crowd. Still, this preference for a handful of registrars can be leveraged to detect new fraudulent webshops.

A newly registered domain name often has its name servers set to those of the registrar. As such, there are clear connections between some popular name servers and popular registrars, as can be seen by comparing table 5.5 with table 5.6. For example, Registrar A and Registrar C both have comparable quantities of domains registered and domains for which they supply name servers. After registration, the owner of a domain can choose to use the name servers from the registrar to operate

Registrar	Fake webshops	Total registrations	Fake webshops per registration
Registrar A	10491 (35.82%)	109310 (5.3%)	0.1
Registrar B	6643 (22.68%)	12128 (0.59%)	0.55
Registrar C	4524 (15.45%)	14855 (0.72%)	0.3
Registrar D	3528 (12.05%)	39931 (1.94%)	0.09
Registrar E	2564 (8.75%)	7047 (0.34%)	0.36
Registrar F	525 (1.79%)	14583 (0.71%)	0.04
Registrar G	482 (1.65%)	3853 (0.19%)	0.13
Registrar H	261 (0.89%)	24719 (1.2%)	0.01

Table 5.5: Registrars with over 100 fake webshop registrations.

Nameserver	Fake webshops	Total registrations	Fake webshops per registration
Registrar A NS1	13600 (46.43%)	64033 (3.11%)	0.21
Registrar A NS2	8918 (30.45%)	13105 (0.64%)	0.68
Registrar C NS	4239 (14.47%)	11580 (0.56%)	0.37
Registrar B NS	4001 (13.66%)	5601 (0.27%)	0.71
dnspod.net	3597 (12.28%)	10292 (0.5%)	0.35
Registrar D NS	3529 (12.05%)	41618 (2.02%)	0.08
2kz.net	3497 (11.94%)	5002 (0.24%)	0.7
orderbox-dns.com	3034 (10.36%)	3825 (0.19%)	0.79
Registar E NS	931 (3.18%)	1657 (0.08%)	0.56
alidns.com	676 (2.31%)	1197 (0.06%)	0.56
cloudflare.com	648 (2.21%)	40964 (1.99%)	0.02
bodis.com	611 (2.09%)	2180 (0.11%)	0.28
hostgou.com	581 (1.98%)	1084 (0.05%)	0.54

Table 5.6: Nameservers associated with more than 500 fraudulent webshops.

Autonomous System	Fake webshops	Total registrations	Fake webshops per registration
48635	8683 (30.0%)	97289 (5.13%)	0.09
197328	4992 (17.25%)	8381 (0.44%)	0.6
64435	3804 (13.14%)	5185 (0.27%)	0.73
26496	3212 (11.1%)	35465 (1.87%)	0.09
29073	3054 (10.55%)	5743 (0.3%)	0.53
59447	2899 (10.02%)	4970 (0.26%)	0.58
57858	2591 (8.95%)	4401 (0.23%)	0.59
18779	1753 (6.06%)	3030 (0.16%)	0.58
21740	1605 (5.55%)	2974 (0.16%)	0.54
26415	1599 (5.53%)	3388 (0.18%)	0.47
63119	1224 (4.23%)	2158 (0.11%)	0.57
8082	1176 (4.06%)	2093 (0.11%)	0.56
41204	1153 (3.98%)	1295 (0.07%)	0.89
12327	1128 (3.9%)	1743 (0.09%)	0.65

Table 5.7: Autonomous systems observed hosting over 1000 fraudulent webshops.

their domain, or switch to name servers of a different party (e.g. a hosting company).

The amount of fraudulent webshops using Registrar A's name servers is larger than the amount of fraudulent shops registered through them. This is because a large number of fraudulent domain names was later transfered to Registrar A, after initially being registered through another registrar. These domain names were all transfered within the same week. It is possible that these domain names were bought by counterfeit webshop operators, and transfered to their preferred registrar. Another possible explanation could be a reseller that decided to move his business to another registrant.

When looking at the distribution of hosting addresses for fraudulent webshops, it appears from table 5.7 that the majority of fake webshops hosts are concentrated to a relative small group of autonomous systems. Furthermore, some autonomous systems host more fraudulent webshops than legitimate websites in the .nl domain space. Given that only fake webshops for which definite proof could be found are used for these statistics, thus representing a lower bound, one could argue that some of the autonomous systems in table 5.7 are only used for illegitimate business. The hosting providers using these autonomous systems are likely either cheap or 'bullet-proof', making them an attractive option for criminals. The reader should keep in mind that a domain can been associated with multiple IP addresses, autonomous systems and nameservers so percentages in table 5.7 and table 5.6 will not sum to 100.

IP address	Fake webshops	Total registrations	Fake webshops per registration
185.87.x.x	8753 (29.88%)	12802 (0.67%)	0.68
127.0.0.1	6475 (22.11%)	15233 (0.8%)	0.43
199.59.x.x	285 (0.97%)	15360 (0.81%)	0.02
199.59.x.x	247 (0.84%)	5279 (0.28%)	0.05
69.64.x.x	204 (0.7%)	1001 (0.05%)	0.2
54.72.x.x	129 (0.44%)	31985 (1.69%)	0.0
69.25.x.x	117 (0.4%)	633 (0.03%)	0.18
144.76.x.x	110 (0.38%)	119 (0.01%)	0.92
144.76.x.x	110 (0.38%)	119 (0.01%)	0.92
50.63.x.x	95 (0.32%)	6486 (0.34%)	0.01

Table 5.8: 10 IP addresses obtained by resolving fraudulent domains.

When looking at infrastructure reuse on the level of individual IP addresses, it becomes apparent that some IPs are reused often and some are used to host almost exclusively malicious .nl domains. Table 5.8 shows two IP addresses with a disproportional amount of fake webshops associated with them: 127.0.0.1 and 185.87.x.x. The former of these two is reserved for loopback purposes, and is not routable over the Internet. Domain names pointing to this IP address are rare in the overall .nl population, while having a significant share among the fake webshops registered within the the .nl zone. When investigating this anomaly, it became apparent that the vast majority of the fraudulent webshops that had an A record pointing to 127.0.0.1 did so right after registration. This ruled out the possibility that this observation was simply the result of domain name take-downs by registrars. Later, the A record would be updated to a routable IP address, indicating that the initial 'localhost' A record might be a default setting of certain registrars. Indeed, further investigation showed that nearly all of these malicious domains were registered through Registrar A. However, these domains made only a small subset of the total amount of domains registered with Registrar A. The majority of domains registered through Registrar A did not exhibit this behavior. It could be speculated that the initial 127.0.0.1 A record is an indicator for a specific domain name reseller that uses Registrar A as its registrar. Unfortunately, information about resellers is not available at SIDN, making it hard to verify this theory. Still, the use of 127.0.0.1 A records seems to be part of the process some counterfeit traders use to register new domains.

The other IP address standing out, 185.87.x.x, seems to be a notification page from Registrar A indicating that the domain has expired. Registrar A states that this page is used in the event of a so called "soft-quarantine": A domain name for which the customer no longer has a valid lease, but Registrar A still has the option to renew the domain. However, only around 10% of the roughly 100.000 domains registered through Registrar A are associated with 185.87.x.x. Of those 10%, two thirds are malicious webshops, indicating that this A record might also be used when Registrar A takes down domains due to a violation of their terms of service. Indeed, for the vast majority of domains that are associated with this IP, it was the last A record they held before expiration. As a result this IP address is ill-suited to detect counterfeit webshops in an early stage.

5.4.2. Temporal features

Aside from the infrastructure used to register and host a website, another interesting feature might be the moment of registration. As prior research was able to detect business day-like patterns in malicious registrations [68], analyzing the moment of registration might uncover useful patterns. From figure 5.2, it is apparent that a share of fraudulent webshops are registered during off-peak hours. More specifically, it seems that most fraudulent domains are registered from 0:00 AM till 12:00 AM, while overall registrations peak between 8:00 and 15:00. This insight might be useful to detect future fraudulent registrations.

As discussed in section 3.1.2, fraudulent webshop operators are likely trying to register domains that have only just been released from quarantine, performing a so called 'drop-catch'. Table 5.9 shows

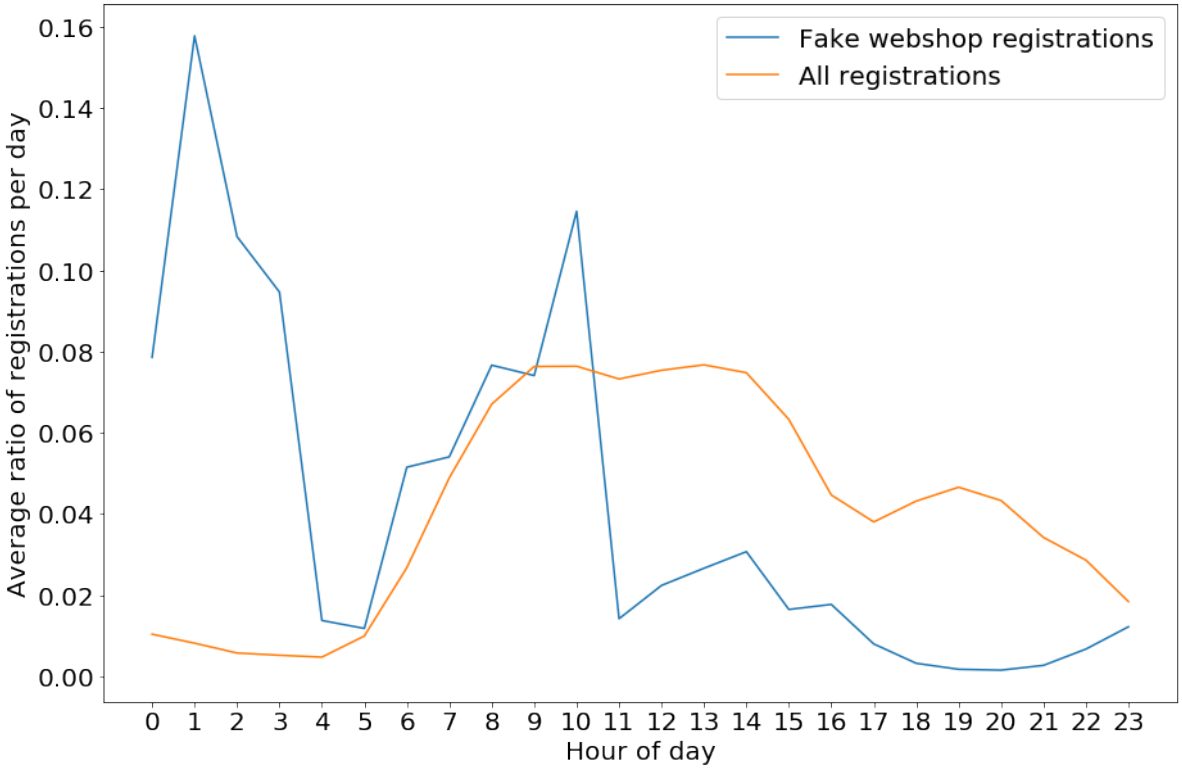


Figure 5.2: Normalized registrations per hour of fake webshops and all registrations.

	Fake webshops	Total registrations	Fake webshops per registration
drop-catch	26257 (89.26%)	205521 (9.93%)	0.13
re-registration	1815 (6.17%)	304500 (14.71%)	0.01
new	1343 (4.57%)	1560319 (75.37%)	0.0

Table 5.9: Distribution of drop-catch (within 31 days of expiry), new, and re-registrations.

that this tactic is indeed employed as over 95% of fake webshops reuse a domain name. The drop-catch feature makes a distinction between drop-catch registrations and re-registrations. A drop-catch domain is registered within 31 days after release from quarantine. A reregistration domain is registered after this period. Figure 5.3 shows that this boundary of 31 days was chosen correctly, as many fake webshops domain names are indeed registered within a month of becoming available.

5.4.3. Registrant features

When inspecting the (presumably fake or stolen) registrant data used for fake webshops, it appears that both email addresses and phone numbers are most often renewed. This is likely due to the fact that both of these go through some form of validation, as discussed in section 3.1.2. Table 5.10 shows the amount unique values per field that has been observed. This data shows that on average, 1.5 fake webshops are registered per registrant unique email address.

Surprisingly, almost half of all fraudulent registrations made use of German phone numbers. In fact, the statistical odds of a registration using a German phone number being a malicious webshop are similar to those for registrations using a Chinese phone number, as can be seen from table 5.11.

Field	Unique values
Email	20022
Phone number	19490
Name	17598
Street	16728
Postal code	11342
City	8506
Country	31

Table 5.10: Amount of unique values per field observed in all 29450 fake webshop registrations.

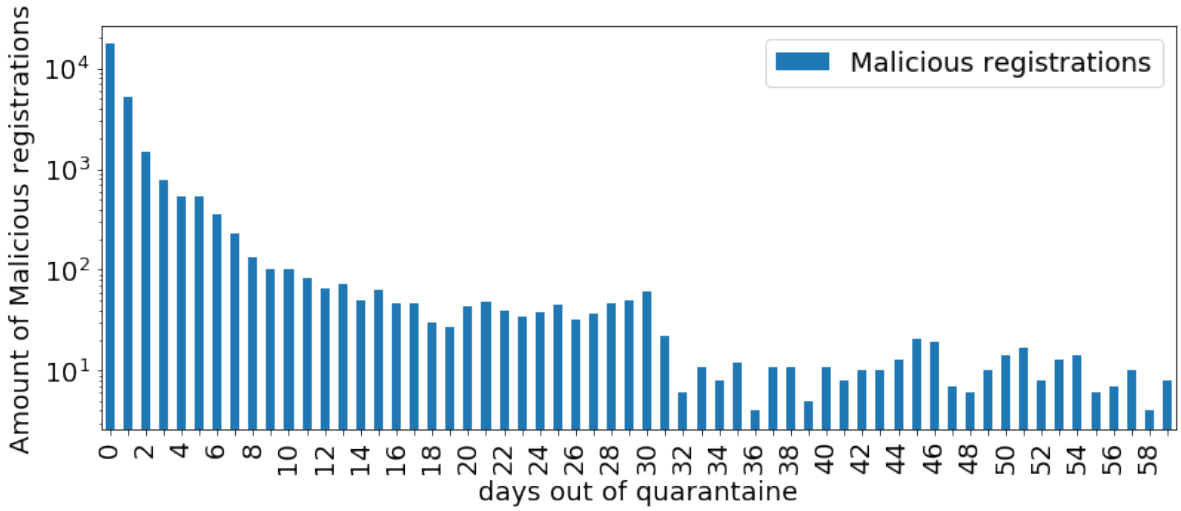


Figure 5.3: Distribution of expired time between domain expiration and abusive re-registration.

Phone country code	Fake webshops	Total registrations	Fake webshops per registration
+49 (DE)	14047 (47.75%)	36874 (1.79%)	0.38
+86 (CN)	3926 (13.35%)	10408 (0.5%)	0.38
+31 (NL)	3224 (10.96%)	1797678 (87.21%)	0.0
+1 (US)	2557 (8.69%)	53937 (2.62%)	0.05
+33 (FR)	1510 (5.13%)	11895 (0.58%)	0.13
+45 (DK)	1423 (4.84%)	4092 (0.2%)	0.35
+44 (UK)	1366 (4.64%)	13733 (0.67%)	0.1

Table 5.11: Phone country codes with over a thousand fake webshop registrations.

This table further shows that statistical odds of phone numbers from certain countries being abused are significantly higher then others, potentially making it a useful feature for detection.

As can be seen in table 5.12, over 80% of fraudulent registrations make use of just eleven mail providers. Many of these mail providers allow users to create accounts free of charge, making them attractive for malicious actors. Some free mail providers, such as gmail and outlook, are very popular under both malicious and legitimate registrations. However, many of the email providers used by fraudulent webshops are relatively unknown and obscure in the Netherlands, making the maildomain a indicator of possible abuse.

5.5. Campaign analysis

Understanding the tactics deployed for different campaigns can give more insight to engineer more features to be able to detect them. This section presents an overview of registration tactics that were encountered by investigating fraudulent webshop registrations. These tactics are used to group registrations together in campaigns, showing that they are used for a prolonged time, before being replaced. First, the process of campaign identification is described. Afterwards, identified campaigns are presented and discussed. Due to privacy and confidentiality regulations, some tables and figures containing email addresses have been omitted from the public version of this document.

5.5.1. Campaign discovery and identification

In general, abusive registrations often occur in bursts of variable size. This property can be leveraged to use as a lead for further campaign identification, as registrations contained in the same burst are very likely to be part of the same campaign. By clustering malicious domains by the moment of registration, it is possible to obtain numerous batches that can then be analyzed for recurring patterns and strategies.

Maildomain	Fake webshops	Total registrations	Fake webshops per registration
163.com (F)	8211 (27.91%)	13926 (0.68%)	0.59
yeah.net (F)	2147 (7.3%)	2928 (0.14%)	0.73
hotmail.com (F)	2124 (7.22%)	196913 (9.55%)	0.01
sina.com (F)	1893 (6.44%)	2945 (0.14%)	0.64
outlook.com (F)	1807 (6.14%)	44060 (2.14%)	0.04
privacyprotect.org	1412 (4.8%)	1863 (0.09%)	0.76
gmail.com (F)	1373 (4.67%)	467718 (22.69%)	0.0
yahoo.com (F)	1347 (4.58%)	15705 (0.76%)	0.09
scvoo.com	1289 (4.38%)	1807 (0.09%)	0.71
ptwmgs.cc	1172 (3.98%)	3643 (0.18%)	0.32
hxmail.com	1028 (3.49%)	1690 (0.08%)	0.61

Table 5.12: Maildomains used to register more than a thousand fraudulent webshops. (F) Indicates the domain is part of a free email service provider

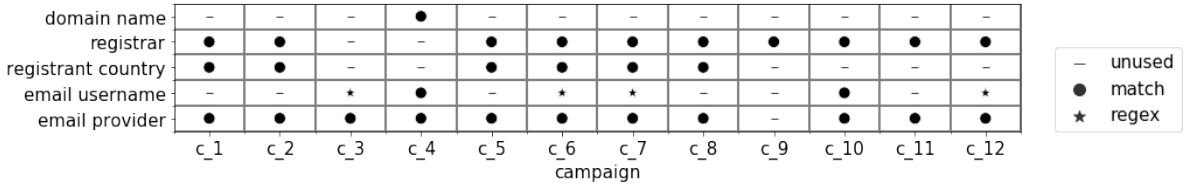


Figure 5.4: Campaigns and the criteria used for their identification.

From these strategies one can construct identification criteria, a set of conditions on different features of a registration, designed to identify domains being registered using the same strategy. These criteria are then evaluated on the dataset as a whole, to check whether they are not to narrow or to broad, but indeed correctly captures the strategy. This process resulted in the identification of 12 campaigns. The fields used to identify each campaign are shown in figure 5.4. As can be seen, the combination of email domain and registrar is often used to distinguish campaigns.

5.5.2. Observed campaigns

Some campaigns exhibited notable patterns that are relatively uncommon in legitimate domain registrations. The domain registration activity of each campaign is shown in figure 5.5. From this figure it becomes apparent that many of the identified campaigns seem to be only very active for a period of 2 to 6 months, while lingering before and after the 'active' time window. For example, campaign 'c_4'

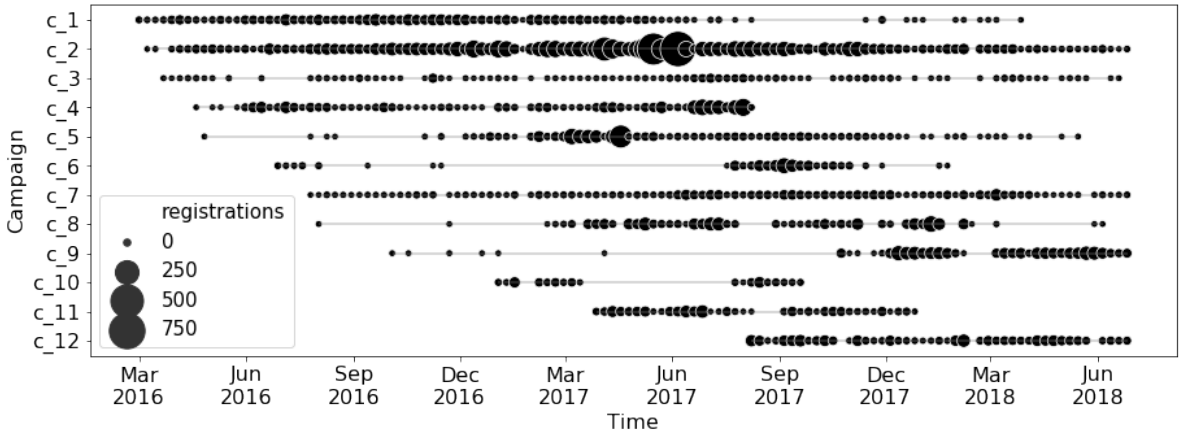
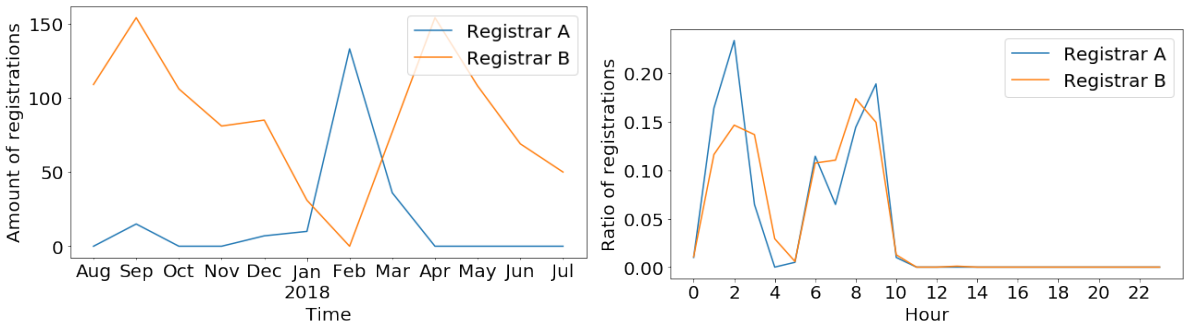


Figure 5.5: Campaign activity over time.



(a) (b)

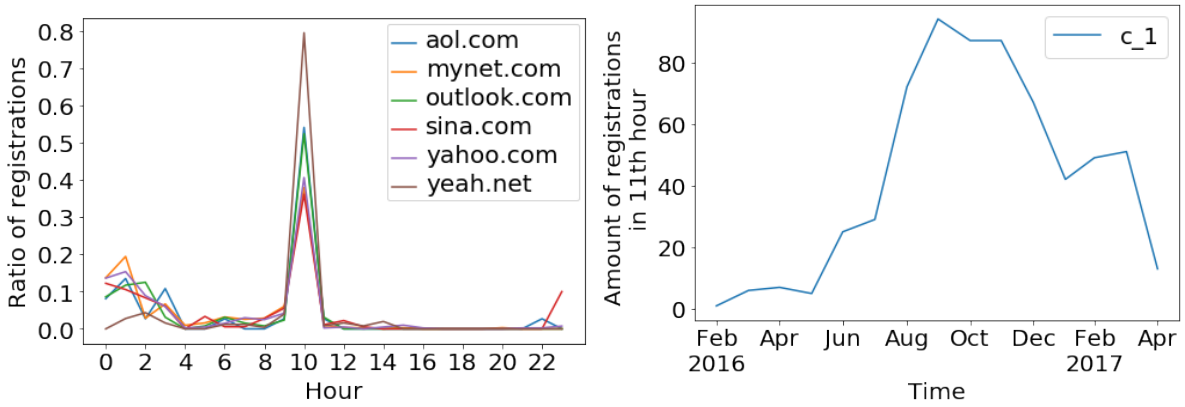
Figure 5.6: (a) Registrar usage over time for campaign c_12. (b) Hourly activity per registrar for campaign c_12.

makes use of dedicated email addresses. That is, every email address observed in this campaign seems to have been created with the express intention to register a specific domain. In many cases the email user is an exact match with the second level domain name that is being registered. Not all registrations in c_4 are an exact match. In some instances the email user is a substring of the domain name being registered, the email user is equal to the domain name with a small amount of characters appended, or a combination of these methods. In some cases the dedicated email address is also used to register other domains, either shortly before or after registering the ‘targeted’ domain.

Where c_4 displays a clear pattern in the method of email address generation, c_12 is noticeable due to its clear lack of pattern in used email addresses. This campaign uses a couple of free email suppliers popular in western countries. All email addresses seem to have been generated with a random generator, consisting of garbled alphanumeric characters. There are strong indicators that these registrations are all part of a single campaign: all registrations use addresses located in either France, Germany, the Netherlands or Spain. Moreover, all registrations are being made during the first 12 hours of the day, as can bee seen from figure 5.6. Campaign c_7 also generates random email addresses, albeit in a very consistent way. All domains are registered with an email address with a username made of 6 alphanumerical characters, all from the same email provider, making it rather easy to spot.

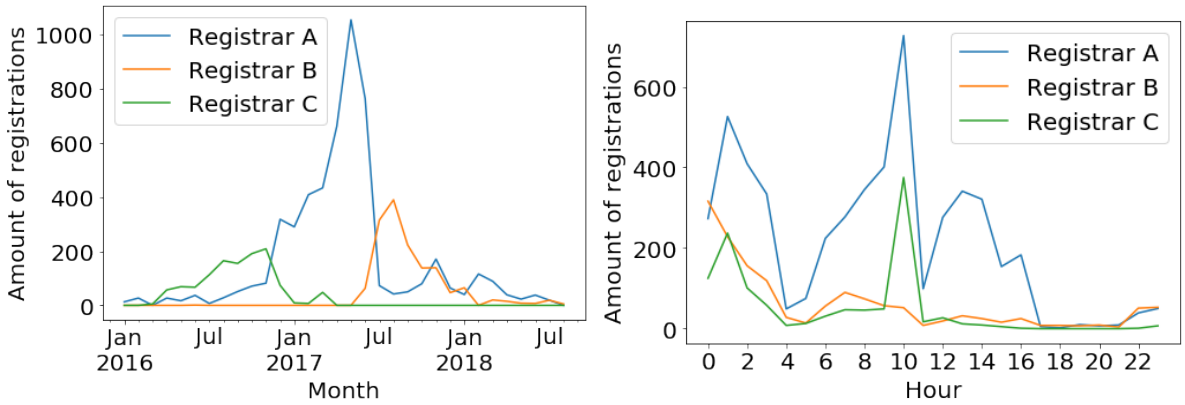
c_1 exhibits a curious peak in their hourly registrations. More specifically, there seems to be a massive increase in activity between 10 and 10:30 AM, as can be seen in figure 5.7(a). From figure 5.7(b) it is clear that this registration behavior takes place during the larger part of this campaign and is not the result of a single ‘burst’ of registrations. This persistent registration pattern could indicate that this campaign is partly automated. Because there are no notable differences in registration strategy within and outside of the automated time frame, it is likely that all registrations are part of the same campaign.

One of the most active campaigns, c_2, makes use of a single email provider, 163.com. This campaign is responsible for roughly 25% of all fake webshop registrations. As can be seen from figure 5.4, this campaign has been based on quite broad criteria. Registrations in this campaign all use randomly generated email addresses of varying structure and length. Because of this, it was not possible to create a more narrow criteria. Despite these broad criteria, there is evidence suggesting that malicious registrations using 163.com as email provider are all part of the same operations. For example, the campaign has been observed to switch their primary registrar, as can be seen in figure 5.8(a). Such a change indicates either a conscious decision to switch to a new registrar, or that all domains within the campaigns use a shared domain name reseller that changed to a new affiliate registrar. From figure 5.8(b) it shows that the first two registrars used show a spike during the 11th hour of the day, similar to c_1. This could indicate that c_1 and c_2 are in fact part of the same campaign, or, at least both campaigns use similar methods to register domains.



(a) (b)

Figure 5.7: (a) Average daily distribution of registrations per mail provider for campaign c_1. (b) Amount of c_1 registrations per month that took place in the 11th hour of the day.



(a) (b)

Figure 5.8: Total amount of registrations for campaign c_2 per registrar: (a) per month and (b) aggregated by hour.

Most campaigns used vast amounts of generated email addresses that were only used for one to three registrations. In contrast to this, some campaigns used a single email address for larger amounts of registrations, seemingly generating 'fresh' email addresses at a much slower pace. Campaigns c_6, c_8 and c_10 are such campaigns, using the same email addresses and contact information for an extended period of time. Campaign c_3 is also observed to reuse an email address, but seems to make an effort to pass the reused email address as a new one. This campaign exclusively uses Gmail as email provider, which purposely ignores dots in email addresses[2]. This means that sending an email sent to exa.mple@gmail.com will be received on example@gmail.com. This feature is used throughout campaign c_3, inserting dots on random positions of the same email address. It is likely a tactic used to outsmart systems that keep track of previously abused email addresses, possibly to avoid the registration being rejected due to prior abuse. Finally, c_7 consists solely of registrations made through a service that obfuscates WHOIS information, causing all registrations to come from a single registrant (i.e. a standard obfuscated registrant). While this makes it considerably harder to attribute to a single source, it can be seen from figure 5.5 that almost all fake webshops registrations through this service occurred in roughly the same period.

[2] https://support.google.com/mail/answer/7436150?hl=en

anon	Mail providers	Registrations	Email addresses	Registrars	Weekly	Daily
c_1	7	1833	1297	2		
c_2	1	7760	6926	3		
c_3	2	564	448	4		
c_4	2	1607	1358	2		
c_5	1	1454	1255	3		
c_6	2	610	8	5		
c_7	1	1317	932	6		
c_8	2	1312	75	1		
c_9	1	1414	1	1		
c_10	1	321	6	1		
c_11	1	815	37	1		
c_12	16	1225	1224	2		

Table 5.13: Statistics and activity patterns for identified campaigns.

5.5.3. Connecting campaigns

Many campaigns share properties that suggest they might be related. Multiple campaigns were seen to partake in the likely automated registration burst during the 11th hour of the day. Moreover, from table 5.13 it becomes clear that many campaigns are active during the same periods of the day, this observation will further be explored in section 5.6. While these observations are not enough to say that these campaigns are related, there is one observation that does build a strong case for this theory. Many registrants across campaigns share recurring first and last names, suggesting that these names are being generated by picking a random first and last name from predetermined lists. This theory is strengthened by a group of 1700 domains, for which it seems the same list has been used for both first and last names. The registrations in this group can all be connected by a list of 59 names, from which every observed combination of first and last name is created. The group spans multiple campaigns and uses a variety of different registrars, mail providers, and email generation methods. The existence of this group strongly suggests that at least some of the campaigns presented here are connected to each other and might be part of coordinated operation.

5.6. Attribution

While the campaigns identified in the previous section have their distinctive properties, there also is some evidence suggesting that many of these campaigns are related to each other. Of course, the peak at the 11th hour shared by some of these campaigns suggests that they might be part of a single, all-encompassing campaign. However, the hourly registration activity for these campaigns show more similarities, as can be seen from table 5.13. Curiously, many campaigns seem to be most active from midnight till noon, assuming European time zones. While not impossible, it is unlikely that malicious actors would purposefully awake in the middle of the night to start registering domain names. Thus, this activity pattern might indicate that these campaigns are executed from another geographical area. When looking for a geographical area with a timezone that might be a better fit with the overall campaign activity, China Standard Time (CST) seems to be a sensible choice. At UTC + 8, CST places the active periods of the campaign between 8 AM and 7 PM, that is, most campaigns seem active during Chinese business hours. From this context, the dip in registrations between the two active periods falls between 12 AM and 2PM and could possibly be ascribed to lunch breaks. These observations seem to hint that fraudulent webshops are predominantly registered by Chinese actors. Moreover, the fact that these actors operate during regular business hours suggests that the registration process is managed as a regular business.

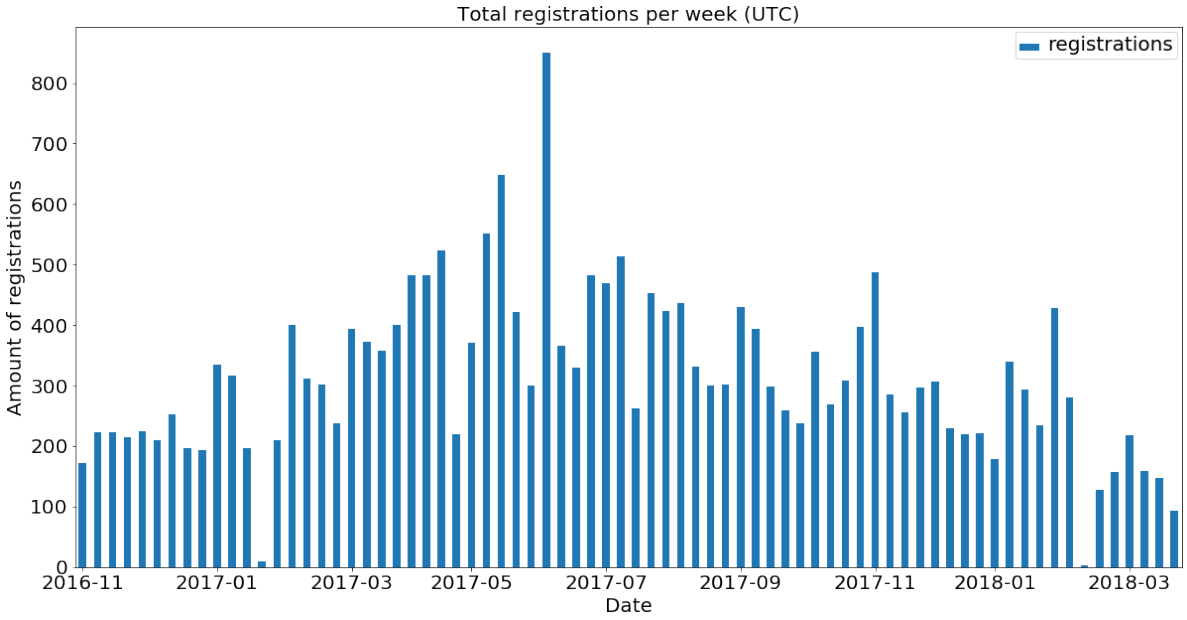


Figure 5.9: Campaign activity per week. The two weeks of inactivity coincide with Spring Festival.

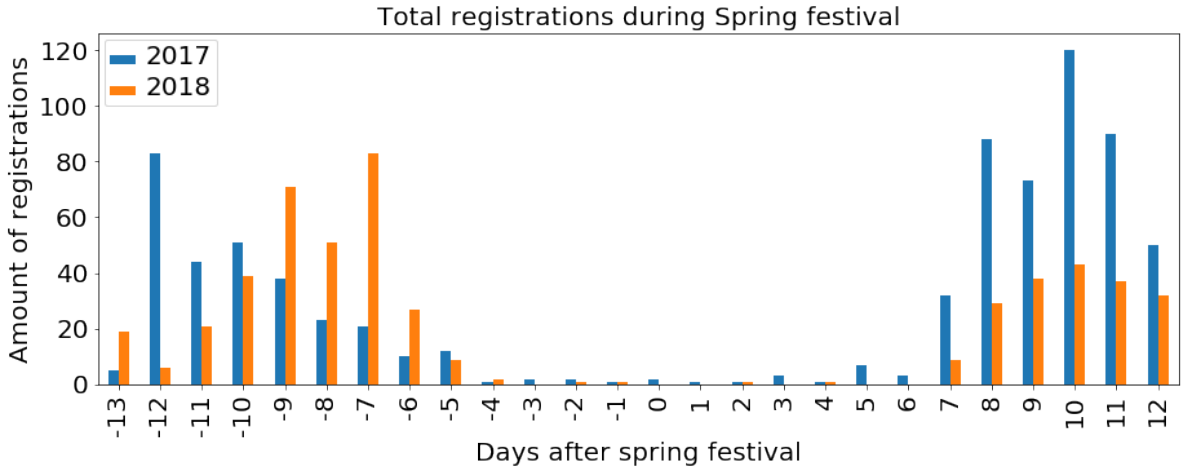


Figure 5.10: Daily registrations during spring festival.

These claims can be further substantiated by looking at weekly registration patterns, and comparing registration volumes between workdays and weekend-days. Especially Sundays show significantly less activity for many campaigns, indicating that these campaigns work along 5.5-6 day work weeks. Further evidence for the Chinese origin of the registrations comes in the form of two peculiar periods with a drastically reduced amount of malicious registrations, as can be seen in figure 5.9. While these two periods do not take place at the same moment of the year on the western calendar, they do share an annual event: Chinese new year. Chinese new year, also known as Lunar new year and Spring festival, is a public holiday in China. The date of the holiday is determined through the traditional Chinese calendar, which is a lunisolar calendar. During the festival friends and families get together to celebrate the start of the new year. In mainland China, the festival is a 7-day long public holiday. It is common for people to take time off from their job to travel to family in the days leading up to the holiday. This is visible in figure 5.10 as well, showing a decrease in activity 4 days before the start of Spring festival that lasts until 7 days after the event. Further inspection revealed similar drops in activity during other Chinese holidays such as the Ching Ming Festival, Mid Autumn Festival and the Dragon Boat Festival.

These observations strengthen both the theory of the malicious registrants' Chinese background and the claim that they operate as a normal business. More evidence of a Chinese background was found in open directories encountered on fraudulent domain names. These directories contained custom themes for content management systems used to host fraudulent webshops and image packs of the merchandise. The custom themes, represented by a handful of PHP files, contained comments in Chinese, detailing the working of the code. Furthermore, as can be seen from table 5.12, over 25% of all fraudulent domain registrations used 163.com as email provider, a Chinese company. In fact, roughly 40% of malicious registrations are using Chinese email providers, further strengthening the suspicions of Chinese actors being behind the majority of these fraudulent webshops.

6

Conclusion

Fraudulent webshops are a popular form of domain name abuse which plagues many Top Level Domains. Consumers, brand owners and registries all suffer damages due to the illegal practices that take place on these websites. Current mitigation strategies focus primarily on damage control and are of a reactive nature. This thesis explores the possibility for a preventive mitigation strategy, aiming to stop malicious actors before they have their window of opportunity.

A Registry for a large Top Level Domain, such as SIDN, has a large dataset available detailing information on their registrants. Personal information such as name, address phone number and email address provide rich data that can be used profile registrants. Unfortunately, because of the lack of authentication of these details, abusive registrants often use stolen or fictitious information for their registrations. Still, even though most data is not related to real persons, the data can be leveraged for detection purposes. Preferences for certain mail providers and countries of residence seem to make good indicators of pending abuse.

Other points of data, such as the choice of infrastructure, can not be fabricated by a registrant. Fraudulent webshop operators clearly favor certain registrars and hosting providers. It is likely that these parties are either financially attractive, or very hesitant to act when receiving abuse reports. However, this predilection can be leveraged to detect domain names that make use of this infrastructure. Similarly, fake webshop operators have the tendency to reregister domain names that have only recently been abandoned to exploit their residual trust. This tactic can also be used to detect these actors, as this behavior can be captured and used as a predictive value for abuse.

Using these available sources of data, a detection system has been designed to spot abusive registrations as close after the moment of registration as possible. The detection system has been designed to be run on a daily basis, retraining each day on the latest abusive registrations in order to keep track of the latest strategies deployed by abusive registrants. Experimental results show that the detection system is capable of detecting domain abuse using registration and infrastructure data. Because the detection tool does not require data regarding what is actually hosted on the domain, it is able to predict fraudulent registrations before the abuse has occurred. Experimental results show that the tool is capable to detect fake webshop registrations with a precision of at least 80%. Further improvements can likely be achieved by designing a method to obtain benignly labeled training data that more closely resembles a random sample of registrations than the current approach. It is likely that this will have a positive effect on the performance of the classifier.

However, detecting a fraudulent registration does not mean that the domain can be immediately taken down. In reality, registrars used for abusive registrations are hesitant to act on abuse reports. It is possible for SIDN to start a notice and take-down procedure, but this is a long process and is considered to be a last resort. In practice this causes criminals to still have a window of abuse, even though the abused domain was already suspected to be malicious. As long as a registry has no legal method to quickly force a domain to be taken down, actual prevention of abuse remains a hypothetical scenario.

Still, the detection system can be used to flag suspicious registrations. These registrations can then be monitored for signs of abuse, allowing SIDN to start sending abuse notifications the moment such a sign is observed.

Using the detection system, a body of over 30.000 confirmed fake webshop registrations was created. These registrations were investigated to understand the tactics used by the malicious actors making them. Interestingly, the data indicates that a subset of the registrations are part of an automated process, while at the same time obvious data-entry mistakes (e.g. typo's) are observed. Even more interesting were points of data that seem to disclose information concerning the identity of the actors behind the large-scale webshop fraud. Observing the pattern of activity exhibited by the malicious registrants reveals a large amount of evidence suggesting that the offending registrants operate from China or a neighboring country. This idea is reinforced by the obvious preference for Chinese mail providers, as well as Chinese comments observed in multiple source files encountered on fraudulent webshops.

Analysis of a large body of malicious registrations further suggest that registrant details are created through the usage of random name generators, some of which have a rather limited corpus. This leads to repeated usage of addresses and first and last names. This recycling of personal data can be used to engineer better features. For example, exploring word or string distances between registrant names might lead to better performance than the categorical approach used in this thesis. Similarly, other strategies observed during campaign analysis can likely be investigated to engineer more sophisticated features.

Bibliography

[1] Alexa Web Information Service. URL http://aws.amazon.com/awis/.

[2] *The Economic Impact of Counterfeiting and Piracy*. 2008. ISBN 9789264045514. doi: 10.1787/9789264045521-en. URL http://www.oecd-ilibrary.org/trade/the-economic-impact-of-counterfeiting-and-piracy{_}9789264045521-en.

[3] Large-margin Convex Polytope Machine. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*, pages 3248–3256, 2014. URL http://papers.nips.cc/paper/5511-large-margin-convex-polytope-machine.pdfhttp://dl.acm.org/citation.cfm?id=2969033.2969189.

[4] Report on EU customs enforcement of intellectual property rights: Results at the EU border 2015. Technical report, European Commission, Directorate-General for Taxation and Customs Union, Luxembourg, 2015.

[5] Cybersecuritymonitor 2017. Technical report, Statistics Netherlands (CBS), 2017. URL https://www.cbs.nl/nl-nl/publicatie/2017/06/cybersecuritymonitor-2017.

[6] Pieter Agten, Wouter Joosen, Frank Piessens, and Nick Nikiforakis. Seven Months ' Worth of Mistakes : A Longitudinal Study of Typosquatting Abuse. *Ndss*, (February):8–11, 2015. doi: 10.14722/ndss.2015.23058. URL https://www.securitee.org/files/typosquatting{_}ndss2015.pdf.

[7] Manos Antonakakis, Roberto Perdisci, David Dagon, Wenke Lee, and Nick Feamster. Building a Dynamic Reputation System for DNS. *USENIX Security'10: Proceedings of the 19th USENIX conference on Security*, pages 1–17, 2010. ISSN 10949224. doi: 10.1145/2584679. URL http://www.usenix.org/events/sec10/tech/full{_}papers/Antonakakis.pdfhttps://www.usenix.org/legacy/events/sec10/tech/full{_}papers/Antonakakis.pdf.

[8] Manos Antonakakis, Roberto Perdisci, Wenke Lee, Nikolaos Vasiloglou Ii, David Dagon, Nikolaos Vasiloglou Ii, and David Dagon. Detecting Malware Domains at the Upper DNS Hierarchy. *USENIX Security Symposium.*, 11:1–16, 2011. doi: 2028067.2028094. URL https://www.usenix.org/legacy/events/sec11/tech/full{_}papers/Antonakakis.pdfhttps://dl.acm.org/citation.cfm?id=2028094.

[9] Xuemei Bian, Kai Yu Wang, Andrew Smith, and Natalia Yannopoulou. New insights into unethical counterfeit consumption. *Journal of Business Research*, 69(10):4249–4258, 2016. ISSN 01482963. doi: 10.1016/j.jbusres.2016.02.038. URL http://dx.doi.org/10.1016/j.jbusres.2016.02.038.

[10] David Bianco. The Pyramid of Pain, 2014. URL http://detect-respond.blogspot.nl/2013/03/the-pyramid-of-pain.html.

[11] Leyla Bilge, Sevil Sen, Davide Balzarotti, Engin Kirda, and Christopher Kruegel. Exposure. *ACM Transactions on Information and System Security*, 16(4):1–28, apr 2014. ISSN 10949224. doi: 10.1145/2584679. URL http://dl.acm.org/citation.cfm?doid=2617317.2584679.

[12] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[13] Marcel Buskermolen and Carolien Govers. Altijd en overal online. Onderzoek Trends in internetgebruik 2016. Technical report, GfK, commissioned by SIDN, 2016.

[14] Juan Caballero, Chris Grier, Christian Kreibich, and Vern Paxson. Measuring Pay-per-Install: The Commoditization of Malware Distribution. In *Proceedings of the 20th USENIX conference on Security*, pages 13–13. USENIX Association, 2011. URL https://www.usenix.org/legacy/events/sec11/tech/full{_}papers/Caballero.pdf.

[15] Claudio Carpineto and Giovanni Romano. Learning to detect and measure fake ecommerce websites in search-engine results. *Proceedings of the International Conference on Web Intelligence - WI '17*, pages 403–410, 2017. doi: 10.1145/3106426.3106441. URL http://dl.acm.org/citation.cfm?doid=3106426.3106441.

[16] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM transactions on intelligent systems and technology (TIST)*, 2(3):27, 2011.

[17] Ming Cheung, James She, and Lufi Liu. Deep learning-based online counterfeit-seller detection. *IEEE INFOCOM 2018 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, pages 51–56, 2018.

[18] Daiki Chiba, Kazuhiro Tobe, Tatsuya Mori, and Shigeki Goto. Detecting malicious websites by learning IP address features. In *Proceedings - 2012 IEEE/IPSJ 12th International Symposium on Applications and the Internet, SAINT 2012*, pages 29–39. IEEE, jul 2012. ISBN 9780769547374. doi: 10.1109/SAINT.2012.14. URL http://ieeexplore.ieee.org/document/6305258/.

[19] Daiki Chiba, Takeshi Yagi, Mitsuaki Akiyama, Toshiki Shibahara, Takeshi Yada, Tatsuya Mori, and Shigeki Goto. Domain profiler: Discovering domain names abused in future. *Proceedings - 46th Annual IEEE/IFIP International Conference on Dependable Systems and Networks, DSN 2016*, pages 491–502, sep 2016. doi: 10.1109/DSN.2016.51. URL http://ieeexplore.ieee.org/document/7579766/.

[20] Mick Cox and Sjors Haanen. Content-based Classification of Fraudulent Webshops. pages 1–14.

[21] L Daigle. Ietf rfc 3912. *Whois protocol specification*, 2004.

[22] Erwin Boogert. 'Tien procent Nederlandse webshops frauduleus' - Emerce, 2017. URL https://www.emerce.nl/nieuws/tien-procent-nederlandse-webshops-frauduleus.

[23] Europol. Biggest hit against online piracy: Over 20 520 internet domain names seized for selling counterfeits, 2017. URL https://www.europol.europa.eu/newsroom/news/biggest-hit-against-online-piracy-over-20-520-internet-domain-names-seized-for-selling-counterfeits.

[24] Mark Felegyhazi, Christian Kreibich, and Vern Paxson. On the Potential of Proactive Domain Blacklisting. *Leet*, 42(8):6, 2010. ISSN 00380717. doi: 10.1016/j.soilbio.2010.03.017. URL https://www.usenix.org/legacy/event/leet10/tech/full{_}papers/Felegyhazi.pdfhttp://scholar.google.com/scholar?hl=en{&}btnG=Search{&}q=intitle:On+the+Potential+of+Proactive+Domain+Blacklisting{#}0.

[25] Fraudehelpdesk.nl. Checklist do's & don'ts bij online winkelen - Fraudehelpdesk Fraudehelpdesk. URL https://www.fraudehelpdesk.nl/nieuws/checklist-dos-donts-online-shoppen/.

[26] Hongyu Gao, Vinod Yegneswaran, Yan Chen, Phillip Porras, Shalini Ghosh, Jian Jiang, and Haixin Duan. An Empirical Reexamination of Global DNS Behavior. *SIGCOMM 2013 - Proceedings of the ACM SIGCOMM 2013 Conference on Applications, Technologies, Architectures, and Protocols for Computer Communication*, pages 267–278, 2013. ISSN 01464833. doi: 10.1145/2486001.2486018. URL http://dl.acm.org/citation.cfm?doid=2486001.2486018{%}5Cnhttp://dl.acm.org/citation.cfm?doid=2486001.2486018{%}5Cnhttp://www.cs.northwestern.edu/{~}ychen/Papers/sigcomm13.pdf.

[27] Michael Hanke and Florian Hauser. On the Effects of Stock Spam E-mails *. 2006. URL http://citeseerx.ist.psu.edu/viewdoc/download;jsessionid=BEBEFEEA2EAFB875292AC9B779D9F83C?doi=10.1.1.505.9805{&}rep=rep1{&}type=pdf.

[28] Gard Hopsdal Hansen and Henrik Kloppenborg Møller. Louis Vuitton in the bazaar: negotiating the value of counterfeit goods in Shanghai's Xiangyang market. *International Journal of Entrepreneurship and Small Business*, 30(2):170–190, 2017. ISSN 17418054. doi: 10.1504/IJESB.2017.081437. URL https://www.researchgate.net/publication/290070420_Louis_Vuitton_in_the_bazaar_negotiating_the_value_of_counterfeit_goods_in_Shanghai%27s_Xiangyang_market.

[29] Shuang Hao, Nick Feamster, and Ramakant Pandrangi. Monitoring the initial DNS behavior of malicious domains. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference - IMC '11*, page 269, 2011. ISBN 9781450310130. doi: 10.1145/2068816.2068842. URL http://dl.acm.org/citation.cfm?doid=2068816.2068842.

[30] Shuang Hao, Alex Kantchelian, Brad Miller, Vern Paxson, and Nick Feamster. PREDATOR : Proactive Recognition and Elimination of Domain Abuse at Time-Of-Registration. *Proceedings of the ACM Conference on Computer and Communications Security (CCS)*, 24-28-Octo:1568–1579, oct 2016. ISSN 15437221. doi: 10.1145/2976749.2978317. URL http://dl.acm.org/citation.cfm?doid=2976749.2978317.

[31] Thorsten Holz, Christian Gorecki, Konrad Rieck, and Felix C Freiling. Measuring and Detecting Fast-Flux Service Networks. *Ndss*, (January):24 – 31, 2008. ISSN 13669516. doi: 10.1.1.140.188.

[32] Guang Bin Huang. What are Extreme Learning Machines? Filling the Gap Between Frank Rosenblatt's Dream and John von Neumann's Puzzle. *Cognitive Computation*, 7(3):263–278, 2015. ISSN 18669964. doi: 10.1007/s12559-015-9333-0. URL https://link.springer.com/content/pdf/10.1007{%}2Fs12559-015-9333-0.pdf.

[33] Guang-Bin Huang, Qin-Yu Zhu, and Chee-Kheong Siew. Extreme learning machine: Theory and applications. *Neurocomputing*, 70(1):489–501, 2006. ISSN 09252312. doi: 10.1016/j.neucom.2005.12.126. URL https://ac.els-cdn.com/S0925231206000385/1-s2.0-S0925231206000385-main.pdf?{_}tid=7d6575d6-b94c-11e7-ae1c-00000aab0f6c{&}acdnat=1508912563{_}0f392e46b1a2f3408340dd7b7669afc1.

[34] Alex Kantchelian, Michael Carl, Tschantz Ling, Peter L Bartlett, Anthony D Joseph, and U C Berkeley. Appendix for Large-Margin Convex Polytope Machine. (5):5–7. URL https://www1.icsi.berkeley.edu/{~}mct/pubs/nips14appendix.pdf.

[35] Maciej Korczynski, Maarten Wullink, Samaneh Tajalizadehkhoob, Giovane Moura, Arman Noroozian, Drew Bagley, and Cristian Hesselman. Cybercrime after the sunrise: A statistical analysis of dns abuse in new gtlds. In *Proceedings of the 2018 on Asia Conference on Computer and Communications Security*, pages 609–623. ACM, 2018.

[36] Lars Kröhnke, Jelte Jansen, and Harald Vranken. Resilience of the domain name system: A case study of the. nl-domain. *Computer Networks*, 139:136–150, 2018.

[37] Jehyun Lee and Heejo Lee. GMAD: Graph-based malware activity detection by DNS traffic analysis. *Computer Communications*, 49:33–47, 2014. ISSN 01403664. doi: 10.1016/j.comcom.2014.04.013.

[38] K. Levchenko, A. Pitsillidis, N. Chachra, B. Enright, M. Felegyhazi, C. Grier, T. Halvorson, C. Kanich, C. Kreibich, He Liu, D. McCoy, N. Weaver, V. Paxson, G. M. Voelker, and S. Savage. Click Trajectories: End-to-End Analysis of the Spam Value Chain. In *2011 IEEE Symposium on Security and Privacy*, pages 431–446. IEEE, may 2011. ISBN 978-0-7695-4402-1. doi: 10.1109/SP.2011.24. URL http://ieeexplore.ieee.org/document/5958044/.

[39] Justin Ma, Lawrence K Saul, Stefan Savage, and Geoffrey M Voelker. Beyond Blacklists : Learning to Detect Malicious Web Sites from Suspicious URLs. *World Wide Web Internet And Web Information Systems*, pages 1245–1253, 2009. doi: 10.1145/1557019.1557153. URL http://portal.acm.org/citation.cfm?id=1557153.

[40] Federica Mangiatordi, Andrea Bernardini, Emiliano Pallotti, Licia Capodiferro, and Fondazione Ugo Bordoni. Multimedia analytics platform for profiling keywords embedded in photo catalogues. (c):1–5, 2018.

[41] Giovane Moreira Moura. *Internet Bad Neighborhoods*. PhD thesis, University of Twente, Netherlands, 3 2013. CTIT Ph.D.-thesis series no. 12-237.

[42] Giovane Moura, John Heidemann, Moritz Müller, Ricardo de O Schmidt, and Marco Davids. When the dike breaks: Dissecting dns defenses during ddos. In *Proceedings of the Internet Measurement Conference 2018*, pages 8–21. ACM, 2018.

[43] Giovane C M Moura, Moritz Müller, Maarten Wullink, and Cristian Hesselman. Title: nDEWS: a New Domains Early Warning System for TLDs. 2016. URL https://www.sidnlabs.nl/downloads/presentations/sidn-annet2016.pdf.

[44] Giovane C. M. Moura, Moritz Müller, Marco Davids, Maarten Wullink, and Cristian Hesselman. Domain names abuse and TLDs: from monetization towards mitigation. *2017 IFIP/IEEE Symposium on Integrated Network and Service Management (IM)*, (Im):1077–1082, may 2017. doi: 10.23919/INM.2017.7987441. URL http://ieeexplore.ieee.org/document/7987441/.

[45] Moritz C Müller and Andreas Peter. SIDekICk SuspIcious DomaIn Classification in the .nl Zone. 2015. URL https://pdfs.semanticscholar.org/13aa/010b41704edbc49ed968c2d5658944afa542.pdf.

[46] OECD. The Economic Impact of Counterfeiting and Privacy. 2015. URL http://www.ey.com/Publication/vwLUAssets/EY-rugby-world-cup-final-report/{\protect\TU\textdollar}FILE/EY-rugby-world-cup-final-report.pdf.

[47] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, 2011.

[48] Roberto Perdisci, Igino Corona, David Dagon, and Wenke Lee. Detecting malicious flux service networks through passive analysis of recursive DNS traces. In *Proceedings - Annual Computer Security Applications Conference, ACSAC*, pages 311–320, 2009. ISBN 9780769539195. doi: 10.1109/ACSAC.2009.36.

[49] Roberto Perdisci, Igino Corona, and Giorgio Giacinto. Early detection of malicious Flux networks via large-scale passive DNS traffic analysis. *IEEE Transactions on Dependable and Secure Computing*, 9(5):714–726, 2012. ISSN 15455971. doi: 10.1109/TDSC.2012.35. URL http://ieeexplore.ieee.org/document/6175908/.

[50] Amit Poddar, Jeff Foreman, Syagnik Sy Banerjee, and Pam Scholder Ellen. Exploring the Robin Hood effect: Moral profiteering motives for purchasing counterfeit products. *Journal of Business Research*, 65(10):1500–1506, 2012. ISSN 01482963. doi: 10.1016/j.jbusres.2011.10.017. URL http://dx.doi.org/10.1016/j.jbusres.2011.10.017.

[51] Dutch police. Aangiften van internetoplichting gedaald | politie.nl, 2018. URL https://www.politie.nl/nieuws/2018/februari/6/aangiften-van-internetoplichting-gedaald.html.

[52] Sara Quach and Park Thaichon. Dark motives-counterfeit selling framework: An investigate on the supply side of the non-deceptive market. *Marketing Intelligence and Planning*, 36(2):245–259, 2018. ISSN 02634503. doi: 10.1108/MIP-04-2017-0069.

[53] Erhard Rahm. Discovering product counterfeits in online shops. *Journal of Data and Information Quality*, 5(1-2):1–3, 2014. ISSN 19361955. doi: 10.1145/2629605. URL http://dl.acm.org/citation.cfm?doid=2667565.2629605.

[54] Laura Elena Raileanu and Kilian Stoffel. Theoretical comparison between the Gini Index and Information Gain criteria. *Annals of Mathematics and Artificial Intelligence*, 2004. ISSN 10122443. doi: 10.1023/B:AMAI.0000018580.96245.c6.

[55] Andrew Rosenberg and Julia Hirschberg. V-Measure: A conditional entropy-based external cluster evaluation measure. pages 410–420, 2007. URL http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.77.7562{&}rep=rep1{&}type=pdf.

[56] Kazumichi Sato, Keisuke Ishibashi, Tsuyoshi Toyono, Haruhisa Hasegawa, Hideaki Yoshino, Senior Member, Tsuyoshi Toyono, Haruhisa Hasegawa, and Hideaki Yoshino. Extending black domain name list by using Co-occurrence relation between DNS queries. *IEICE Transactions on Communications*, E95-B(3):794–802, 2012. ISSN 09168516. doi: 10.1587/transcom.E95.B.794. URL https://www.jstage.jst.go.jp/article/transcom/E95.B/3/E95.B{_}3{_}794/{_}pdf.

[57] Yong Shi, Gong Chen, and Juntao Li. Malicious Domain Name Detection Based on Extreme Machine Learning. *Neural Processing Letters*, 2017. ISSN 1370-4621. doi: 10.1007/s11063-017-9666-7. URL http://link.springer.com/10.1007/s11063-017-9666-7.

[58] Kyle Soska and Nicolas Christin. Automatically Detecting Vulnerable Websites Before They Turn Malicious. *23rd USENIX Security Symposium (USENIX Security 14)*, pages 625–640, 2014. URL https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/soska.

[59] Chiel Van Spaandonk, Elmer Lastdrager, and Erwin Lansing. Fake webshops on .nl and .dk. *CENTR Jamboree*, 2018.

[60] Andrea Stroppa, Politecnico Milano, Bernardo Perrella, Alessandra Spada, and Carlo Turri. Online Advertising Techniques for Counterfeit Goods and Illicit Sales. 2013.

[61] Brandon A. Sullivan, Steven M. Chermak, Jeremy M. Wilson, and Joshua D. Freilich. The nexus between terrorism and product counterfeiting in the United States. *Global Crime*, 15(3-4):357–378, 2014. ISSN 17440580. doi: 10.1080/17440572.2014.919227. URL http://dx.doi.org/10.1080/17440572.2014.919227.

[62] Janos Szurdi, Balazs Kocso, Gabor Cseh, Mark Felegyhazi, Chris Kanich, Balazs Kocso, Gabor Cseh, Jonathan Spring, and Mark Felegyhazi. The Long Taile of Typosquatting Domain Names. *USENIX Security Symposium*, 2014. URL https://www.usenix.org/conference/usenixsecurity14/technical-sessions/presentation/szurdihttps://www.usenix.org/system/files/conference/usenixsecurity14/sec14-paper-szurdi.pdf.

[63] Felix Tang, Vane Ing Tian, and Judy Zaichkowsky. Understanding counterfeit consumption. *Asia Pacific Journal of Marketing and Logistics*, 26(1):4–20, 2014. ISSN 17584248. doi: 10.1108/APJML-11-2012-0121.

[64] Hongwei Tian, Stephen M. Gaffigan, D. Sean West, and Damon McCoy. Bullet-proof payment processors. *eCrime Researchers Summit, eCrime*, 2018-May:1–11, 2018. ISSN 21591245. doi: 10.1109/ECRIME.2018.8376208.

[65] UNODC. The Illicit Trafficking of Counterfeit Goods and Transnational Organized Crime. 2014. URL https://www.unodc.org/documents/counterfeit/FocusSheet/Counterfeit{_}focussheet{_}EN{_}HIRES.pdf.

[66] Roland Van Rijswijk-Deij, Mattijs Jonker, Anna Sperotto, and Aiko Pras. A High-Performance, Scalable Infrastructure for Large-Scale Active DNS Measurements. *IEEE Journal on Selected Areas in Communications*, 34(6):1877–1888, jun 2016. ISSN 07338716. doi: 10.1109/JSAC.2016.2558918. URL http://ieeexplore.ieee.org/document/7460220/.

[67] Thomas Vissers, Wouter Joosen, and Nick Nikiforakis. Parking Sensors: Analyzing and Detecting Parked Domains. In *Proceedings 2015 Network and Distributed System Security Symposium*, 2015. ISBN 1-891562-38-X. doi: 10.14722/ndss.2015.23053. URL http://www.internetsociety.org/doc/parking-sensors-analyzing-and-detecting-parked-domains.

[68] Thomas Vissers, Jan Spooren, Pieter Agten, Dirk Jumpertz, Peter Janssen, Marc Van Wesemael, Frank Piessens, Wouter Joosen, and Lieven Desmet. Exploring the ecosystem of malicious domain registrations in the .eu TLD. 2017. doi: 10.1007/978-3-319-66332-6. URL https://lirias.kuleuven.be/bitstream/123456789/591728/3/Vissers-RAID2017.pdf.

[69] Wadleigh. Tracking How Cybercriminals Compromise Websites To Sell Counterfeit Goods. *Tylermoore.Ens.Utulsa.Edu*. URL https://tylermoore.ens.utulsa.edu/thesis/wadleigh.pdf.

[70] John Wadleigh, Jake Drew, and Tyler Moore. The E-Commerce Market for Lemons: Identification and Analysis of Websites Selling Counterfeit Goods. *Proceedings of the 24th International Conference on World Wide Web*, pages 1188–1197, 2015. doi: 10.1145/2736277.2741658.

[71] David Y Wang, Matthew Der Mohammad, Lawrence Saul, Damon Mccoy, Stefan Savage, and Geoffrey M Voelker. Search + Seizure : The Effectiveness of Interventions on SEO Campaigns. *Proceedings of the 2014 Conference on Internet Measurement Conference (IMC '14)*, pages 359–372, 2014. doi: 10.1145/2663716.2663738.

[72] Hayden Wimmer and Victoria Y. Yoon. Counterfeit product detection: Bridging the gap between design science and behavioral science in information systems research. *Decision Support Systems*, 104:1–12, 2017. ISSN 01679236. doi: 10.1016/j.dss.2017.09.005. URL https://doi.org/10.1016/j.dss.2017.09.005.

[73] Maarten Wullink, Giovane C. M. Moura, and Cristian Hesselman. Dmap: Automating Domain Name Ecosystem Measurements and Applications. In *IFIP/IEEE Network Traffic Measurement and Analysis Conference (TMA 2018)*, Vienna,Austria, June 2018.

[74] Sandeep Yadav, Ashwath Kumar Krishna Reddy, A. L. Narasimha Reddy, and Supranamaya Ranjan. Detecting algorithmically generated domain-flux attacks with DNS traffic analysis. *IEEE/ACM Transactions on Networking*, 20(5):1663–1677, 2012. ISSN 10636692. doi: 10.1109/TNET.2012.2184552.