Master Thesis
as part of the major in
Security & Privacy at the *EIT Digital Master School*

SIDekICk
SuspIcious DomaIn Classification in the .nl Zone

delivered by Moritz C. Müller
moritz.muller@utwente.nl

at the University of Twente

2015-07-31

Supervisors:
Andreas Peter (University of Twente)
Maarten Wullink (SIDN)

Abstract

The Domain Name System (DNS) plays a central role in the Internet. It allows the translation of human-readable domain names to (alpha-) numeric IP addresses in a fast and reliable manner. However, domain names not only allow Internet users to access benign services on the Internet but are used by hackers and other criminals as well, for example to host phishing campaigns, to distribute malware, and to coordinate botnets.

Registry operators, which are managing top-level domains (TLD) like *.com*, *.net* or *.nl*, disapprove theses kinds of usage of their domain names because they could harm the reputation of their zone and would consequentially lead to loss of income and an insecure Internet as a whole. Up to today, only little research has been conducted with the intention to fight malicious domains from the view of a TLD registry.

This master thesis focuses on the detection of malicious domain names for the *.nl* country code TLD. Therefore, we analyse the characteristics of known malicious *.nl* domains which have been used for phishing and by botnets. We confirm findings from previous research in *.com* and *.net* and evaluate novel characteristics including query patterns for domains in quarantine and recursive resolver relations. Based on this analysis, we have developed a prototype of a detection system called *SIDekICk*. It is able to detect newly registered phishing domains and other online scams as soon as they propagate through the Internet with a false positive rate of 0,3 percent. It relies solely on features that can be collected from the vantage point of any TLD registry like DNS query patterns, geographic features of querying resolvers, and domain registration information. A second component of *SIDekICk* reports suspicious domain names that were formerly used for benign purposes but might have been compromised to become part of a malware infection chain or a phishing campaign. This component demonstrates that DNS traffic analysis has the potential to detect compromised domains as well and in this thesis, we suggest additional features to improve the detection rate.

Acknowledgments

First of all, I want to thank my supervisors Maarten Wullink and Andreas Peter for the support during the six months of this graduation project. They always provided helpful and inspiring feedback and asked the right question to steer me towards my research goal. Furthermore, I would like to thank my fellow colleagues at SIDN Labs, who always had an open ear for technical questions and contributed to an inspiring and fun working environment.

Furthermore, I would like to thank for the support from my family and friends during my studies abroad. Especially, I would like to thank my father Herbert Müller for always supporting my decisions and for being there in times when advice and help was needed. I would like to thank my friends who visited me abroad and with whom I kept close contact and finally I thank all my fellow students and staff of the EIT Digital Master School program who made the last two years existing, inspiring and engaging.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Domain names provide a human readable representation of (alpha-) numeric Internet addresses. The Domain Name System (DNS) allows the translation of these domain names to corresponding IP-addresses and vice versa. It is hierarchically structured such that every domain name is part of a top level domain (TLD). These TLDs can be for generic purposes like *.com* and *.net* or can be associated with a country like *.uk*, *.de*, or *.nl*, which are referred to as country code TLDs (ccTLD). Each TLD is managed by a registry operator which is responsible for registration and delegation and guarantees its reachability. The registry operator for *.nl* is the *Stichting Internet Domainregistratie Nederland* (SIDN). SIDN manages the registration of domain names under the *.nl* ccTLD and provides the infrastructure that allows Internet users all over the world to translate these names to IP addresses in a fast, secure, and reliable manner. However, domain names not only allow Internet users to access benign services but are used by hackers and other criminals as well.

For example, websites, which are reachable under a domain, can host malicious code that infects the computers of its visitors and thereby personal information can be stolen or the infected machine can become part of a botnet. Botnets themselves can use domain names to enable infected clients to communicate with central command and control servers (C&C) in order to coordinate attacks or to receive updated malware. Last, domain names are part of phishing attacks were criminals impersonate legitimate services like banking websites or websites of social networks to trick users into entering their credentials. In the second quarter of 2014, Aaron and Rasmussen (2015) observed over 95.321 unique domain names involved in phishing campaigns and the website *malwaredomains.com* has listed 8.517 domains that were involved in the command and control of bots and in the distribution of malware in July 2015. According to the security company Kaspersky, distributed denial of service attacks executed by large botnets can cause companies a damage of over 400.000 EUR (Kaspersky Lab, 2015).

Identifying and taking down domain names that are involved in these attacks can reduce the damage.

SIDN and other registries rely on businesses, organisations and individuals that have an interest in hosting websites and therefore registering domain names in their zones. Every registered domain is a continuous revenue stream for the registry. A TLD that is mostly used for malicious purposes is not attractive for registrants with legitimate businesses. Thus, a secure, reliable, and trusted TLD attracts potentially more customers and increases revenues that can be used to maintain and strengthen DNS and the Internet as a whole. For this reason, SIDN runs several projects internally and in collaboration with other partners to actively fight misuse of domain names. This thesis is part of such an initiative.

A registry is in the unique position that it is able to observe DNS queries for every domain name in its zone from all over the world. Query patterns can indicate when a domain name is used for malicious activities and has the advantage that it does not rely on the analysis of the content of a website or the communication between a bot and a botnet server. Due to the large amount of domain names, it is cumbersome to detect these malicious domain names manually. Therefore, automatic methods have been proposed.

So far, only few attempts have been made to fight malicious domains on TLD level (Hao et al., 2010, 2011; Antonakakis et al., 2011). This thesis contributes to a more secure Internet by gaining insight into DNS activities of malicious domains from the vantage point of an ccTLD shortly after registration and after infection. It assesses whether previously described characteristics of malicious domains in other TLDs exist in *.nl* as well. Additionally, novel ways to detect malicious domain names are proposed and it is discussed if they are adequate for identifying previously unknown malicious domain names. Based on this analysis a prototype called *SIDekICk* (SuspIcious DomaIn Classification) is developed that can automatically detect malicious domain names based on DNS queries and registration data, collected at a TLD. We show that ccTLDs like SIDN are able to detect domain names used for malicious purpose few days after their registration with high precision. Also, we show how we can identify suspicious domain names that might be compromised and could be part of malicious activities. Thereby, domain names can be selected for further examination.

In the remaining part of this chapter, we explain the basic components and mechanisms of DNS and how a TLD is operated. Chapter 2 describes how DNS can be misused in order to steal data from infected machines, coordinate botnets and to host phishing campaigns to steal credentials or banking details. In Chapter 3, we present existing approaches to detect malicious domain names, describe the characteristics of domain names that make a detection possible, and list common techniques to classify domain names automatically briefly. Chapter 4 provides an analysis of benign and malign domain names in *.nl*. We describe the characteristics of known malicious

.nl domains and explain how they differ from benign domain names. Based on these observations we define the focus of *SIDekICk*. *SIDekICk* primarily has the goal to detect phishing domains. In Chapter 5 the architecture of *SIDekICk* is described, including the data collection, filtering, classification, and presentation components. The performance of *SIDekICk* is evaluated in Section 5.4. Finally, in the last chapter the results are summarised, an outlook is given how the performance of *SIDekICk* can be improved, and we discuss how *SIDekICk* can be part of a process to fight malicious activities on the Internet.

1.1 The Domain Name System (DNS)

In order to understand how hackers misuse DNS for their purposes and how these activities can be detected it is necessary to first understand the basic mechanisms of the system.

DNS helps to resolve human-readable domain names like *www.example .org.* into the actual IP-address of the server where the services are hosted - and vice versa. Each element of a domain name separated by a dot is called a *label*. Domain names are organised in a tree structure (see Figure 1.1) where the label on the most right is the root node (represented by the dot '.') (Mockapetris, 1987a). Below the root node is one of the publicly known top-level domains (TLD) which can be country specific as well. In case of the Netherlands it is the country-code top-level domain (ccTLD) *.nl.* which is maintained by SIDN. Each label on the left of the top-level domain label specifies a subdomain. Thus, *.example* is a subdomain of *.org.* and *www* is a subdomain of *.example.org.*

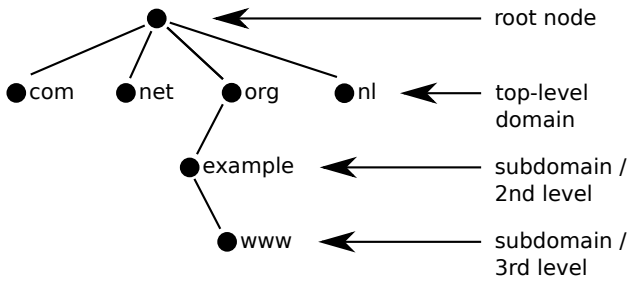

Figure 1.1: DNS tree

1.1.1 Address Resolution

In order to be able to resolve a domain name such as *www.example.org.*, a host must rely on services that provide the translation from a domain name to an IP-address. This includes five steps as depicted in Figure 1.2:

1. First, the host looks up locally if it already knows an IP-address that corresponds with the searched domain name (Step *1*). For this, the host usually has an own cache in which it stores domain names that have been resolved recently. If there is no entry for the requested domain in the cache, then the host has to contact another entity. The DNS functionality within the host are referred to as a stub-resolver.

2. If no corresponding entry in the cache of the stub-resolver has been found, it has to contact a DNS resolver. The address of the DNS resolver is often given by the Internet Service Provider (ISP) of the host. This resolver is usually a recursive DNS (RDNS) resolver that will take over the responsibility to resolve the domain name and will return the IP address or an error message if the domain does not exist. Similar to the stub resolver, the RDNS resolver has a cache as well. It first checks if the domain has already been resolved and if not, it continuous with Step 3. In case the domain is already in the cache, the RDNS resolver will respond to the host with a non-authoritative answer which includes the IP address.

3. The RDNS resolver will start by contacting one of the DNS root servers in order to find out who is responsible for the *.org.* domain. The root server will reply with the IP address of the TLD server which is responsible for *.org.* (Step *3a*). Top level domains are managed by name servers that do not reply with an IP-address for the whole domain but instead respond by sending the domain name of another DNS server that is responsible for the *example.org* domain (Step *3b*). In order to avoid that every RDNS first queries a root server, caches are here in place as well.

4. As soon as the recursive resolver has received the IP-address of the next DNS server it sends out another query. If this DNS server is responsible for the domain *www.example.org.*, it will respond with the IP address (Step *4*). Because this DNS server is responsible for the domain it is called an Authoritative Name Server (AuthNS) and it replies with an authoritative answer. The RDNS then will know that the answer was not cached but comes directly from the responsible name server.

5. Last, the RDNS server sends the IP address of the requested domain to the host and the host can connect to the Internet service (Step *5-6*).

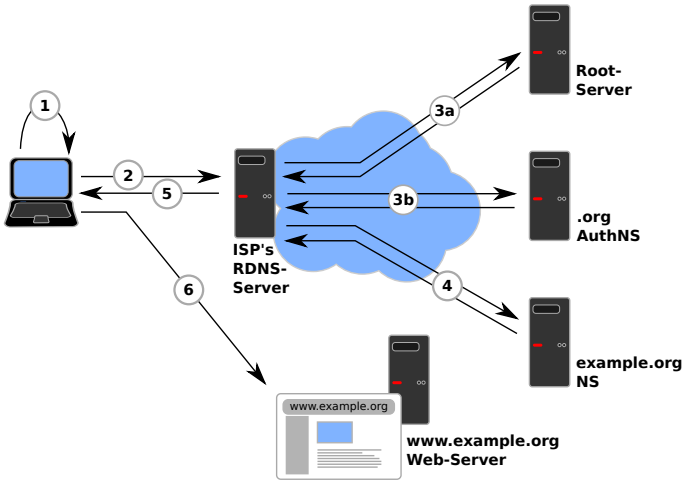

Figure 1.2: DNS flow

Each DNS entry in a cache has a limited life-time called time-to-live (TTL). The entry is deleted from the cache after the TTL has been expired and after a new request the IP-address has to be resolved again. The TTL is defined by the authoritative name server.

1.1.2 DNS Message Format

All DNS messages have the same format. It includes a *header*, a *question*, an *answer*, an *authoritative* section, and a section for *additional information* (Mockapetris, 1987b).

- *Header*: The header defines, which sections are present and whether it is a DNS query or a response. An Authoritative Answer (AA) flag defines if the answer comes from an AuthNS. A Response Code (RCODE) *3* can inform the requesting client that the requested domain does not exist. This response is called *NXDOMAIN*.

- *Question*: The question section defines for which domain an answer is requested.

- *Answer, Authoritative, Additional Information*: The answer, authoritative, and additional information section all have the same format called Resource Record Format (RR). The name field specifies which section follows. A type field defines the resource record and a TTL field specifies how long a record should be cached. Different types of RRs are described below.

Many different RR types have been defined, however only a small number are used in the wild. The following are the most relevant types for this paper:

- *A and AAAA*: The IPv4 or IPv6 address of a host.
- *NS*: Defines the address of a name server that might know the answer to the requested domain name.
- *CNAME*: The canonical name record is used as an alias for existing domain names.
- *MX*: Host information for a mail exchange.
- *PTR*: Used for recursive DNS lookups.
- *TXT*: Can hold descriptive text. No semantics are defined.

If for example a RDNS server queries a ccTLD server for the domain *example.org*, it will receive the address of the name server which is responsible for this domain (the AuthNS). Additionally, a section is attached where already the IP address of the authoritative name server is added. This additional section is called a "glue record".

1.1.3 Domain Registration

The Internet Corporation for Assigned Names and Numbers (ICANN) has the oversight over domain names. ICANN delegates the right to use TLDs and hands out domains to registry operators. For example, the top level domain *.com* is managed by VeriSign and the ccTLD *.nl* is managed by SIDN. SIDN does not sell domain names directly to the end users but uses intermediaries and provides a central register of the registered domain names. It contains for each registered *.nl* domain the name of the owner of the domain (the registrant), contact information, name-servers, a creation date, and information about the company that sold the domain name. This company is called a registrar.

After a claim for a *.nl* domain name has expired, it is put into quarantine for 40 days. During this time, only the former owner can reclaim the domain name. After these 40 days, the domain name is again available for registration for every interested buyer. Other registries have similar mechanisms in place. This prevents for example that registrants accidentally lose their domain names, when they forget to pay their bill at the registrar.

There are companies in the domain name eco-system that buy popular domain names from registrars and resell them for a higher price. They are called *domainers* and are specialised in registering domain names that leave quarantine within seconds. Domainers play a role in the detection of malicious domain names because they are responsible for certain query patterns.

1.2 The Situation at SIDN

The SIDN is responsible for managing the ccTLD *.nl*. The ccTLD *.nl* is the 5th largest ccTLD and the 10th largest TLD in the world (Verisign, 2014). In December 2014, over 5,5 million domains were registered in total (SIDN, 2014b). Additionally, SIDN provides the registry service for the new gTLD *.amsterdam* and the ccTLD of Aruba. SIDN runs authoritative name servers to answer queries from all over the world, including four unicast and multiple anycast servers. Over 15.000 queries are answered every second (Hesselman et al., 2014). These queries are received from 3.211.225 DNS resolvers, where the majority sends less than hundred queries every day (Hesselman et al., 2014). In order to store and analyse the large amount of requests, SIDN has introduced the *ENhanced Top-level domain Resilience through Advanced Data Analysis* (ENTRADA) big data platform (Wullink, 2015). ENTRADA is embedded into a privacy framework that includes legal, organisational and technical aspects to protect personal information of the users behind the recursive resolvers. The framework specifies among others the type of personal data which is processed by each project that uses ENTRADA, the instances that have access to the data, and it allows to filter out sensitive information such as IP addresses of certain sources or to aggregate queries to remove personal details. So far, ENTRADA stores query data of one name server in a Hadoop framework which then can be accessed by researchers of SIDN. Further, SIDN manages a database where information about each domain is stored. This database can be queried with the WHOIS protocol.

SIDN does not sell *.nl* directly but provides intermediary registrars the possibility to enrol as a reseller or use resellers themselves. Each registrar pays fees for registry transactions and monthly fees to SIDN to be allowed to sell *.nl* domains. For the *.nl* domain over 1.500 registrars from 27 different countries are allowed to sell domain names. These include registrars from the Cayman Islands and Singapore[1].

The business model of SIDN relies on the high reputation of *.nl* domains and their trustworthiness. Thus, it is in the interest of SIDN that the domains are not misused by miscreants to run SPAM and phishing campaigns or to host their botnet infrastructure. Therefore, SIDN has the goal to identify these malicious domains as soon as possible in order to take them down. At the moment, there are no statistics about how many malicious domains are registered under the *.nl* domain. Previous researchers of botnets and malicious domains world wide have not listed *.nl* among the top 10 of infected domains. It is considered as one of the ccTLD's that have strong security-related policies, next to countries like Iceland, Sweden, Japan and Canada (Futai et al., 2013; Nazario and Holz, 2008). For example, SIDN

[1] www.sidn.nl/registrars

does not allow refunding of registration costs in case a claim for a domain is dropped few days after registration. Thereby, misuse by spammers and phishers can be reduced (Wisniewski, 2009). Therefore, it would be expected that these ccTLD make it harder for hackers to register malicious domains. However, Antonakakis et al. (2011) have developed a method to detect malicious domains in the Canadian *.ca* domain and still found several malicious domains, despite strong policies. Moreover, a quick look at Domain Black Lists like *www.malwaredomains.com* already shows, that malicious *.nl* domains exist as well. In order to get a clear understanding about the existence of malicious domains in *.nl*, it is necessary to analyse the situation in more detail. Additionally, this will allow SIDN to estimate if the the number of domain name abuse increases in the future and to implement proactive countermeasures to fight this behaviour.

Chapter 2

The Misuse of Domain Names

On one side, DNS makes it easier for a user to find resources on the Internet and can increases their availability. On the other side, DNS can help miscreants to increase the availability of their services as well and allows them to hide their activities. In this chapter, three typical malicious use-cases of domain names and their characteristics are described. It is shown, that domain names play a critical role in the life cycle of malicious activities on the Internet. This serves as a motivation for SIDN to develop a system that detects those domain names.

2.1 Exploit Kits

Exploit kits are toolkits that allow miscreants to infect a computer of a victim easily (Grier et al., 2012). They bundle different exploits that target for example vulnerabilities in web-browsers or browser plugins like Java and Adobe Flash. These exploits are then used to load malware on the targeted machine in order to steal passwords, deploy ransomware or to make the machine a member of a botnet. Multiple domains can be involved in the process from leading web-users to an attack page, infecting their machines, to transferring stolen data back to the attacker.

Although exploit kits are not a new phenomenon, they have gained popularity in the recent years and are now among the most popular web-based attacks (Chen and Li, 2015).

2.1.1 Exploit Kit Infection Chain

Usually, a machine is not infected directly but gets redirected through multiple websites and communicates with several servers (Grier et al., 2012). These steps are depicted in Figure 2.1. At first, the user visits a website

that triggers the infection chain. This can occur for example by clicking on a link in a SPAM mail or by visiting a benign website that hosts malicious code directly or within an advertisement banner from a third party (Step *1*). This code initialises a number of redirects through multiple websites with the intention to hide the location of the site that hosts the exploit kit (Step *2*). The first websites are often compromised, the latter website is often a dedicated website, registered and hosted by the attacker. The exploit kit tries to identify the software and plugins that run on the victim's computer and tries to gain access to the machine by using exploits that it has in its tool box. If the kit was able to gain access, malware can be loaded onto the machine. It can fulfil different purposes such as stealing credentials or making the computer part of a botnet (Step *3*). Malware can be hosted on the same server as the exploit kit or can be loaded from other machines in the Internet. Depending on the used malware, the infected machine starts communicating with other servers to transfer stolen data or to receive further command and control information.

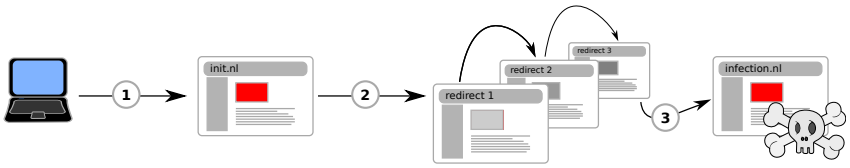

Figure 2.1: Exploit kit infection chain

Normally, there are at least three different domains involved in the infection chain. The initial domain can exist for several weeks, whereas the domain that hosts the actual botnet has a lifetime of only a few hours. During their research, Grier et al. (2012) have detected over 16.000 unique domains that were used as initial redirection website and over 6.000 unique domains that hosted the actual exploit kit. Furthermore, over 91.000 unique URLs were discovered. The median number of daily DNS queries for the initial compromised domains was 30, which leads to the assumption that especially websites were compromised that were used by private individuals or small companies and were not well maintained.

2.1.2 Obfuscation and Extensions

In order to make it harder to detect and take down websites that are involved in the delivery of exploit kits, developers of these kits implement self-defensive techniques (Eshete and Venkatakrishnan, 2014). These include modifying and obfuscating the malware code to avoid detection based on signatures and to make reverse engineering harder. Furthermore, they deny access to search-engines crawlers for example through IP blocking or by setting access permissions in the *robot.txt* file. Also, manual inspection

of suspicious URLs is made more difficult by displaying an empty page or by returning an 404 error after the infection was successful.

A technique that uses subdomains to obfuscate the infection chain becomes more popular among exploit kits developers. With domain shadowing, a miscreant uses hijacked registrant accounts to create a large number of subdomains under a benign domain name without the knowledge of the owner of the website (Biasini and Esler, 2015). These subdomains are then either used in the redirection chain or as a final exploitation site. Subdomains used for redirection can be a string of random looking characters like *mdfct6lfx8hccp56knyxlxj* or can contain English words like *says.imperialsocks .com*. Subdomains used for infection are mainly composed of random characters. Third and fourth level domains have been observed (Biasini, 2015). The lifetime of the infected domain can be only a few minutes, which makes blacklist attempts more difficult. Even in the case a domain is blacklisted, it is very likely that the benign site which is hosted on the 2nd level is blacklisted as well.

Malign subdomains of the same 2nd level domain share the same IP address and in some cases, subdomains of different 2nd level domains were directed to the same IP address.

2.2 Botnets

Botnets are a large group of infected computers, connected to the Internet. Botnet software can be delivered through an exploit kit or through a dedicated drive-by download, by tricking a user into installing a file, or through other vulnerabilities in the operating system or the browser. This allows the hacker to take over the control of the infected machine, which is called a bot from then on. Botnets can include up to several hundred thousand computers, distributed all over the world (Nazario, 2012).

These botnets are used for different malign purposes. From Distributed Denial-of-Service (DDoS) attacks, launching big SPAM campaigns and phishing to stealing data and carrying out click fraud - the possibilities for a owner of a botnet are wide and lucrative. DNS has a specific role in order to maintain these botnets, but before its role is described in more detail, the basic technical features of typical botnets are explained.

The hacker, who distributed the malware and has control over the bots is further referred to as *botmaster* (Rodríguez-Gómez et al., 2013).

In order to control the bots, the botmaster must be able to send commands to the infected machines. Botnets make use of different architectures like Peer to Peer (P2P), a centralised client-server communication or a combination of both (Rodríguez-Gómez et al., 2013). In this section, we only focus on client-server communication where DNS plays an especially important role.

In a centralised botnet that relies on HTTP the topology is similar to a client-server infrastructure where the infected bots are the clients and a central Command and Control (C&C) machine is the server. The C&C server is responsible for communicating with bots and is under the control of the botmaster. The bots use the Hypertext Transfer Protocol (HTTP) to receive commands from the server. Therefore it queries the server frequently for new commands (so called "pull"-style (Gu et al., 2008b)).

At first, botmasters only used a small number of C&C servers to control the bots, hardcoding the IP addresses into the malware. In order to increase the availability of their botnets, botmasters started to use domain names to address the servers and hiding the servers behind a second layer of proxies.

2.2.1 A Botnet Lifecycle

A bot goes through different stages during its lifetime: *initial infection, secondary injection, connection, malicious command and control, update and maintenance* (Feily et al., 2009).

Initial Infection First, the initial malware is downloaded to the computer that should be infected. This can happen, for example by tricking the user into downloading a malicious file or by using a vulnerability in the browser or operating system.

Secondary Injection Second, the initial malware allows the attacker to download the actual bot binary from a central FTP or HTTP server. After the binary is downloaded it is installed on the computer.

Connection Third, the bot tries to establish a connection with the C&C server to join the botnet. For example, bots of the ZeuS network connect to a C&C server right after the host has been infected (Mahjoub et al., 2014).

Malicious Command and Control Depending on the malicious action that should be carried out, the bot receives different commands. Popular attacks include distributed denial of service (DDoS) attacks, SPAM, phishing and click-fraud (Rodríguez-Gómez et al., 2013).

Update and Maintenance In order to prepare a bot for a new malicious activity or to equip it with the latest evasion techniques, the bot has to download and install the newest version of a bot-binary. Also, in some cases it might be necessary to delete the bot from the infected machine remotely.

2.2.2 Botnet Detection Evasion

It was a long way from the early days of simple, relatively small botnets to the sophisticated and large botnets we find today. A "*cat and mouse game*" started where security researchers on one side try to find ways to identify and take down botnets and hackers on the other side continuously develop new ways to evade detection and increase availability and resilience of their botnets.

In the beginning of the raise of centralised botnets, bots usually had the IP address of the C&C server hard-coded into the binary of the malware (Morales et al., 2009). This allowed researchers to identify these servers easily, for example by re-engineering the malware or by observing its communication behaviour. As soon as the central server is identified, communication of the bots with the server can be blocked by a firewall, the traffic to this address can be rerouted, or the server itself can be taken down from the network.

As a reaction, botmasters searched for ways to increase the resilience of their botnets. The basic approach is to provide multiple C&C servers. However, if again every address of those servers are hard-coded into the malware then they are still easy to take down and managing the addresses is cumbersome. Thus, hackers made use of DNS in order to increase redundancy of their C&C servers, to hide the infrastructure of their botnet and to simplify the management of their servers. Different approaches are explained below.

Domain Flux

Domain flux is a technique which makes use of a *Domain Generation Algorithm (DGA)*. This algorithm generates pseudo-random domain names, depending on a given random seed (e.g. the current date) (Antonakakis et al., 2012). Each bot is rolled out with a DGA. In order to contact a C&C server, it generates a number of domains and tries to resolve each them with until it receives a valid IP address from the DNS server. Before, hackers already have registered some of these domains and assigned their C&C servers to them. For every attempt to resolve a domain name which is not registered, the queried DNS server replies with an NXDOMAIN response. Thus, botnets which use a DGA usually generate many NXDOMAIN responses, which can be used to identify bots (see Chapter 3).

If an IP address of a server is identified and taken down, the botmaster just registers a new domain, generated by the DGA, and assigns a new C&C server to this domain. Thus, domains generated with a DGA usually have a rather short life-span (Bilge et al., 2011). Also, those domain names often have different lexical characteristics than benign domain namse (Bilge et al., 2011). Examples of botnets which use a DGA are Bobax, Torpig (Stone-Gross et al., 2009) and the Conficker bots (Porras et al., 2009a; Porras, 2009).

Variants of Conficker generate up to 50,000 domains every day (Porras et al., 2009b). A recent version of the bot Rovnix generates domains that look like benign domain names on first sight (Kruse, 2014). As shown in Table 2.1, Rovnix domains are more similar to legitimate domain than domains used by Conficker (Schiavoni et al., 2014). Also the Matsnu bot doesn't generate randomly-looking domains but relies on a dictionary of 1.300 words to create domain names which look legitimate in order to trick known DGA detection techniques (Mimoso, 2014).

Conficker Domain Name	*jbkxbxublgn.biz*
Rovnix Domain Name	*accordinglytathdivine.com*

Table 2.1: Examples of domain names generated by different DGAs

Fast Flux Service Networks (FFSN)

The basic idea of Fast Flux Service Networks is to increase the availability of botnets and to be thereby more resilient against take-downs of C&C servers. In FFSNs, multiple IP addresses of different C&C servers are assigned to one domain. If a bot queries a domain name, a DNS server replies with multiple IP addresses (A-records) at once. Then, the bot selects randomly or by a certain scheme (e.g. round robin) one of the IP addresses. If a bot queries the same domain name few minutes later, the DNS server might respond with a set of partially or completely different IP addresses. One domain can have thousands of different IP addresses assigned to it over time (Riden, 2008). Thus, the botnet stays operational even if multiple C&C servers are taken down. The returned IP addresses do not belong to the actual C&C servers but to an additional layer of bots - so called *flux-agents*. These flux-agents act as proxies for the actual C&C servers. This technique allows botmasters to hide their C&C servers and increase the resilience of their botnets which is not only useful for botnets but is used in phishing campaigns as well. Research by (Holz et al., 2008) has identified over 600 flux-agents in one botnet.

Today, two different implementations of FFSN can be found - *Single Flux* and *Double Flux*.

Single Flux In a Single Flux configuration, only the IP addresses of the flux-agents change over time. The communication of a bot with a C&C server is explained with the help of the fictive malicious domain *cnc.bad-domain.com* as depicted in Figure 2.2:

First, the bot sends a DNS request to one of the the DNS root servers. The root server will respond with an address of an NS which is responsible for

the *.com* top-level domain. The NS will respond with the IP address of the AuthNS which is responsible for the sub-domain *bad-domain.com*. Often, multiple name-server are responsible for one domain (first steps omitted in graphic). Then, the bot sends a DNS query to the name server and receives a number of A-records of flux-agents (Step *1*). The bot selects one of the IP addresses to which it sends its request (Steps *2* and *3*). From the view of the bot, it communicates directly with the C&C server, but in reality the flux-agent acts as a proxy and forwards the requests to the actual C&C server (Riden, 2008).

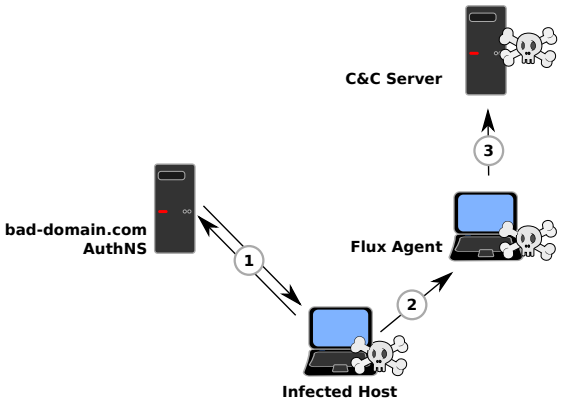

Figure 2.2: Single Flux flow

Double Flux Compared to a Single Flux configuration, not only the A-records, but also the NS-records change over time. Using the example above, the bot again queries a root server, followed by the NS of the *.com* domain. The NS replies with a number of AuthNS responsible for *bad-domain.com*. However, the returned names-servers are now already part of the FFSN. In the configuration of a Double Flux botnet, even the IP addresses of the name servers change frequently. The bot picks one of the name-servers and sends a DNS request for *cnc.bad-domain.com*, receives multiple IP addresses and picks one of them (step *1*). The subsequent flow is the same as in single Flux networks (Steps *2* and *3*). Usually, name-servers and the flux-agents run on the same computer (Riden, 2008).

This configuration give botmasters the advantage that flux-agents are disposable and hide the actual C&C servers. Flux-agents do not have to be as powerful machines as C&C servers and can therefore choose from a larger pool of infected computers. Furthermore, botnet researchers can often only observe the communication from the bot to the flux-agent which protects the C&C server from being identified and taken down.

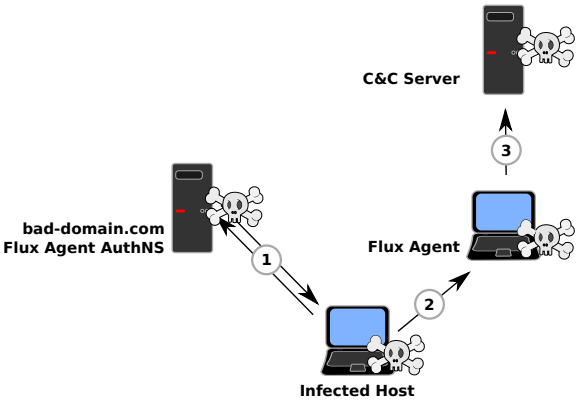

Figure 2.3: Double Flux flow

Similarities to Benign Load-Balancing Techniques FFSNs have similarities to legitimate techniques to improve reliability, availability and performance for web services which makes it harder for researchers to distinguish them. Round-robin DNS (RRDNS) is a method that returns not only one, but a list of A-records with every DNS response. The client then chooses one of the returned IP-addresses. Often, A-records have a TTL of less then 1.800 seconds (Holz et al., 2008).

Content Delivery Networks (CDN) are another possibility to perform load-balancing and to increase responsiveness for web-services. Again, several A-records are assigned for one domain name but in this case the DNS server only returns the IP-address of the server that fits best to the requested client. This is for example determined based on the location of the server and the client or based on the load on the link which connects the server with the client. The servers of CDNs are usually distributed all over the world but also closely clustered in central geographic locations (Stalmans et al., 2012). A-records of CDNs have a lower TTL than A-records of RRDNS services (Holz et al., 2008).

FFSN Characteristics FFSNs often have characteristics that differ from benign load-balancing techniques. First, botmaster usually cannot freely choose which host to infect. Therefore, bots and flux-agents vary in geographical location, variety of Autonomous Systems (AS), number of prefixes, IP diversity, and have variable and unpredictable up-times (Stalmans et al., 2012; Huang et al., 2010; Martinez-Bea et al., 2013). Also, the number of A-records for a fast-flux domain is often 5 or higher whereas legitimate services do not exceed three records. The same is true for the number of NS-records in case of a double flux infrastructure (Holz et al., 2008). Furthermore, the total number of returned IP addresses for one domain over time is very large and their TTL is comparably low. However this can be

true for CDNs as well (Perdisci et al., 2009). Additionally, the hosts which are part of a FFSN need to have a unique IP address that is globally accessible (Nazario and Holz, 2008). Passerini et al. (2008) further identified two typical characteristics of FFSN-domains. First, benign domains usually have a longer lifetime whereas the lifetime of FFSN-domains is on average only five weeks. Second, FFSN-domains are usually registered at registrars in countries with lax legislation against Internet crime.

DNS Blacklist Checking

Security researchers run several websites that list domain names which have been categorised as malicious. Two examples are the websites *www.malwaredomains.com* and *www.malwaredomainlist.com*. Both services run own detection mechanisms as described in Chapter 3. These websites should help for example administrators to filter out SPAM mails or to block malicious content from entering their local networks. However, also botmasters have discovered the usefulness of those services and have instructed their bots to query these services to check whether their phishing domains and bots are blacklisted (Lee and Lee, 2014).

DNS Record Hacking

One approach that helps botmasters to prevent their malicious domains from showing up on blacklists is to connect their domains to a legitimate domain. In 2012, hackers used compromised accounts of the private domain registrar *godaddy.com* to add additional subdomains to already registered, legitimate hostnames. The hostname still directed to the original website whereas the A-record of the subdomain pointed to a malicious website of the botmaster (Howard, 2012). Thereby, they not only can avoid blacklisting but also can imitate benign websites for SPAM and phishing. In case that malicious activity from the domain is detected, it is likely that the complete domain is blacklisted and thereby the benign domain is not reachable anymore as well.

Fake DNS Queries

Because some detection mechanisms, which are described in Chapter 3, identify botnets based on communication patterns, botmasters now modified their bots such that they send out legitimate DNS queries as well (Lee and Lee, 2014). Also, they are trying to avoid synchronous DNS and C&C traffic of their bots. Thereby, they can deceive detection mechanisms that rely on detecting group activities of botnets (Choi et al., 2009).

DNS Tunneling

As described by Dietrich et al. (2011), botmasters have misused the DNS protocol to hide C&C traffic. DNS has the advantage that it is one of the few protocols that is usually allowed to pass even very restrictive firewalls. Furthermore, it makes it possible to hide botnet activities from detection techniques that only focus on HTTP traffic. The *Feederbot*-network uses the TXT RR to send encrypted messages from the bot to the C&C servers.

Because the DNS messages cannot be resolved by normal DNS resolvers, *Feederbot* bypasses pre-configured DNS resolvers. However, in case this is not successful the pre-configured DNS will fail to resolve the domain name and will reply with a NXDOMAIN response.

Encryption

In order to bypass firewalls and to hide their activities, bots often encrypt the communication between their peers and C&C servers (Zhao et al., 2013; Rodríguez-Gómez et al., 2013). Encryption hinders botnet-researchers to identify bots based on the packet content and forces them to develop content agnostic detection techniques.

Polymorphism

Polymorphism describes the possibility of bots to change the source code of their malware, while the functionality stays the same. Thereby, signature based malware detection on a host becomes more difficult (Rodríguez-Gómez et al., 2013).

2.3 Phishing

Phishing attacks have the goal to steal valuable information from their victims. These include credit card information and user credentials. According to the recent report by the Anti-Phishing Working Group, the most targeted industry sector were eCommerce websites, followed by bank and money transfer organisations (Aaron and Rasmussen, 2015). The 10 most targeted organisations accounted for more than 75 % of the phishing attacks. The number of phishing campaigns stayed constant compared to the first half of 2014 but is at its highest number since 2009.

95.321 unique domains were involved in phishing campaigns in the second half of 2014 and 70 % percent of those campaigns were hosted under the TLDs *.com, .tk, .pw, .cf* and *.net*. In comparison, only 432 unique domains in the *.nl* zone were involved. 28 % of the total number of involved domains were registered only with a malicious purpose in mind. This means that 72 % of the domains are owned by legitimate users whose server got

compromised to host the phish. Most of the domains which were registered for malicious purposes were registered by Chinese phishers (84 %). One third of these domains were registered at Chinese registrars and half of the registrations were made at registrars in the USA. Only 1,9 % of the registered phishing domains contained a brand-name or a misspelled brand name. 6 % of phishing attacks were hosted on subdomains ran by one of over 800 subdomain providers. These subdomains are often free of charge, allow anonymous registration and many have lax takedown procedures. A phishing website is on average reachable for almost 30 hours and half of the websites have an uptime of over 10 hours.

Initiators of phishing campaigns use botnets to increase the resilience against take-down attempts (Holz et al., 2008; Gu et al., 2008a). Therefore they can share similar DNS characteristics as botnets.

Chapter 3

Detecting Malicious Domain Names

Because of the wide spread of botnets and phishing campaigns and their harmful attacks, researchers are trying to find ways to detect them and to take them down. They are analysing the initial infection and propagation of bots by setting up honeynets in order to monitor their communication. Phishing domains are detected through automatic mail filters, analysing the email content or through browser toolbars (Fette et al., 2007).

Many researchers are focusing on the communication patterns of bots. This includes the communication of bots with their peers, with a C&C server or with an attacked target. One approach is to observe only the frequency of bot communication, without taking the actual content of the communication into account (AsSadhan et al., 2009; Gu et al., 2008a). Also, the spatial attributes such as the location of bots is one common feature (Stalmans et al., 2012; Gu et al., 2008b). Other researchers look into the communication protocols and analyse the content or flow-patterns of the transferred packets (e.g. Mazzariello (2008); Chen et al. (2010)).

The approach that is the most relevant for this paper is the communication of bots with other, not compromised services - especially DNS servers. This section categorises detection techniques which are based on DNS characteristics and extracts relevant features. Domains that are involved in botnets share similar attributes with domains that are used in other malicious activities such as phishing and scam (Hao et al., 2010). Thus, the described techniques are not only relevant for detecting botnets but for discovering malicious domains in general. Besides DNS traffic patterns, phishing domains can be detected based on domain name features and domain registration information as well (Fette et al., 2007).

DNS base detection techniques have in common that they all rely on the observation of DNS request and response packets sent and received by clients requesting the IP address of a malicious domain. The detection techniques

differ among others in their feature selection and in their deployed location in a network. We categorise DNS detection techniques as follows:

- *Vantage Point*: The location at which DNS traffic is observed has an impact on the collected data and its features. Locations include *hosts*, *local RDNS and Internet gateways*, *RDNS server on ISP level* and *name servers at TLD level*.
- *Feature Selection*: Features describe typical characteristics that can be used to identify bots and malicious domains. The following feature sets have been used previously:
  - *Temporal Features*
  - *Spatial Features*
  - *DNS Record Features*
  - *Domain Registration Features*
  - *Domain Name Features*
  - *Domain Content Features*
- *Data Mining Techniques*: In order to detect malicious domains automatically, different data mining and machine learning approaches have been proposed.

3.1 Vantage Point

It is important to define where DNS traffic can be observed. Depending on the location, different features can be observed.

First, DNS traffic can be collected at the lowest level of the DNS hierarchy - at the host (Morales et al., 2009). Every single DNS request can be analysed, without caches in between. Additionally, IRC, HTTP or P2P communication can be observed as well. To achieve a holistic view of a botnet, observed information from many hosts has to be collected at a single point.

Second, DNS traffic can be observed in a LAN, for example at a network gateway or at a local RDNS server (Choi et al., 2009; Zhao et al., 2013; Gu et al., 2008b; Jiang et al., 2010; Villamarín-Salomón and Brustoloni, 2008). There, a larger number of DNS requests can be observed at one central point. Observations at a local network gateway have further the advantage that other bot traffic, which is not related to DNS, can be collected as well.

RDNS server are also widely deployed in networks of Internet Service Providers (ISP). Thus, it makes sense to observe DNS traffic at this location as well (see Lee and Lee (2014); Stalmans et al. (2012); Nagaraja et al. (2010); Lee et al. (2010); Futai et al. (2013); Perdisci et al. (2009); Bilge

et al. (2011); Antonakakis et al. (2010); Nazario and Holz (2008); Yarochkin et al. (2013); Antonakakis et al. (2012)). ISPs have often a large number of customers which send many DNS requests every second. For example Perdisci et al. (2009) have observed 2.5 million queries at an ISP's RDNS server every day.

The location which is most relevant for this Master-thesis is an observation at the level of an authoritative name server (AuthNS). Compared to the vantage points described above, AuthNSs have a global view on the Internet. They not only collect DNS traffic from one host, one LAN or one ISP, limited to a location or area, but can gather data about one domain worldwide. Despite this advantage, this vantage point comes with some drawbacks as well. First, the authority for one top level domain could make it harder to detect botnets that rely on multiple top level domains. For example, the *Torpig* botnet relies mainly on *.com* domains but uses *.net* domains as a backup as well (Stone-Gross et al., 2009). Second, because of the high position in the DNS hierarchy, DNS caching comes stronger into effect (Antonakakis et al., 2011).

Besides vantage points within the DNS hierarchy, detection techniques can be deployed at DNS related services as well. Ramachandran et al. (2006) have proposed an observation of DNS Blacklist (DNSBL) traffic. Botnet C&C servers and exploit kits query DNSBLs before an attack or a SPAM campaign to determine which domains and bots are black listed. Phishing domains can be detected at the receiving mail server or in the browser of a user.

3.2 Feature Selection

Features describe characteristics of malicious domain names that are distinct from benign domain names. Researchers have identified features based on the DNS requests of a bot, on the DNS specific characteristics of a domain, on the domain name itself as well as data from the WHOIS database. In the following paragraphs, previously identified features are described.

Temporal Features Temporal features describe characteristics that are related to the frequency of DNS requests, their distribution over time, and their relationship to previous or following DNS requests. For example, Lee and Lee (2014) have observed DNS requests of a host for a known malicious domain in order to identify previously unknown malicious domain names. They have analysed previous and following DNS requests by this host under the assumption that an infected bot will query multiple malicious domains over time. Also, by observing hosts with similar sequential DNS patterns, they expect to detect unknown bots as well (see also Villamarín-Salomón and Brustoloni (2009); Lee et al. (2010)). For this thesis, we usually cannot

observe the queries of the hosts directly. Nevertheless, the same observations can be made of recursive resolver queries to some extent as well .

The frequency and their daily similarity of DNS requests by bots has been observed by Choi et al. (2009); Stone-Gross et al. (2009); Bilge et al. (2011). For example, Stone-Gross et al. (2009) have observed that bots of the Torpig botnet contact their C&C servers every 20 minutes. The DNS request growth rate has been used by Perdisci et al. (2009) to identify DNS requests for malicious domains and Hao et al. (2010) have discovered that the number of DNS requests for malicious domains increase more rapidly after registration for benign domains. Antonakakis et al. (2011) have monitored DNS traffic at AuthNS level and have assumed that DNS requests for a malicious domain from a popular RDNS server are more likely than from a rather unpopular RDNS server.

The similarity of domain names that have been generated by a DGA is measured by Schiavoni et al. (2014). They observe the IP-addresses that have been assigned to two domains over time. In case two domains resolve in certain period to the same IP addresses, both domains most likely belong to the same botnet.

Li et al. (2013) have discovered that bots often query two DNS servers at once. Often, hosts have two DNS servers entries for reliability reasons. While benign hosts usually only query one DNS server at a time, bots seem to query both DNS servers at once. It could be possible to observe this feature at ccTLD level as well. When a bot queries two RDNS-servers at once, both servers should query the ccTLD NS shortly after (under the assumption that non of the them have cached the DNS entry).

Another feature that can be observed over time are failed DNS requests. When a host queries a domain name that does not exist, an NXDOMAIN domain response is sent from the NS to the host. FFSNs have a high number of failed DNS lookups because of not-reliable flux-agents (Holz et al., 2008) and bots that use DGAs to contact C&C servers query a high number of not existing domains as well (Zhu et al., 2009). In case a DGA generates new domains on 2nd domain level, then a RDNS queries the domain name every time at the responsible AuthNS of the TLD. Therefore, domains generated by DGAs can be detected on TLD level.

In general, temporal features have the disadvantage that they have to be observed over time, for example over one day. Therefore detection mechanisms based on those features cannot discover malicious domains in real time (Huang et al., 2010). Caching has an influence on the observation of temporal attributes at an AuthNS. However, if a botnet spreads fast or if a phishing campaign reaches many users, then many DNS queries from RDNSs can be observed at an AuthNS as well.

Spatial Features This set of features describes characteristics related to the geographic distribution and the Autonomous Systems (AS) or the sub-network of DNS requests. Spatial features can be observed at the source of an DNS request or at the location of the server to which an A-record or an NS-record is pointing.

Stalmans et al. (2012); Bilge et al. (2011) have assumed that benign domains refer to servers within the same time zone whereas servers hosting malign domains are distributed all over the world. Other researchers have used the Autonomous System (AS) in which a server is hosted as an indicator (Perdisci et al., 2009). These features can be observed on ccTLD as well (Antonakakis et al., 2011). For example, Hao et al. (2010) have observed that different malign domains are queried from the same AS. Furthermore, they have shown that benign domains are queried over time by the same set of resolvers whereas the sources of DNS requests for malign domains is varying stronger.

DNS Record Features Here, features are listed that can be extracted directly from the DNS records. One attribute of FFSN is that more A-records are returned than for CDN or round-robin domains. According to Holz et al. (2008), FFSN return five or more A-records whereas the average of benign domains lays by only three. Also, the number of NS servers that are responsible for a domain, their number of A-records, and their TTL are significant (Futai et al., 2013). Furthermore, the TTL of malicious domains is shorter than 150 seconds (Mahjoub et al., 2014; Perdisci et al., 2009). Bilge et al. (2011) calculate further the average TTL of a domain, the number of distinct TTL-values, and the number of changes in TTL for a domain.

Many of these features can additionally be observed over time and geographical distribution as well. For example the A records for a domain or an NS are replaced frequently. This can be used to detect malicious domains as well.

At an AuthNS of TLDs, only the number of NSs for a domain can be observed and only if the NS is in the zone of the TLD, its IP address is known by the AuthNS as well.

Domain Registration Features Features about the domain registration can be gathered from a WHOIS database. Kheir et al. (2014) use the elapsed time between the registration date, the creation date, the date of the last modification and the remaining time before a domain expires to identify malicious domains. Yarochkin et al. (2013) compare WHOIS registration information (e.g. phone number, contact name, contact address) to identify domains that belong to the same botnet. The age of a domain name is used by Zhang et al. (2007) to discover newly registered phishing domains.

Domain Name Features Especially domains that are generated by DGAs have often unique characteristics that help researchers to identify them. The number of alphabetical and numerical characters, their ratio, the total length, and special character sequences have been used by Bilge et al. (2011); Frosch et al. (2013); Antonakakis et al. (2010); Yarochkin et al. (2013). Schiavoni et al. (2014) have further measured how good a domain name can be pronounced. Domains generated by a DGA usually are harder to pronounce than benign domains. Zhang et al. (2007) check, if the URL contains special characters like '@' or '.' to identify phishing websites. Fette et al. (2007) use the number of dots in an URL as an indicator to detect phishing domains.

Domain Content Features Additionally, Kheir et al. (2014) have analysed the content that is hosted on suspicious domains. They have looked at the HTML code, *robot.txt* files, hosted images and number of CSS stylesheets to distinguish between benign and malign domains. Furthermore, they measured the popularity of a website counting the number of outbound links to social networks, the number of inbound links from social networks, and the domain's Google pagerank. Analysis of web-content is used to detect phishing domains as well (Zeydan et al., 2014). For example, the use of well known images of brand-logos or the use of forms can indicate a phishing attack (Zhang et al., 2007).

3.3 Data Mining Techniques

Different data-mining methods have been used in previous research. The general goal is to feed observed DNS data to a classifier that determines, based on features described above, whether the domain is used for malicious purposes or not. Therefore, the classifier usually first has to be trained with a dataset that includes information of already known benign and malign domains. The precision of the classification algorithm can be defined based on the number of correctly identified malicious domains (called *True Positives* - TP) and the number of legitimate domains that erroneously have been identified as malicious domains (called *False Positive* - FP). The higher TP and the lower FP the better the algorithms works.

In this section, effective data-mining methods from previous research are categorised and their use-cases are described. Most of the methods rely on training data that provide a ground truth about the behaviour of malicious and benign domain names. Hence, common training data-sets are listed as well. This section helps us to select an adequate algorithm for *SIDekICk*.

Decision Trees A decision tree is a classification algorithm that extracts features from a data set where each object in the set is represented as a tuple. Futai et al. (2013) use the J48 algorithm to identify the features

in a training set consisting of domain samples of the Alexa 1.000[1] list and domain black lists. Then, real-time data is fed to the classification algorithm to detect unknown malicious domains. The J48 Decision Tree classifier out-performs Support Vector Machines (SVM), Logistic Regression (LR), Bayesian Network and Random Forest with a TP rate of 95,5 % and FP of 0,03 % even with a low number of known FFSN domains in the training set. These results can be confirmed by Perdisci et al. (2009) and Bilge et al. (2011). They both have achieved a TP rate of over 99 % and a FP rate of 0,3 %. Antonakakis et al. (2010) have used the Logit-Boost strategy for their Decision Tree to achieve a TP-rate of 96,8 % and a FP-rate of 0,38 %. Another variation of Decision Trees are random forests, as used by Antonakakis et al. (2011). Random forests avoid over-fitting, and increase the overall performance of the final model. Antonakakis et al. (2011) have achieved a TP rate of 98,4 % and a FP rate of 0,3 %. They have evaluated their results against Naive Bayes, k-nearest neighbour, SVMs, neural networks and random committee classifiers.

Decision Trees have further the advantage that they are in general easy to interpret by a human which makes the classification more comprehensible.

X-Means Clustering X-Means clustering is an extension of the K-Means clustering algorithm. The K-Means algorithm does not rely on pre-classified training data but tries to find partitions for data sets, depending on the similarity of their features. The X-Means algorithm works as the K-Means algorithm but after the algorithm is finished it tries to split the clusters in two additional separate clusters. Thereby clusters are found more reliable and faster (Pelleg et al., 2000). Antonakakis et al. (2010) have used the X-Means algorithm to identify clusters during the training phase.

Spatial Correlation Research that uses geographical features to detect malicious domains have used *spatial auto correlation* to analyse differences in malign and benign DNS requests (Stalmans et al., 2012). Spatial auto correlation measures the dependence of points in two-dimensional space. The researchers used *Moran's coefficient* to detect FFSNs with a TP-rate of around 99 % and a FP-rate of 6 %. They used the 1.000 most popular websites, based on the Alexa statistics, and domains from FFSN-trackers to train the classifier.

Bayesian Network Classifier This classifier has been used by Huang et al. (2010) to identify FFSN, based on the geographic distribution of the infected hosts. The classifier has been trained with the K2 algorithm. Also, Kheir et al. (2014) use this classifier to assess if domain are legitimately listed on a DNS blacklist. They achieve a TP rate of 99,02 % and a FP rate of 0,98

[1] www.alexa.com

%. Bayesian Networks are graphical representations of interdependencies between different probabilities where nodes represent random variables and the edges represent assumptions about conditional dependencies (Murphy, 1998). The K2 algorithm makes heuristic searches to find the best structure of the Bayesian network Cooper and Herskovits (1990).

Support Vector Machine (SVM) A SVM is a classifier which is first trained with labelled data sets and then tries to separate these data sets in an n-dimensional space with an $n-1$ dimensional hyperplane. Each object in a data set is mapped to a point in the space. The hyperplane then separates these points as good as possible to achieve a clear classification (Statsoft, 1995). Martinez-Bea et al. (2013) have built a real-time classifier based on a SVM to detects FFSN with a TP-rate of 98,78 % and a FP-rate of 1,22 %.

Naive Bayes Classifier Naive Bayes Classifiers are a very simplified classification technique based on Bayes' theorem. It works under the assumption that each feature is independently contributing to the final classification of an object. Although the classifier oversimplifies real-world situations, it has proven to generate valuable results in the past (Chaney and Blei, 2012). Passerini et al. (2008) used the Naive Bayes Classifier to build their FFSN detection system FluXOR.

k-Nearest Neighbour (kNN) kNN is a supervised learning method that works under the assumption that objects in a vector space are more similar the closer they are together. It has been used by Frosch et al. (2013) to detect malicious botnet domains. In order to compute the distance between the objects, the Euclidean Distance has be used. They have achieved a TP rate of 94 % and a FP rate of 1,54 %. Schiavoni et al. (2014) have used the *Mahalanobis distance* to measure the distance between features of a previous unseen domain name and the expected features of a set of benign domains. The *Mahalanobis distance* takes correlation between the features into account as well.

Hidden Markov Model (HMM) In order to detect domains that are generated by a DGA, Antonakakis et al. (2012) use a HMM. A HMM is a supervised learning classifier applied to sequence observations over time. Each state is not directly visible, but only its output (Ramage, 2007). In their paper, Antonakakis et al. (2012) use one DGA to train an HMM where each DGA has its own HMM. If a new domain is detected, it is used as an input to the HMMs and the HMMs will return with which probability the domain is part of an DGA. The HMMs see the domains as string of

characters where the character is an output at a certain state. The number of hidden states has been set to the average length of the training set.

Graph Based Methods that try to identify malicious domains based on sequential correlations construct graphs to build a representation of the relationship of DNS queries. Lee and Lee (2014) construct a *Domain Name Travel Graph* where each requested domain is represented by a node. Edges between the nodes represent domain names that have been consequentially queried by the same client. Then, the correlation between the domains is determined by the *Jaccard index*. The more clients queried domains in the same order, the higher the correlation becomes between the domain. Edges with a low correlation are removed. Thereby clusters of domain queries are built. Whether a cluster represents malicious DNS activities is determined with the help of a domain blacklist. A cluster represents malicious bot activity if it contains a known malicious domain. Domains that have a high correlation in the graph with this domain are then most likely malign domains themselves. Lee and Lee (2014) have achieved a TP-rate of 99 % and a FP-rate of 0,5 %.

Jiang et al. (2010) built DNS failure graphs, where the nodes are hosts and domains. Each host that has at least one failed DNS query for a domain is connected with this domain through an edge. The graph is clustered into sub-graphs with the *Non-negative matrix factorisation* by using re-engineered DGAs as training sets.

3.4 Training Data

Most techniques described above rely on supervised learning of the algorithms. There, the algorithm needs data sets that are already classified as benign or malign. Different data sets are used, depending on the vantage point. The most common ones are listed below.

Benign Domains The most common way to get information about the behaviour of benign domains is to rely on the provider of web traffic data *Alexa Internet*[2] (Bilge et al., 2011; Frosch et al., 2013; Kheir et al., 2014). It provides lists of the most popular websites worldwide, filtered by a region or a country. Based on these lists and depending on the required features, information, for example about their A-records, WHOIS information or the location of their servers, can be retrieved. Besides using lists provided by Alexa, Hao et al. (2010) have additionally created a set of supposedly benign domains that do not get many daily queries. Therefore, they have selected a sample of domains that have been queried more than 20 times on the first

[2] www.alexa.com

and last day of a two month long period. Domains that were among the most popular domains of Alexa or were listed on blacklists were discarded.

Malicious Domains A set of known malicious domains is necessary to help the classification algorithm to learn the difference to benign domains (Bilge et al., 2011; Futai et al., 2013; Kheir et al., 2014). There are already many services which provide lists of malicious domains, including *Spamhouse*[3], *Malwarebytes*[4], *Malwaredomains.com*[5], *Malwaredomainlist.com*[6], the *Security Information Exchange*[7], *Netcraft* [8], *virustotal* [9] and *Abuse.ch*[10]. The latter provides a database that lists malicious domains used by the botnets ZeuS, SpyEye, Palevo and Feodo.

These blacklists collect malicious domains from different sources. For example *Netcraft* uses among others a browser toolbar to classify domain names automatically. *virustotal* relies on multiple anti-virus engines and website scanners to classify domain names. These scanners analyse for example, whether malicious scripts are hosted on the website or malware is delivered.

[3] www.spamhaus.org/dbl
[4] hosts-file.net
[5] www.malwaredomains.com
[6] www.malwaredomainlist.com
[7] www.dnsdb.info
[8] www.netcraft.com
[9] www.virustotal.com
[10] www.abuse.ch

Chapter 4

Domain Names in *.nl*

The *.nl* zone is not known for many malicious domains. The Global Phishing survey counted only 432 *.nl* domain names that were involved in phishing campaigns in the second half of 2014 and the domain-blacklist, provided by *malwaredomains.com*, has listed at the end of June 2015 only 40 unique *.nl* domains that were recently involved in the distribution of malware (Aaron and Rasmussen, 2015). 26 of those domains have already been listed in 2014 or earlier and 30 domains have been re-submitted to the blacklist at least two times.

Because the majority of malicious domains are registered in *.com, .net* or *.tk*, most of existing research focuses on these domains and little is known about malicious domains in *.nl*. This chapter provides an overview of the known malicious domains in *.nl*, which are listed by third parties or have been discovered by researchers of SIDN Labs during previous projects and manual observations. It describes the purpose of the malicious domains and how many people are or have been affected by their malicious activity. This knowledge is necessary to better estimate whether these domains give a general picture of malicious activity in *.nl* and to make assumptions of the expected DNS characteristics and behaviours. Additionally, it is important to know, how malign domains differ from the majority of the *.nl* domains that are used for legitimate purposes.

First, we describe our data set that is examined and which is partially used to build a classifier for *SIDekICk*. It includes typical benign domains in *.nl* followed by a summary of known malicious domain names. Then, we describe metrics by which the domain names are compared. We select features that have been discussed in existing research (see Chapter 3) and propose new possible approaches to detect malicious domain names. We focus on features that can be observed directly at a ccTLD registry. The last section explains for each metric the difference between benign and malign domains.

4.1 Data Sets

4.1.1 Benign Domain Names in *.nl*

The set of benign domains contains the 1.000 most popular *.nl* domains according to Alexa. Besides these domains, we have additionally added domain names to the list that belong to the most popular web-hosting firms. Many data centre providers are located in the Netherlands, especially in the region around the city of Amsterdam which has attracted many web-hosting services (Association, 2015). These hosting companies provide name servers for the websites of their customers. As soon as users visit a site hosted at one of the companies, their stub-resolver queries the webhoster's name server. Therefore, they need to resolve the domain name of the name server and contact the servers of SIDN. Ergo, domain names of popular webhosters receive a high number of queries as well and show up among the most queried domain names. In total, we have created a set of 1.300 popular domain names. In May 2015, these domains were responsible for 46,4 % of the total number of queries and received on average 33.735,55 queries per day (14.607 median). On the other side, 98 % of the 5,3 million unique domain names that have been queried in May 2015 have received less than 100 queries per day on average. These domains are further referred to as the long tail. Figure 4.1 depicts the distribution of the domains by the number of daily queries. It can be seen that the majority of domains receive between 10 and 1000 queries. In order to get a sample of this long tail, Hao et al. (2010) created a set of benign domains that did not show up in their set of popular benign domains, not in their set of malign domains, and have been queried more than 20 times at the start and end of a period of two months. For *.nl* we created the same set (start date 2015-03-02, end date 2015-04-30) but limited the size of the sample to 300. This was necessary because the knowledge of malicious domains in *.nl* is limited and we had to validate the benign domains manually. From this list 9 domains were removed because they were not reachable anymore and 4 domains appeared on the blacklist of *virustotal* and *Sucuri*[1].

Additional to this set, 223 domains have been selected to represent benign domain names that have been registered recently. These domains have been registered at the 2015-04-07 and the 2015-04-08 and still received at least 20 queries two months later. They account only for 3,9 % of the total number of domains that have been registered on these days. 19 domains lead to bogus webshops, redirected to obvious scams or were classified by *virustotal* as malicious and were therefore removed from the set. On average 2.532 domain names have been registered in May 2015 daily. Newly registered domains and domains from the long tail received on average 31 queries per day.

[1] `https://sitecheck.sucuri.net/`

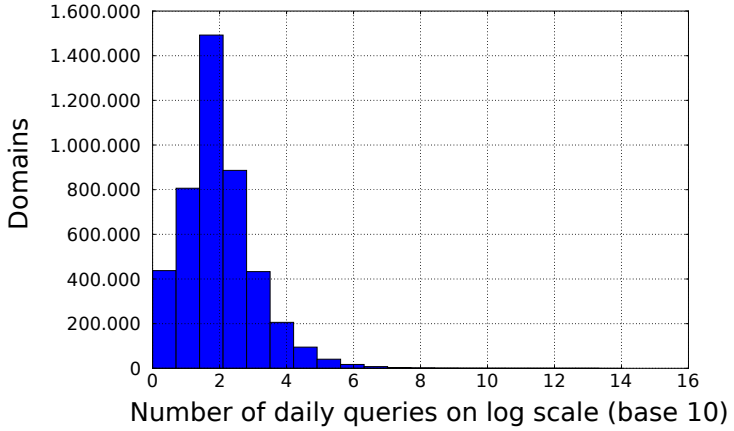

Figure 4.1: Histogram that represents the distribution of domain names by their daily queries.

4.1.2 Known Malicious Domains

Botnets - controlled with *.nl* Domain Names

In this section, domains are analysed that have been or are still used for the command and control of botnets. The Dutch company *Quarantainenet* develops software to monitor local network traffic and has provided SIDN with a list of domain names that have been categorised as malicious by their detection software. This list contains 49 distinct domain names. 18 additional domains come from various other sources like the binary analysis service *totalhash*[2] or *cybercrime-tracker.net* and 15 domains are in the sinkhole of SIDN Labs. Domains in the sinkhole have been discovered by staff of SIDN or by other sources and have been registered by SIDN when the former owner dropped the claim. Most of the domains receive requests from clients that are still infected and try to contact their former C&C server. In total, the data-set contains 82 (former) botnet domains of which only 8 have received more than 20 daily queries on average in June 2015. We assume, that the other domains are very likely not active anymore. Below, we describe the botnet types to which the domains belong and give a brief overview about the activity of the domains since November 2014. Since this date, the ENTRADA platform aggregates the number of queries for each domain name daily such that the number of queries can be observed fast.

Figure 4.2 shows the distribution of known botnet domains. Over half of the domains belong either to the ZeuS or to the Pushdo botnet. Four domains in the sinkhole were identified as Andromeda domains and eight are contacted by Backdoor-Flashback bots.

[2]totalhash.com

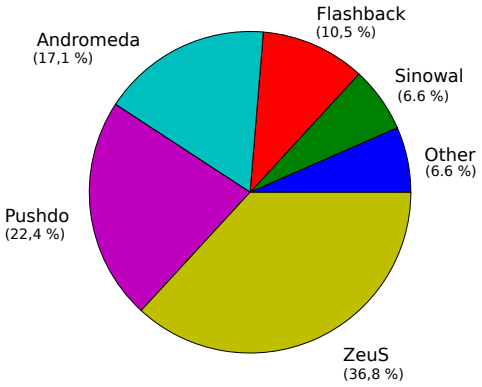

Figure 4.2: Classes of known botnet domains

ZeuS This is a crimeware toolkit that was the number one botkit in 2010 with over 3,6 million infections in the US alone (Binsalleeh et al., 2010). It gained popularity due its capability to easily steal banking account information and credit card numbers by logging keyboard inputs. Infected clients communicate usually with three different URLs. One URL provides the configuration file, another one links to the latest version of the ZeuS binary and the last is used to upload the stolen information. These three URLs can be distributed over different domains. Kryptik is another variant of the ZeuS malware.

Pushdo The Pushdo botnet was first discovered in 2007. It usually gets shipped with the Cutwail module that is used to launch large SPAM campaigns (Decker et al., 2009). A special feature of Pushdo is the DNS resolver that is built directly into the bot-software and which bypasses the pre-configured DNS resolver of the infected client. Therefore, we see DNS-queries of the infected clients directly at our name servers. Pushdo uses the internal resolver to resolve IP addresses of targeted mail-servers and for connectivity checks. According to Decker et al. (2009), Pushdo usually uses for the communication with the C&C server hardcoded IP addresses. However, we have still found domains associated with this bot in different datasets. Very likely, other versions of the bot have changed their communication behaviour towards domain names.

Andromeda Andromeda is a modular bot that can be used to load other malicious software onto the infected system. It has been updated several times and has been spread through links to malicious websites in emails and through malicious attachments (Rascagnres, 2015). Version 2.9 of the bot uses Google's open DNS resolvers to perform lookups for the IP address of C&C servers. The communication with the server is Base64 encoded and encrypted with the RC4 cipher (Kimberly, 2014).

Backdoor-Flashback This botnet is only targeting computers running the operating system Mac OS X. It has infected over 650.000 machines in 2012 by pretending to be a Flash Player update (Soumenkov, 2012). Flashback installs a backdoor that enables the attacker to control the infected machine.

Other Malware Two domains are used by the Pony botnet, which is designed to misuse infected machines for example to mine the crypto-currency Bitcoin (Brook, 2014). Five domains belong to the Torpig/Sinowal botnet, which is mainly used to steal banking and credit card information. It is distributed through the Meebot Rootkit. A Torpig bot contacts its C&C server every 20 minutes to upload stolen information (Stone-Gross et al., 2011). The other domains are used by more generic trojans that were not assigned to known botnets.

The activity of the known botnet domains varies strongly. For this analysis, it is examined in which week the domain was queried most often and then the average number of queries is calculated. This number is compared to the average number of queries between 2015-06-15 and 2015-06-21. Thereby we can estimate, when the domain was most active and whether it is still used recently.

11 domains never received more than 20 queries a day. It is assumed that those domains were either active before November 2014 or were never actively used for C&C but only showed up in an analysed malware binary. 7 domains were assigned to the Andromeda bot and 4 to the Torpig/Sinowal bot. These domains are left out in the further examination.

14 domains received on average more than 1.000 queries per day during their period of highest activities. 7 domains belong to the Pushdo botnet, 3 to the Andromeda botnet, and 1 to the ZeuS/Kryptik net. The most queried domain belongs to the Andromeda botnet with over 100.000 queries. 18 domains got queried between 500 and 1.000 times. 8 domains of this set belong to the Mac OS X trojan Backdoor.Flashback and 5 domains belong to the ZeuS network. 18 domains received between 100 and 500 queries per day and belong mostly to ZeuS (11 domains) and Pushdo (6 domains).

On their day of the query peak, Andromeda domains were queried by over 4.300 unique sources, Pushdo domains by more than 1.000 and BackDoor.Flashback by almost 1.000. For most domains caching and the use of anycast at resolvers hinders us to pin down the actual number of infected machines. Nevertheless, it can be seen that a significant number of clients are still infected, despite the age of the botkits. The webserver in the sinkhole allows us to count the actual number of unique IP addresses that try connect to sinkholed domains. For the most popular Andromeda domain, more than 476.509 unique IP addresses were counted in April 2015. The number is still only a rough estimation due to dynamic IP addresses and

Network Address Translation.

At least 25 domains have been registered for a malicious purpose. This can be derived from the fact that most of these domains consist of a random looking numbers and characters and that they are registered with bogus information. 9 domains belong to Pushdo, 8 to Backdoor-Flashback and 5 to Andromeda. Every other domain is very likely owned by legitimate registrants whose webserver has been compromised to serve in a botnet.

Phishing Campaigns

SIDN receives a continuous feed of *.nl* phishing domains from the company *Netcraft*. For this thesis, all domains are considered that have been reported from December 2014 on. In these 7 months, 1.923 unique domains have been listed. The majority of phishing domains were registered more than one year before they were actually used in a phishing campaign (82 %). Only 6 % of the domains were used in a phishing campaign less than one week after they have been registered. This leads us to the conclusion that the majority of phishing campaigns are rather hosted on compromised websites than on websites that are registered for malicious purposes. This number is even 10 points higher than the global share, as reported by Aaron and Rasmussen (2015).

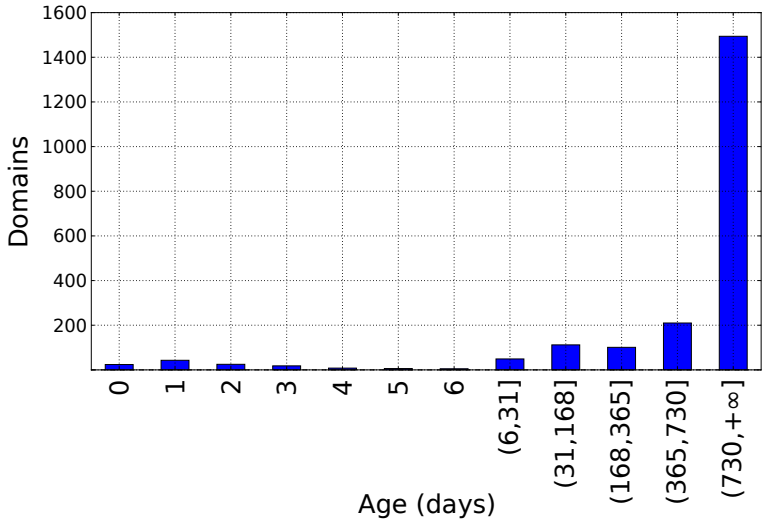

Figure 4.3: Age of phishing domains at the day they got reported (in days). The five bars on the right are grouped by age-intervals.

268 domains where used at least two times for phishing campaigns and 9 domains where used more than 5 times during the 7 month period. Among those domains are *blogspot.nl* (56 times), *axc.nl* (9 times), *r4u.nl* and *pg-press.nl* (both 6 times), which allow to host a website on a subdomain,

often for free. Thus, these 2nd level domains did not provide the phish but it was served on one of the subdomains created by a user of the subdomain provider.

4.2 Comparative Metrics

The previous sections have shown that knowledge of malicious activities in *.nl* is limited. Especially the set of known botnet domains is small, limited to a few botnet classes and includes many old domain names. Nevertheless, it is possible to use most of these domains to gain a better understanding of their DNS and registration characteristics which can give us insight into how they differ from legitimate domain names.

The characteristics of the described domain names are compared based on the following metrics.

- Geographic location of querying resolvers
- Relationship between small resolvers and unknown malicious domains
- Temporal characteristics of the number of queries
- Resolver lookup similarity
- Domain name server characteristics
- Subdomain characteristics
- Domain registration characteristics

They include characteristics that have been previously used to detect malicious domain names on TLD level but also on lower vantage points in the DNS hierarchy. Below, the metrics are explained in more detail.

Geographic Location of Querying Resolvers At the moment when a DNS request is processed by the name sever at SIDN, the query and parts of the response are stored in our database. Additionally, the source IP address of the query is used to determine the country where the resolver is located. The Maxmind[3] database is used to assign a country to an IP address. These geo-locations databases have shown to be accurate enough to locate an IP address on country level and can therefore be used as a reliable metric (Poese et al., 2011).

The location of the resolver is not necessarily identical with the location of the querying client. This is especially true for queries from the US. *Open Resolver Services* like *OpenDNS* and Google Public DNS often set

[3] `https://www.maxmind.com`

up their resolvers in the US and therefore distort the results towards the US. The ccTLD *.nl* is mainly focusing on the Dutch market. Therefore it is expected that the majority of queries for *.nl* domains have their origin in the Netherlands and bordering countries like Belgium and Germany.

Relationship Between Small Resolvers and Unknown Malicious Domains Botnets like ZeuS and Exploit Kits use multiple domains to communicate with their C&C server to download new malicious binaries and to upload stolen information. Also, bots might first spread within their own network infecting multiple machines behind a resolver. Therefore, it can be assumed that a resolver that queries one malicious domain also queries other malicious domains at the same time or in the future (Grier et al., 2012; Lee and Lee, 2014).

Resolvers that serve thousands of clients can send queries for thousands of different domain names every day which might make a detection of other malicious domains harder. The approach of this metric is to observe queries from resolvers with less than 250 daily requests and which have queried a malicious domain on a certain day. Thereby, we expect to reduce the number of unique domain names that need to be analysed.

Temporal Characteristics of Queries This metric measures the number of queries of benign and malign domain names over time. We compare sudden increases in queries from one day to another and over a period of several weeks. We differentiate between domain names that have been registered recently and domain names that have already a lifetime of more than one week. Query growth has already been used by Perdisci et al. (2009) to identify malicious domain names and Hao et al. (2010) showed that newly registered malicious domain names receive more queries after registration than benign.

Resolver Lookup Similarity As shown by Hao et al. (2010), malign domains differ from benign domains in the stability of the set of resolvers that query a domain name over time. Benign domains tend to be queried from the same resolvers every day whereas malign domains are more often queried from a changing set of resolvers. The similarity is measured by comparing the sets of resolvers of two consecutive days (A and B) with the formula of the *Jaccard Distance* which is defined as follows:

$$J(A, B) = \frac{|A \cap B|}{|A \cup B|} \tag{4.1}$$

Two sets contain exactly the same resolvers if the Jaccard Distance is 1 and two completely distinct sets have the value 0. To take into account that an ISP might provide multiple resolvers for their customers, we group

the resolver IP addresses by */24* (IPv4) and */52* (IPv6) subnets. The daily similarity for benign domain names is calculated during the week from the 11. to the 17. May 2015.

Domain Name Server Characteristics FFSN change the IP address of their domain names in a high frequency. AuthNSs of a TLD do not know the A-record of a domain name but only respond with the domain name of the name server which is responsible for the domain on a lower level. Therefore, we cannot observe changing A records directly. However, double-flux botnets not only change the A-records but also the IP address of the corresponding name server (Riden, 2008). In case the name server is within the *.nl* zone, SIDN keeps track of its IP address which is attached as a glue record to the DNS response. Thereby, a recursive resolver does not has to send another query to resolve the domain name of the name server.

The goal of this metric is to analyse if domains in *.nl* are misused for double fluxing. This would be a sign that *.nl* is used by more sophistic malware despite from what we have seen so far.

Subdomain Characteristics A large number of unique subdomains for a single 2nd level domain can be a sign for a domain shadowing attack (Biasini and Esler, 2015). In such an attack, we would expect a sudden increase in the number of unique subdomains for a domain name from one day to another as soon as an account of an registrant gets hacked. Furthermore, multiple resolvers would query for the subdomain for a short period. Because of the effect of caching, we would not see a query if a resolver queries a second subdomain. The subdomains might have a different IP address than the 2nd level domain if the attacker uses a different server to host the malicious content.

In order to be able to identify these domains, we count the number of unique subdomains that have been observed on two consecutive days for every *.nl* domain. Then, we examine domains that were queried for 10 times more unique subdomains than the previous day. Thereby, DNS amplification attacks, which query for pseudo random subdomains in order to target a name server, would be detected as well (SECURE64, 2014).

Domain Registration Characteristics Domain registration details can be used to identify domains that belong to the same botnet (Yarochkin et al., 2013). SIDN has restricted the access to the WHOIS domain registration data for the public and only shows information about the registry, the name servers, and whether DNSSEC is enabled for the domain name. For this project we have access to additional information like registrant, administrative contact and technical contact.

For newly registered domain names we compare the country of the registrant's address, email provider and registrar. The report by Aaron and Rasmussen (2015) showed that certain registrars are more popular among phishers than others.

4.3 Comparison

This section describes characteristics of malicious and benign *.nl* domain names using the metrics described above. We use the aforementioned malicious domains and the sets of verified, benign domains. For the evaluation, 1.810 domains are in the benign set and 1.972 are in the set for malicious domains. 49 domains of the malign set are related to botnet activities.

4.3.1 Geographic Location of Querying Resolvers

This section describes characteristics that are related to the origin of a query. We show that malicious domain names are often queried from countries that are not common for benign *.nl* domains.

Benign domains The set of benign domains is separated into two sets. The first includes very popular domains from the Alexa 1.000 list and the domains of big hosting companies. The second set includes domains that represent the long tail of domains that receive only a few queries a day. It consists of the long lived domains and the newly registered domains. This separation is based on the assumption that popular domains and domains from the long tail might have different characteristics. For example, nameservers of Dutch hosting companies might be responsible for domains outside of the *.nl* zone as well.

The ccTLD *.nl* is mainly focusing on the Dutch market. Therefore it is expected that the majority of queries for *.nl* domains have their origin in the Netherlands. This assumption is true for the second set of domain names where the majority of queries come from the Netherlands and the neighbouring countries Germany and Belgium (43,5 %, 7,4 % and 6,1 % of total queries). The large number of queries from the US can be explained with the aforementioned location of open resolvers (21,4 % of total queries). The share of queries from the US is higher for the most popular domains of *.nl* (26,5 % of total queries). They even exceed the queries from the Netherlands (13,7 % of total queries). Also, it can be noticed that queries from Russia and China have a relatively high share. Russia is the country with the 3rd and China with the 5th most queries. They account for 7,4 % and 5 % of the total number of queries. The origin of DNS queries is depicted in Figure 4.4. The size of the circles indicate the share of queries coming from each country.

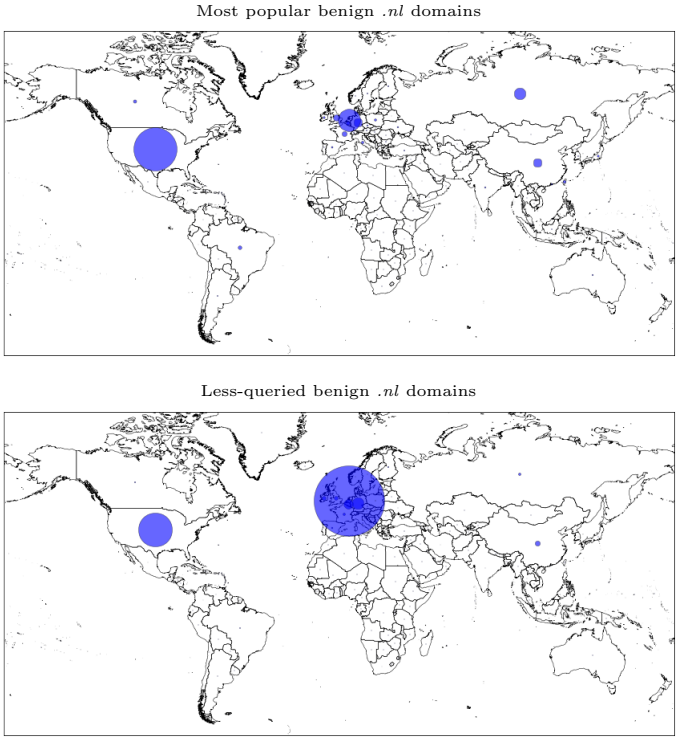

Figure 4.4: Geographical distribution of queries for benign domains

Botnet Domains Figure 4.5 shows the origin of queries for the domains of the ZeuS, Pushdo, Anromeda and Backdoor-Flashback botnet. The origins of the queries is measured on the day on which the domains had the highest number of queries and therefore were most active. It can be seen that each type of botnet has infected clients from different countries. ZeuS and Pushdo domains were queried often from the UK and Belgium. Additionally, Pushdo has many infected clients in Turkey, Taiwan and China. The Andromeda botnet receives the majority of queries from Turkey, Iran and India and the Backdoor-Flashback botnet has many infected clients in the USA, Canada and Russia.

Phishing Domains For the analysis of phishing domains, we measure the geographic distribution of queries on the day on which the domain got reported by *Netcraft*. Domains of subdomain providers are left out, because they host also many legitimate content. Figure 4.6 depicts the observed origin of the queries.

Phishing campaigns in *.nl* often target Dutch customers. For example, they try to impersonate popular Dutch banks and are using domain names like *digipas-vervangen.nl* or *uitgifte-raboscanner.nl*. Therefore, we expect

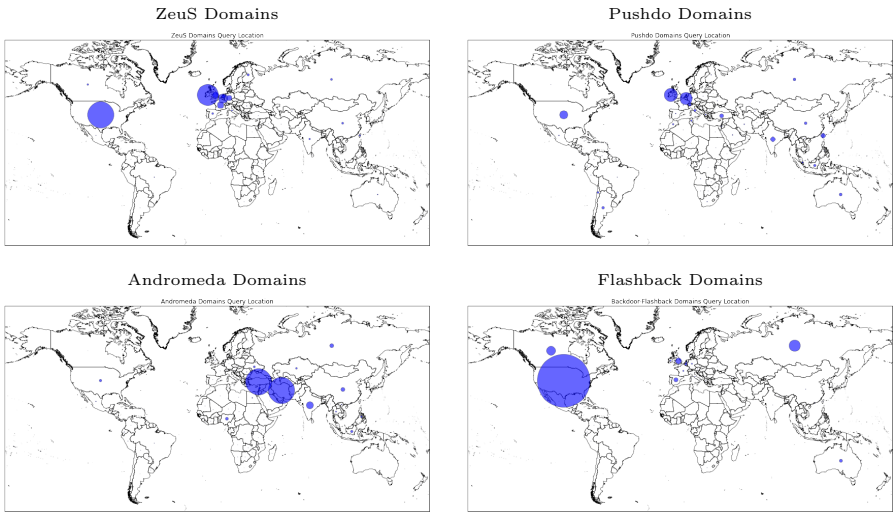

Figure 4.5: Geographical distribution of queries for botnet domains

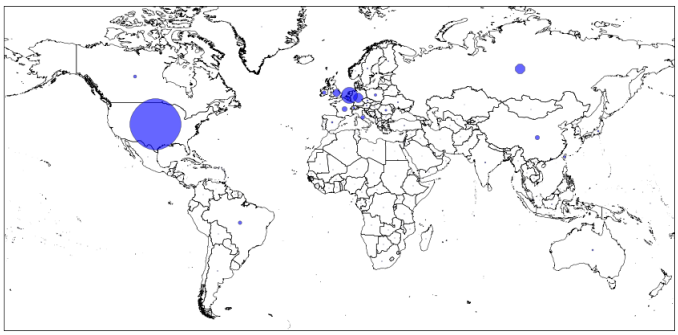

Figure 4.6: Geographical distribution of queries for phishing domains

that most visitors come from the Netherlands.

This assumption seems to be valid for some domains. The Netherlands are the second most popular country of origin behind the US. However, the total share is only 10.9 % (NL) and the rest of the queries are more equally distributed among other countries all over the world. This leads to the conclusion that phishing with *.nl* domains are not mainly targeting Dutch customers.

Results The *.nl* domains target mainly Dutch Internet users. Therefore, domains that get many queries from countries like Russia or China are rather unusual and we have shown that especially botnet domain names get queried from uncommon countries.

4.3.2 Relationship Between Small Resolvers and Unknown Malicious Domains

Observation of queries for one known malicious domain can reveal previously unknown malicious domains as well.

As an example, we take a known domain of the ZeuS botnet and select a random day at which the domain was part of malicious activity. Then, we filter for every resolver that queried this domain and sent less than 250 queries that day in total. For each resolver, we look at the other domains that have been queried. The most popular Alexa domains and domains from webhosters are excluded.

This returns over 2.000 unique domain names but only 19 domain names that were queried more than 9 times. Among these domains, 5 have been classified later by *virustotal* as malicious. 3 of them were among the top 5 queried domains and 4 of these domains were not blacklisted at the date of the observation by *Quarantainenet*. If the number of resolvers is not limited by the number of queries, only 4 domains out of the first 19 domains have been classified later by *virustotal* as malicious.

Repeating the same approach with a domain from the Pushdo botnet, even 11 of the 20 most queried domains are classified as malicious. Without the limitation of using small resolvers, only 4 domains are among the top 20 queried domains. Observations for the Backdoor-Flashback botnet reveal 5 of 20 malicious domains. This is noteworthy because in total only 8 different domains of this botnet in *.nl* are known and shows that one infected machine likely queries multiple domains. When queries from every resolver are considered no known domain of the Backdoor-Flashback botnet appears among the 20 most queried domains.

Additionally it can be observed that malicious domains from the ZeuS and Pushdo observations overlap. Thus, this approach does not necessarily only reveals domains that are part of the same botnet. It might be possible that a client is infected with different malware or that the resolver is part of a company network, where many clients with the same vulnerable software are located.

These numbers show that focusing on small resolvers is a useful approach to narrow down the number of suspicious domain names. A major limitation is the fact that botnets and exploit kits not only rely on domains from the same TLD. This is especially the case when the domains are compromised and are not registered only for malicious purposes. Compromised domains are usually chosen by the vulnerabilities of their web servers and not necessarily by the TLD.

4.3.3 Temporal Characteristics of Queries

In this section, we describe the temporal behaviour of benign and malign domain names. This includes the difference in the number of queries between two consecutive days and over a period of four weeks.

Benign Domains Benign domain names are separated into domains that are popular domains, less popular domains and domains that got recently registered. The queries of these domains are measured in May 2015. Additionally, the queries of the recently registered domains are analysed in the month of their registration.

Queries for popular domains are rather stable. Only 39 of 1.017 domains (3,8 %) have experienced a sudden query peak where the daily received queries exceeded the queries of the previous days by a factor of 2. This result is confirmed when looking at the number of queries of the first and last seven days of the month. 85 domains (8,4 %) had in the last seven days 10 % more queries than the first days whereas 241 domains (23,7 %) received less than 10 % less queries. The rest of the domains did not experience a major growth or decrease in queries. Domain names that are from the long tail experience a higher variance in daily queries. 168 of 508 domains (33,1 %) have experienced at least one peak in May 2015, 133 domains (26,2 %) experienced a decrease in queries of at least 10 % and 207 domains (40,7 %) experienced a growth of 10 % or more. Figure 4.7 shows the number of queries for three popular domains and three domains with few queries. On the top graph, weekly query patterns can be seen clearly. On the graph on the bottom, query patterns are not as regular.

Newly registered domains receive on average only a small number of queries before the registration date. On the day of the registration, the number of queries increases slightly to around 20. A few days later, more queries are received and after around 10 days queries for most of the domains become more stable (see Figure 4.8).

Before a domain gets registered, it is possible that it has been in quarantine. Domains are in put into quarantine after a *.nl* domain name has been cancelled. During this time, only the 'old' registrant can buy the domain name again. After 40 days, the domain name is again available for every interest buyer. Figure 4.9 shows the average number of queries for domains that have been registered on 8. June. We have divided the set of newly registered domains into one set which contains domains that have been released from quarantine the same day and domains that have not been in quarantine the days before. It can be observed that domains in quarantine already receive queries before they get registered and experience a stronger increase in queries after the registration than domains that have not been in quarantine beforehand. After a few days, the number of queries decreases again. One of the reasons for this behaviour can be so called *domainer*.

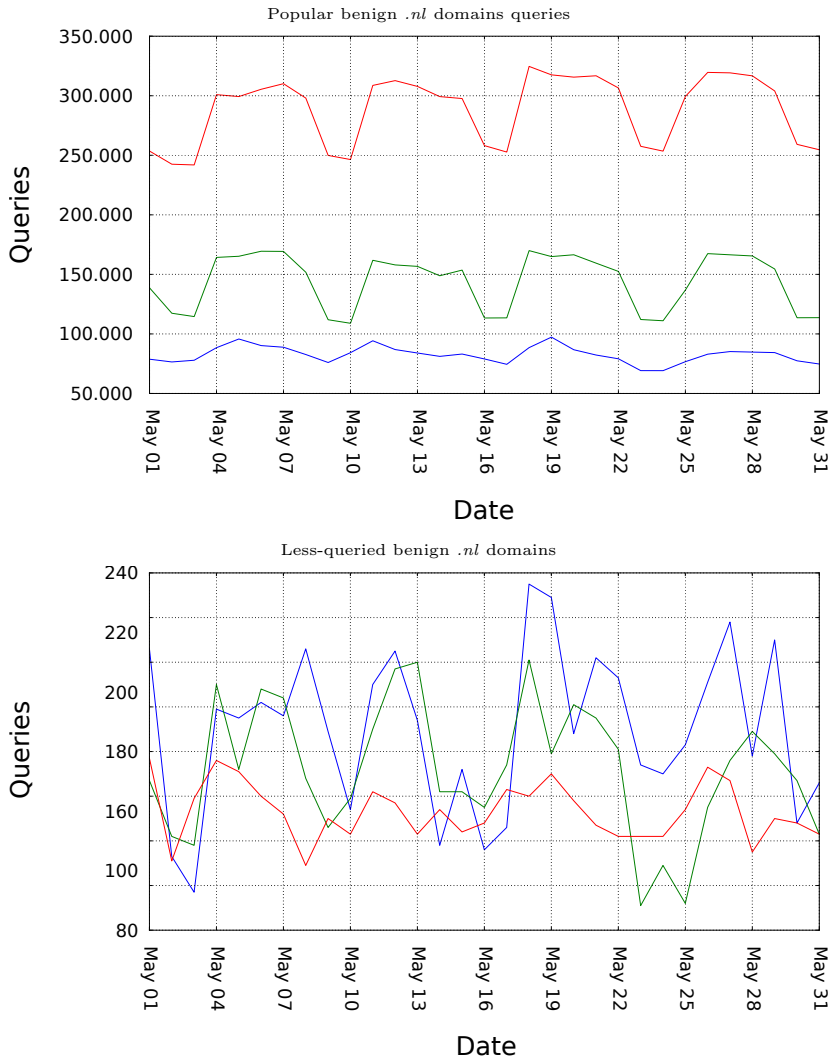

Figure 4.7: Benign domains - Number of queries in May 2015

They buy popular domain names from a registrar for a fairly low price with the expectation to resell the domain later to a considerably higher price. After the domain has been bought by a domainer, they place a parking page and offer the domain on their website for sale. This can cause the higher number queries.

On the day of the registration, quarantined domains receive on average 9,04 queries (median 6) whereas previously free domains receive 2,05 queries (median 1). In the first seven days after registration, quarantined domains receive on average 9,49 queries (median 6) and free domains 3,14 (median 2).

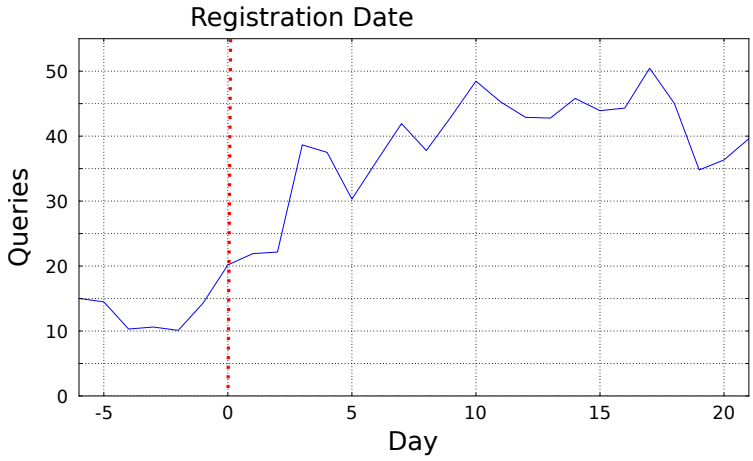

Figure 4.8: Benign domains - Average number of queries before and after registration

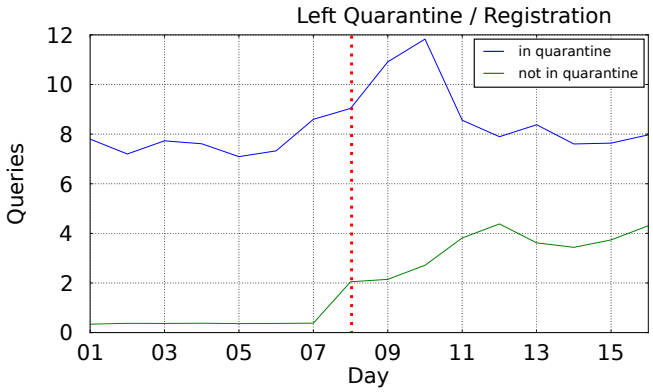

Figure 4.9: Number of queries for newly registered domains that have been released from quarantine the day of the registration and newly registered domains that have not been in quarantine beforehand. The red vertical line marks the registration date and the date on which the domain left quarantine.

Botnet Domains For each botnet type, we select domains that have experienced a significant query growth during our observation phase. We assume that this growth indicates the time at which the domain was first used for malicious purposes.

Domains of the ZeuS, the Pushdo and the Flashback-Backdoor botnet experience a steep query growth when they first get used for malicious purposes (see Figure 4.10). This growth can be seen from one day to another and the number of queries are more than two-times higher than the day before. After the first query peak, the number of queries does not grow further, but either stays stable or decreases again almost to the number of

queries before the infection. Domains of the Flashback botnet have almost identical query patterns, whereas the domains of the ZeuS bot seem mostly unrelated to each other.

The initial use of Andromeda domains and most of the Pushdo domains cannot be observed due to our limited history data set.

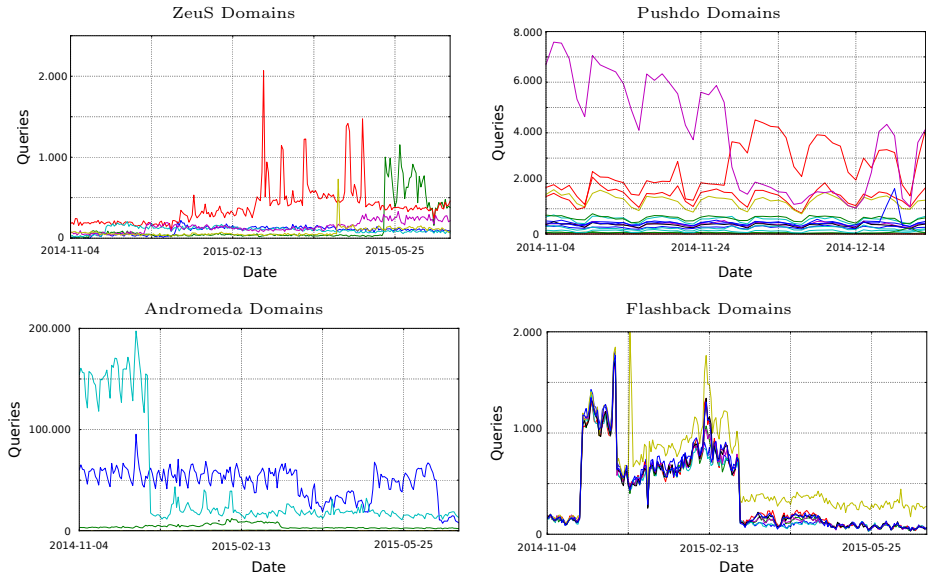

Figure 4.10: Number of queries for botnet domains

Phishing Domains Phishing domains are separated into domains that are younger than one week and domains that are older. Domains that are used in a phishing campaign within 7 days after their registration were most likely registered for this malign purpose, whereas old domains were most likely compromised domains.

For older domains, we observe the number of queries on the day they got reported and compare it with the average number of queries the domain received the week before. There, a significant increase can be observed. On average, old phishing domains received 15,5 times more queries on the day they got reported than on average one week before. Figure 4.11 shows the number of queries of a sample set of hijacked phishing domains, 20 days before and 10 after the reporting date. The reporting date is marked by a vertical red line. For these domains, a high increase in queries can be observed on the reporting date or the day before. A few days after the phish got reported, the number of queries decrease again.

Figure 4.12 shows the average number of queries that new phishing domains received 6 days before and 21 days after their registration. Compared to new benign domains (see Figure 4.8 on page 45), the number of queries

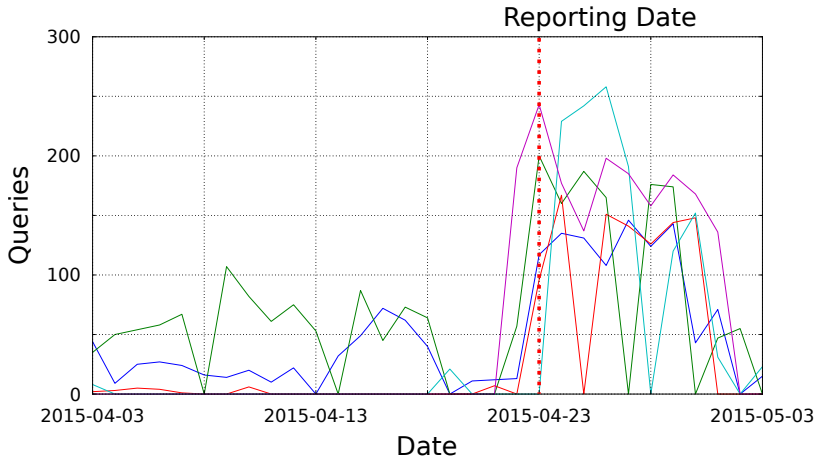

Figure 4.11: Hijacked phishing domains - Number of queries before and after reporting date

increases more rapidly and drops again after a few days, most likely because they have been listed on a blacklist or because the phisher ended the campaign.

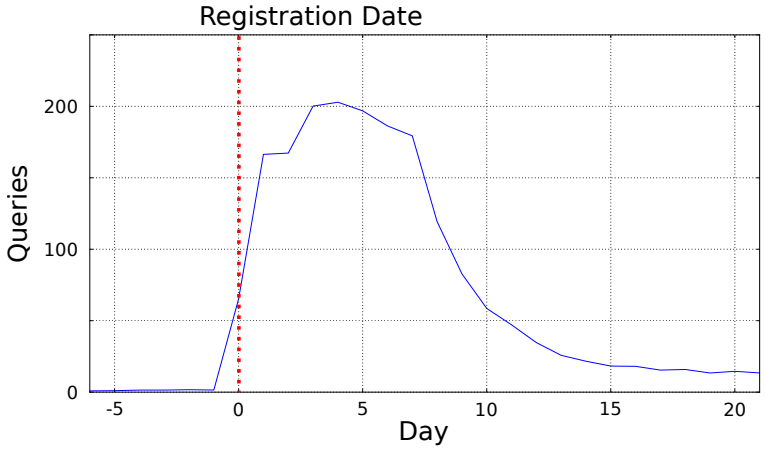

Figure 4.12: New phishing domains - Average number of queries before and after registration

A rapid growth in DNS queries might not only have malign reasons. Unpopular domains might host content that suddenly goes viral and is spread through social networks or a rather popular website moves to another domain which causes a steep increase in queries after registration. Also, modern browsers resolve domain names that they find in a website automatically to improve performance (so called *DNS prefetching*). Even this behaviour might cause peaks in traffic (Lloyd, 2015).

Results It can be observed that domain names change their traffic patterns as soon as they are used for malicious purposes (see Figure 4.13). In the month before the botnet domains become most active 78 % experience an increase in queries of more than 10 %. Also, 81,4 % of the malicious domains have experienced a rapid increase in queries during this time. Furthermore, we showed how non-malicious events, like leaving the quarantine, can cause significant changes in query volume as well.

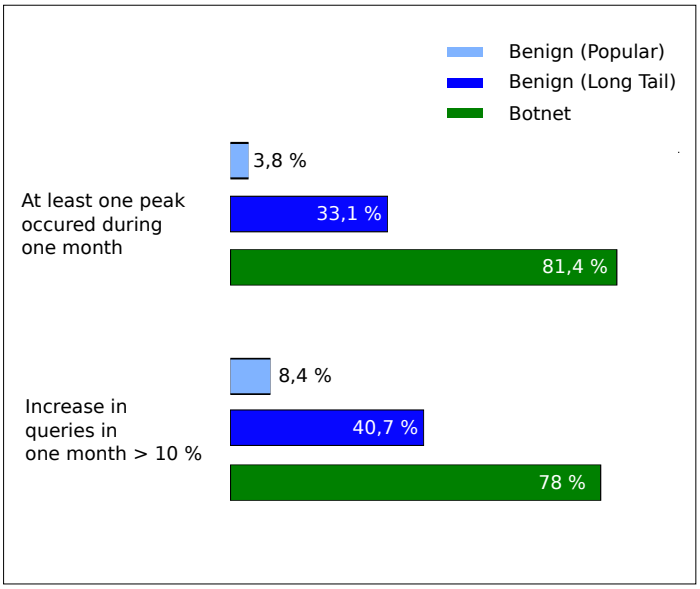

Figure 4.13: Comparison between benign and botnet domains based on query peaks and query growth.

4.3.4 Resolver Lookup Similarity

For the similarity of resolver lookups, newly registered domains are examined 2 days before and 4 days after registration. For each known botnet domain, the day with the highest number of queries is selected. Then, the daily resolver similarity is calculated 2 days before and 4 days after the peak occurred. Phishing domains that have been hijacked are analysed 2 days before they were reported by *Netcraft* and 4 days after. For newly registered phishing domains the registration date is selected instead of the reporting date.

Table 4.1 shows the average and median Jaccard Similarity for each set of domains. Furthermore, the average standard deviation indicates how similar the Jaccard Similarity is between the domains in a set. The last column shows the average difference between the highest Jaccard Similarity and the lowest Jaccard Similarity. A high value indicates that, on some

	Mean	Median	Std.	Avg. Dif. Min/Max
Popular Domains	0,238	0,282	0,154	0,046
Unpopular Domains	0,23	0,219	0,129	0,207
New Domains	0,133	0,088	0,167	0,256
New Domains Phish	0,245	0,327	0,135	0,478
Phishing Domains	0,243	0,232	0,106	0,311
Botnet Domains	0,282	0,268	0,189	0,184

Table 4.1: Jaccard Similarity of Querying Resolvers

consecutive days, same resolvers query for the domain name but on other days very different resolvers send queries.

Figure 4.14 shows the variability of the Jaccard Similarity of popular, unpopular, botnet and phishing domains. This variability is depicted by the cumulative distribution of the variance of each domain. The steeper the curve, the more often are domains queried from a constant set of resolvers. This result coincides with the results by Hao et al. (2010). Popular benign domains are queried from a stable set of resolvers most often, followed by benign unpopular domains. The query sources of malicious domains are more inconstant. If a botnet is not growing further, domains for C&C receive queries of a stable set of infected clients. This can explain the difference between phishing domains and botnet domains.

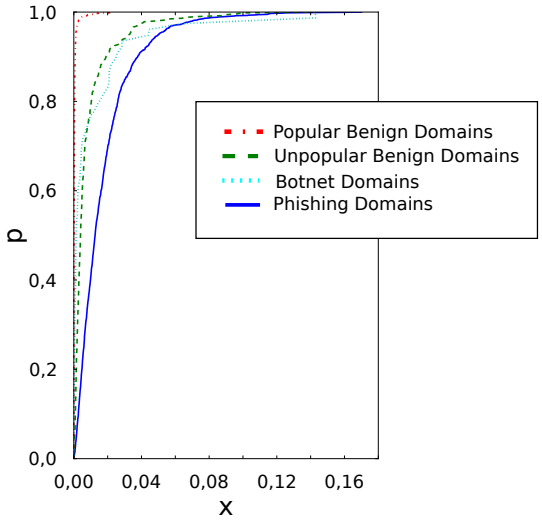

Figure 4.14: Cumulative distribution of the variance of the Jaccard Similarity for benign, botnet and phishing domains.

This confirms results from previous research in the gTLDs *.com* and *.net*. Benign domain names are queries from a more consistent set of resolvers than malign domain names.

4.3.5 Domain Name Server Characteristics

We have observed the name server changes over the period of one month. We have selected domains that changed their name servers every 24 hours or less. Although we were able to observe domain names that had a new name server entry every 60 minutes, a closer examination revealed that these domains belonged to the same name servers of a web hosting company which shared 6 different IP addresses and rotated through them over time. Other domains changed name servers only for one minute which might indicate a mis-configuration.

These observations lead us to the conclusion that double flux networks are not prevalent in *.nl*. However, this might change again in the future and should be observed further.

4.3.6 Subdomain Characteristics

The most used subdomains in *.nl* include *www., ns1, ns2..., mail., dmarc.* and *smtp.* Also, many subdomains include strings like *ldap* or *kerberos* that indicate the usage of special access and authentication protocols on a website.

We inspected domains that received queries for significant more unique subdomains from one day to another. There, often the subdomains included strings like *dmarc* or *ldap*, but occasionally subdomains like *opypgdl.*, *lxw-wobifchgd.* or *nfufqfxvg.* appeared. These look similar to domains that have been observed during domain shadowing attacks and have been queried on multiple days.

We have implemented a script to automatically observe domain names which have been queried for suspicious looking subdomains before. The script continuously keeps track of every incoming query for this domain. As soon as a new subdomain is queried, the script automatically resolves the subdomain in order to check if the subdomain has a valid IP address and whether it is different from the IP address of the 2nd level domain. During an observation of one week, none of the observed subdomains were successfully resolved.

This leads us to the conclusion that the observed domains were not part of a domain shadowing campaign. However, observing an increase in unique subdomains can be a good first step on the way to detect these attack methods.

4.3.7 Domain Registration Characteristics

We have examined the registration information of 65 newly registered phishing domains and 220 new benign domains. We have analysed the names of the registrants, the country of their addresses, their phone numbers, email address and registrars at which the domain has been bought. As it can be

seen in Figure 4.15, benign and phishing domains were mostly registered with information from the Netherlands. The country codes of the phone numbers correspond with the country of the address. 2 phone numbers have been used to register more than one phishing domain.

Phishing domains were more than twice as often registered with free-email addresses like *outlook.com*, *hotmail.com*, *mail.com* and *gmail.com* as benign domains and one phishing domain has been registered at *sharklasers.com* that offers disposable email addresses. 1 address was used for more than one phishing domain registration.

Registrations for benign domains are spread over a large variety of registrars whereas 48 % of the phishing domains are registered at the same Dutch registrar. Only 9 % of the phishing domains have been registered at a registrar outside of the Netherlands. Even supposedly benign domain have been registered with obvious fake phone numbers like *+31.0123456789* (3 %).

The fact that many phishing domains are registered at one specific registrar sticks out and is a distinctive feature of malicious domains. Also, using a private email address might indicate that the domain name is registered for benign purposes. By using Dutch names and addresses for registration, phishers blend into the expected patterns of benign registrant of *.nl* domain names.

4.4 Remarks and Summary of Findings

We have shown that malicious *.nl* domain names have different characteristics than benign domain names. A sudden increase of queries and abnormal geographic query patterns are a common feature of phishing and botnet domains and these observations coincide with findings in previous research. A novel characteristic has been observed for domains that have been released from quarantine. They receive a higher number of queries than other newly registered domains which can be useful to distinguish new benign domains from new phishing and botnet domains.

The geographic attributes of benign *.nl* domain names have the unique characteristic that they get the majority of queries from the Netherlands and the US. Therefore, the geographical origin of DNS queries is a stronger distinctive feature in *.nl* than for generic TLDs like *.com* or *.net*. Further, we have demonstrated that newly registered phishing domains mostly have registration information that are similar to benign domains. Phishers use often Dutch names and addresses which shows that they adapt their behaviour to the misused TLD.

In addition, we were able to rule out certain malicious activities that do not need be taken into account in *SIDekICk* for now. First, double flux botnets in *.nl* can be neglected with a high certainty and second, no

shadowing attacks have been observed so far.

These findings will be used in the next chapter to select adequate features to build an effective classifier for malicious domains.

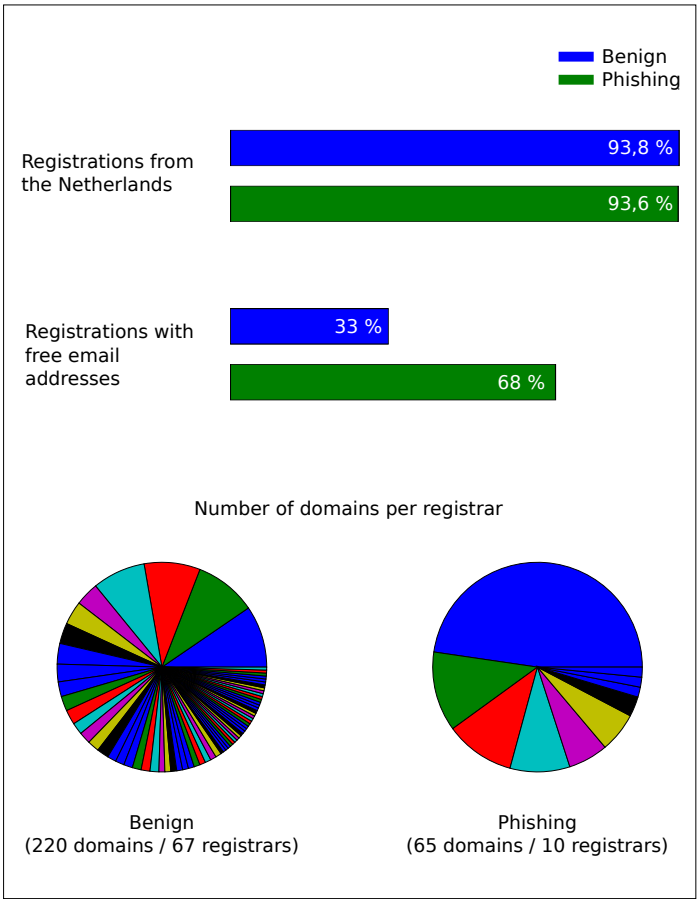

Figure 4.15: Characteristics of domain registration information of benign and phishing domains.

Chapter 5

Detecting Malicious Domains in *.nl* with *SIDekICk*

Detecting malicious domains from the perspective of a TLD operator was so far mainly conducted by researchers that focused on the TLDs *.com* and *.net*. Hao et al. (2010) measured at AuthNSs how DNS lookup patterns for malign domains differ from benign domains. A year later, the same authors narrowed down their research to detect malicious domains that got newly registered and included the ccTLD *.ca* as well (Hao et al., 2011).

The TLDs *.com* and *.net* differ a lot from the ccTLD *.nl*. In 2014, 115,6 million *.com* and 15 million *.net* domain names were registered (Verisign, 2015) compared to 5,5 million registered *.nl* domain names. This also affects the number of domains which are used for malicious purposes. Aaron and Rasmussen (2015) counted over 60.000 domains used in phishing campaigns in *.com* and *.net* in the second half of 2014 whereas only around 400 where reported for *.nl*. Furthermore, domains in *.nl* mainly focus on a Dutch audience, whereas *.com* and *.net* are generic TLDs with visitors from all over the world. Therefore, it might not be possible to apply the same detection methods to *.nl* as for the other TLDs.

5.1 Goals and Challenges

In the previous chapter, we have shown which characteristics are typical for malicious domains and how they differ from benign domain names. Phishing domains are a more common phenomenon than botnet domains in *.nl*. This allows us to make more general assumptions about the characteristics of phishing domains than botnet and other malicious domain names. The small number of known botnet domains would make it hard to build a solid data set that can be used as a ground truth in order to detect botnet domains automatically and makes the validation of botnet domain detection algorithm difficult. Therefore, the goal of *SIDekICk* is mainly to detect

phishing domains on a daily basis. There, we have a comparably large number of known phishing domains and also, new phishing domains are reported on daily basis.

Phishing domains are either registered especially for the campaign or are compromised web servers reachable with a previously benign domain name. Therefore, *SIDekICk* should be separated into two main components. The first has the goal to detect newly registered domains, the second to detect domains that got recently compromised and are now used for the purpose of phishing. As output, both components list the domains that have been classified as phishing. On this basis, we can verify whether a tool can be built that distinguishes benign and malign domains from the view of a ccTLD. Because phishing campaigns can share characteristics with botnet domains and domains of exploit kits, it can be expected that some of those domain names can be detected as well.

Newly registered phishing domains have rather distinct query patterns and are often queried from unusual countries. Also, the number of daily registered domains is only around 2.500 domains which limits the set of domains that need to be analysed. In comparison, in May 2015 the name server of SIDN received queries for over 7,5 million unique, existing and non-existing domains per day. Also, the ENTRADA platform collects only data from one name server for now. SIDN has deployed 4 unicast name servers and a set of anycast servers. As a consequence, only around 15 % of the total number of queries can be analysed, which has an effect on the origin of DNS queries as well. Our data set is further limited by the fact that query data is only accessible until May 2014. In case a domain was used for malicious purposes before that date, no observations of DNS query before and after the infection can be made.

5.2 *SIDekICk* Overview

In order to identify phishing domains, we divide *SIDekICk* into two components. The first component has the goal to identify domain names that have been recently registered to be part in a phishing campaign and is further referred to *SIDekICk-New*. The second component analyses the continuous query stream for every domain and tries to identify patterns that indicate that a domain has been recently compromised and is now used for a phishing campaign. It is referred to as *SIDekICk-Comp*. Besides the different goals, the schema of reading data, processing data and generating an output is similar (see Figure 5.1).

SIDekICk works in epochs of one day. At the end of each day, both components collect the domains that need to be analysed. *SIDekICk-Comp* selects every domain from the Hadoop database that got queried at least 50 times and *SIDekICk-New* first selects the domains that have been registered

on the selected day (Step 1). Then, for each domain the features, which are needed to determine whether a domain is used for malicious activities or not, are collected (Step 2). Both components apply individual filters to exclude domains that have a low chance to be malicious (Step 3). Then, the remaining domains are fed into the classifier modules (Step 4). *SIDekICk-New* and *SIDekICk-Comp* rely on different underlying classifications models. In the last step, the domains that have been classified are reported (Step 5). *SIDekICk-New* additionally reports domain registration information that should allow a human to detect false positives easier.

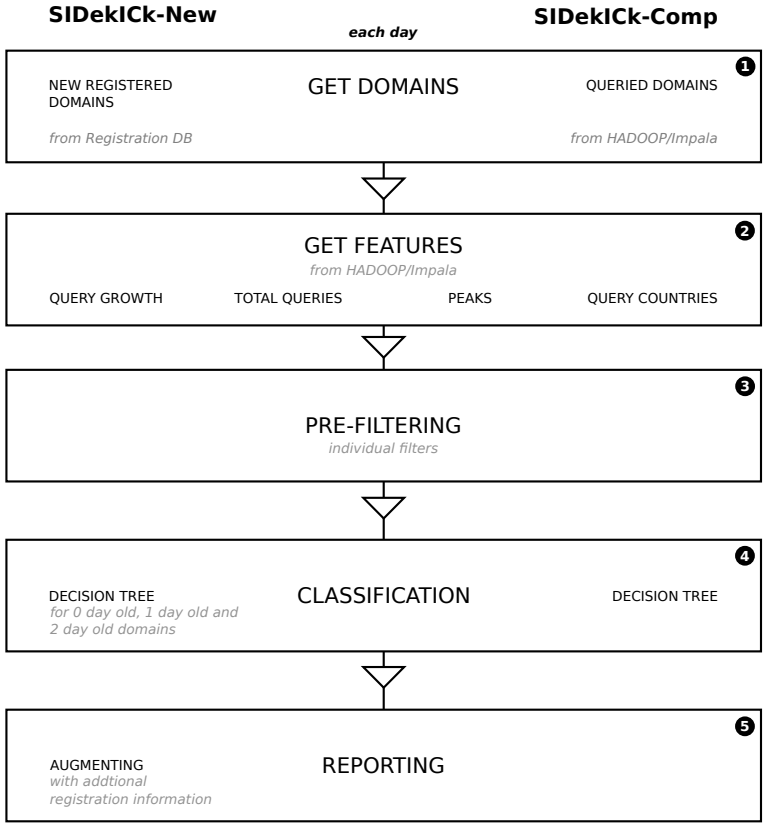

Figure 5.1: Schematic structure of SIDekICk

SIDekICk-Comp only verifies the domains that have been queried on a single day. *SIDekICk-New* follows an iterative approach for every newly registered domain (see Figure 5.2). At the end of an epoch, it collects the domains that have been registered on the current day, the day before and two days before. Domains are not necessarily used in a phishing campaign the same day they get registered. Therefore, *SIDekICk-New* observes newly registered days consecutively on the day of the registration and the following two days after. Because of different characteristics, different classifiers are used for domains that are one or two days old (Step 3).

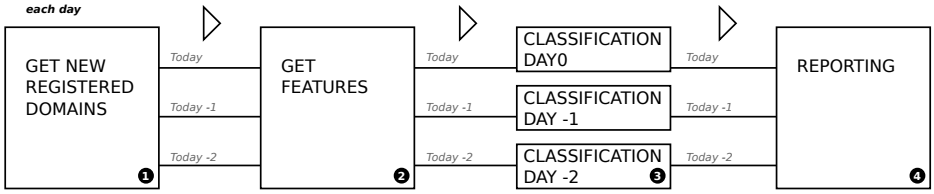

Figure 5.2: Schematic structure of the *SIDekICk* module to detect newly registered domains.

5.3 *SIDekICk* System Implementation

As part of this thesis, a prototype of *SIDekICk* is developed. *SIDekICk* is implemented in *Python 2.7.8* using the underlying machine learning library *scikit-learn 0.15.2*[1]. Data is accessed from different sources. Most of the data related to DNS queries is stored in *Hadoop* clusters and is accessed through the *Cloudera Impala*[2] parallel processing SQL query engine. *Impyla*[3] is used to query data directly from python. Domain registration data is stored in the internal Oracle SQL Database. It includes for example information about registrants, registrars and name servers and is queried with the Python extension module *cx_Oracle*[4]. The queried data is stored into labelled data structures provided by the *Pandas*[5] library that allows to manipulate and analyse large data sets easily.

The following sections first describe which features for *SIDekICk* are selected, which data sets are used to train the classifiers and how they are calculated, followed by a description of the parameters of the classifiers. Finally we describe briefly how domains, which have been classified as malicious, are augmented with additionally information that should support a user to reduce the false positive rate. In case the implementation differs, we first describe the details of *SIDekICk-New* and then of *SIDekICk-Comp*.

5.3.1 Feature Selection

Features describe characteristics of domains with which a classifier can identify whether a domain is used for phishing or not. In Chapter 4 we have described how the characteristics of phishing domains differ from the characteristics of benign domains. Based on this domain knowledge, four features have been selected. This section describes the selected features in more detail and explains how the features are calculated such that they can be processed by machine learning algorithms.

[1] `http://scikit-learn.org/stable/`
[2] `https://github.com/cloudera/impala`
[3] `https://github.com/cloudera/impyla`
[4] `http://cx-oracle.sourceforge.net/`
[5] `https://github.com/pydata/pandas`

Selected Features

For classification, *SIDekICk* stores each domain that needs to be analysed into a Pandas data frame, which is a table-like structure and is kept in memory. Each domain is represented by a row, using the domain name as an index. Features are stored in separate columns.

Geographic Deviation The location of a resolver that queries a domain name has shown to be a good indicator whether a domain name is used for malicious purposes or not. The goal of this feature is to compare the origin of queries of a suspicious domain with the geographic origin we would usually expect for *.nl* domain names. Calculating the geographic deviation of a domain name from the expected geographic distribution of DNS queries includes the following 5 steps:

1. Based on the Alexa 1.000 and Webhoster-Domains, we calculate how many queries in percent for each country and domain name have been observed in April 2015. For example, in April 2015 *ns.nl* received 35,8 % of the queries from the US, 19,8 % of the queries from the Netherlands and 5,3 % of the queries from Germany.

2. Then, we calculate for each country the mean share μ and standard deviation σ.

3. For each new domain that is going to get classified, the observed geographical origin of the queries is collected for the day of classification and the share γ for each country is calculated.

4. Next, the share of each country γ is compared with the expected share μ. An unexpected high deviation is observed when $\gamma > \mu + 3 * \sigma$ or $\gamma < \mu - 3 * \sigma$.

5. The number of countries that have an unexpected high deviation divided by the total number of observed countries defines the numeric value of the geographic deviation. It can be between 0 and 1 where 0 means no deviation from the expected countries.

Steps 1 and 2 are calculated once, steps 3 to 5 are repeated for every domain name that needs to be classified. Because especially unpopular domains in *.nl* often get queries only from the Netherlands and the US, higher shares of these countries are not taken into account.

Query Count The query count is the most basic feature. It reflects the number of queries that have been received from the SIDN name server for a domain name within an epoch and is an integer value. The epoch in *SIDekICk* for this feature is one day.

Query Peak A query peak (or rather rapid query growth) can occur, when a domain is first used for malicious purpose. Because phishing campaigns send mails that contain a link to a domain to thousands of recipients within a short time period, a rapid increase in queries is expected. Query peak is boolean feature.

1. For each domain, the number of queries on the day of the classification and the day before are fetched.
2. Then, it is calculated if the number of queries of the current day are two times higher than the day before.
3. If this is the case, the feature query peak is set to 1 else to 0.

Query Growth The query growth has the goal to measure, whether a domain name is experiencing an increase in queries in the last three weeks. It is represented by a positive floating point number.

1. For each domain, the number of daily queries of the last three weeks is fetched.
2. Then, the average number of queries for the first 7 days and the last 7 days of this time period is calculated. By considering 7 days we expect to take weekly query variations into account.
3. Finally, the average number of queries of the last 7 days is divided by the average number of queries of the first 7 days. The result shows, if there has been an increase or decrease in queries over the last three weeks.

This number can be used as an confirmation whether a measured query peak was actually an unexpected event. For example, some benign domains might receive a rapid increase every third day. These domains would not have a significant query growth. In comparison, a malicious domain that gets newly infected is expected to receive a query peak and also a significant increase in total queries compared to three weeks earlier.

Discarded Features

Aside the selected features, we do not consider the domain relationship between malicious domains because the main focus of *SIDekICk* lays on phishing campaigns. Phishing campaigns mostly do not rely on a distributed architecture as some exploit kits and botnets do and therefore, no benefits for detecting unknown phishing domains are expected.

Furthermore, the resolver similarity is not taken into account as well. Calculating the resolver similarity for several thousand domains is time consuming. The *SIDekICk* prototype is running on a virtual machine with four

2 GHz cores and 16 GB RAM and analysing the resolver similarity of 300.000 domain names for a period of seven days would take around 8 hours. Also, Hao et al. (2010) have identified a correlation between strong variations in querying resolvers and a rapid increase in queries from one day to another. Because the Query Peak feature already covers this characteristic, analysing the resolver similarity would be redundant.

Domain registration information are not taken into account during classification but are used in *SIDekICk-New* to support a user in verifying the results of the classifier. Also, the quarantine-release-date for a domain is not part of the feature set but is used as a filter that is applied after classification to reduce the number of false positives.

5.3.2 Selecting Domains for Training and Verification

Training and verification data is necessary to provide a ground truth for the classifiers and to measure the performance of the models. This data must include domains that are known to be malicious and domains that are known to serve a benign purpose. Each domain in the training set is labelled as phishing or benign. Because the focus of the two *SIDekICk* modules differs, different training sets must be used.

SIDekICk-New 70,5 % of the newly registered phishing domains are active at the same day of registration or one or two days after. Therefore, we have decided to build a classifier that is mainly trained to detect domains maximum two days after registration. Because the number of queries increases with each day, we have decided to train a separate classifier for day 0, day 1 and 2 after registration. Thus, the phishing domains are split into three sets, depending on which day they have been active. Some domains have already been discovered by *Netcraft* before they have received significant queries. This might have been possible when phishing domain names contained brand names or words that have been used in other phishing campaigns before. For this reason, we have selected phishing domains that received at least 20 queries. Thereby we expected to include only domains in our training set from which meaningful features can be extracted. The final training set for phishing domains contains on registration day 36 domains and for the two following days 40 domains. For each of the domains in the sets, features are collected starting on the day the domain got reported by *Netcraft*.

The training set for benign domains contains the domains that have been registered on 2015-04-07 and 2015-04-08, received more than 19 queries two months later, did not show up on blacklists and were furthermore manually validated. It contains 222 domains. For each of the domains, features have been collected starting on the day of registration and the two following days.

SIDekICk-Comp The second module focuses on long lived domains that have been compromised and misused for phishing. The training set of compromised phishing domains contains over 1.800 domain names.

Because the majority of domain names in *.nl* receive only a small number of queries we do not use the Alexa 1.000 domains and hosting domains as training data but rather use them as a filter in the second step of the classification process. Instead, the set of benign domains contains a sample of old domains that have received at least 20 queries on the start and end-date of a period of two months. These domains are not part of the Alexa and hosting set and do not appear on blacklists. Additionally, the newly registered domains of *SIDekICk-New* are used as well. In total, 510 benign domains of the long tail are used as training data. Features of those domains are collected starting from 2015-04-30.

5.3.3 Building the Classifier

The classifier is the core module of *SIDekICk*. It relies on an algorithm that uses the selected features and labled training data to build a model with which previously unknown domain names can be classified. Both components of *SIDekICk* rely on Decision Trees. Decision Trees have proven to be effective in previous research (see Section 3.3), do not rely on normalised features and their results can be interpreted easily in form of a plotted tree. We compare the results of the Decision Tree with the results of a Support Vector Machine (SVM). SVMs have shown to achieve a good separation of data points and have performed well with small training sets like ours (Martinez-Bea et al., 2013; Davuth and Sung-Ryul, 2013). Both algorithms are already implemented in the *scikit-learn* library. The Decision Tree is based on the CART (Classification and Regression Trees)[6] implementation and the SVM uses the implementation by Guyon et al. (1993).

Both algorithms first have to be trained. Therefore, they provide a function that is expecting a two-dimensional array of training data and an array of labels as input and returns a model with which unknown domains can be classified. In the two-dimensional array, each row is a domain name and each column is a feature. The second array is a list of labels that assigns each domain either a 0, if the domain is benign, or a 1, if the domain is a phishing domain. A classifier function uses the created model and a one-dimensional array of the features of one domain name as an input and returns either a 0 or a 1 depending on the classification.

Both algorithms have different parameters to influence the classification of domain names. We have followed the recommendations of the scikit-learn documentation to improve the algorithms in order to achieve the most optimal results (Scikit-learn, 2015a,b).

[6] `http://scikit-learn.org/stable/modules/tree.html#tree-algorithms`

In the following two sections, we first explain models for detecting newly registered domains followed by the models for detecting compromised domains.

SIDekICk-New

The set of known new benign domains is separated into a set of 166 training and 55 verification domains (25 % verification data). The set of known new phishing domains is split into 24 training and 12 verification domains for the registration day and 27 training and 13 verification domains for the first and second day after registration (33,3 % verification data). The performance of the classifiers is measured with the help of the false positive and true positive rate. The false positive rate defines the share of domains that erroneously have been classified as malicious, the true positive rate defines the share of domains that have been correctly classified as malicious.

During the training phase, we have weighted the samples such that there was a five times higher chance that a domain is classified as a phish. Also, the decision tree algorithm did not create new child nodes if less than five domains of the training set would have ended up in this node [7]. Thereby, over-fitting was reduced and the best performance has been achieved (see Table 5.1).

The outcomes of the training phase are two very simple Decision Trees (see Figure 5.3) where the Decision Tree for the first and second day of the registration works best for both days. The tree is read from the top to the bottom. For each classified domain, first the top condition is tested. If the condition is fulfilled, the domain moves to the node of the left branch, if not, then the domain moves to the right branch. The domain is classified as soon it reaches a leaf node.

It can be seen that for the day of the registration, only the query growth and the geographic deviation is considered. For the first and second day, only the number of queries and the occasion of a query peak play a significant role. The training function selects the most adequate features on its own.

We have built a classifier based on Support Vector Machines to assess the results of the Decision Tree. Due to our unbalanced training set, we automatically balanced the training data before training the classifier[8].

It can be seen in Table 5.1 that Decision Trees perform slightly better than the SVM. Therefore, we decided to use the Decision Tree classifier in the final prototype of *SIDekICk*-New. In Chapter 5.4 the classifier is then evaluated over a period of one month.

[7] class *DecisionTreeClassifier* constructor parameter $min\_samples\_leaf = 5$

[8] class *SVC* constructor parameter $class\_weight = auto$

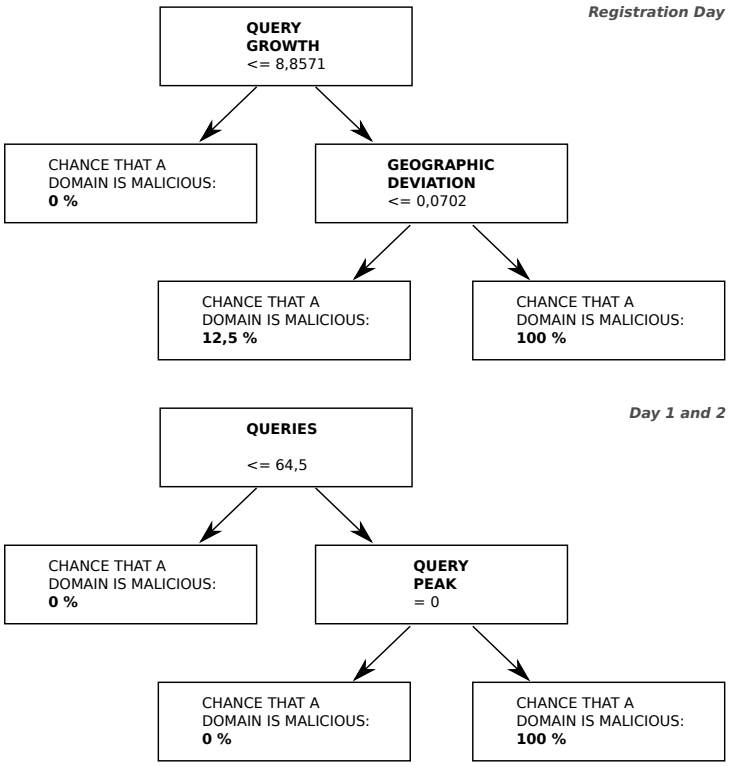

Figure 5.3: Newly registered domains - Decision Trees

SIDekICk-Comp

In order to reduce the false-positive rate, the training sets for the Decision Tree and the SVM are pre-selected. Both rely on the same set of benign domain names, split 3 : 1 into training and verification set (382 domains to 127 domains). The Decision Tree only uses known malicious domains for training that experienced a growth bigger than 2, a geographic deviation bigger than $0,2$ and had a peak on the query date (67 domains for training and 33 domains for verification). The SVM is using a training set of malicious domains that had a query growth bigger than 1 and a geographic deviation bigger than $0,1$ (294 domains training to 147 domains verification).

The benign domains of the training set of the Decision Tree are weighted 5 times more than malign domain names and the minimum samples in a leave is set to 5. Using these parameters, a tree with the depth of 3 is created (see Figure 5.4). The tree takes the query growth, the geographic deviation and the occurrence of a peak into account.

The SVM weighs benign domains in the training set twice as much as malicious domains. Thereby the number of false positives can be reduced. Table 5.2 shows the performance of both classifiers. The Decision Tree

	False Positives		True Positives	
Algorithm	Tree	SVM	Tree	SVM
Day 0	1,75 %	3.6 %	90,0 %	83,3 %
Day 1	1,75 %	1,8 %	90,9 %	84,6 %
Day 2	1,72 %	7,2 %	90,0 %	84,6 %

Table 5.1: Newly registered domains - Classification evaluation

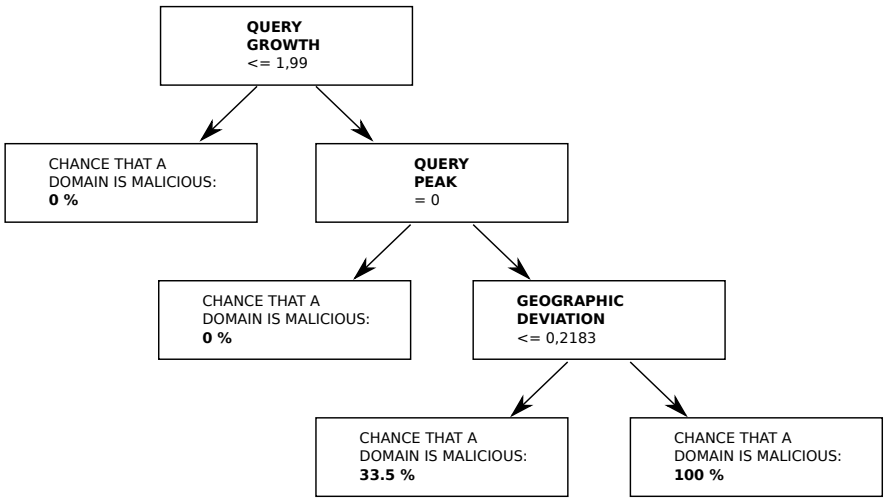

Figure 5.4: Old domains - Decision Tree

outperforms the SVM in the number of false positives and true positives and is therefore the choice for the final prototype.

	False Positives		True Positives	
Algorithm	Tree	SVM	Tree	SVM
Classification	3,1 %	15,0 %	90,1 %	70,5 %

Table 5.2: Old domains - Classification evaluation

5.3.4 Reporting the Results

In the last step of the *SIDekICk* classification process, the domains that have been classified as phishing are reported. So far, no user interface is provided by *SIDekICk* and only the classified domain names are listed by the *python* script. For each domain that has been classified as malicious by *SIDekICk-New* additional registration information are displayed. These include the name, address, phone number and email address of the registrant, as well

as the used registrar and whether the domain is reachable at the moment. These information have shown to be helpful during evaluation to identify false positives. For example, domains that have been registered with the name of a popular domainer are very likely false positives.

5.4 Evaluation

In this section, we evaluate the modules *SIDekICk-New* and *SIDekICk-Comp* over a longer period using formerly unknown and unclassified domains. *SIDekICk-New* performs very well. Every domain that was later reported by *Netcraft* got detected and the false positive rate is 0,3 %.

The performance of the *SIDekICk-Comp* module is harder to evaluate. Within one week, we detected over 14.000 domains that might be suspicious but we miss efficient tools to evaluate them. Domains that were reported by *Netcraft* were classified as malicious only in few cases and often the total number of supposedly malicious domain was too high to be realistic.

5.4.1 SIDekICk-New

Process *SIDekICk-New* has been evaluated over a period of 31 days from 2015-06-08 until 2015-07-08. In this period, 61.100 domain names were registered in *.nl*, with a daily average of registrations of 1.970,97 domain names. In the same time, *Netcraft* reported 10 domain names as phishs that have been misused within 2 days after the registration.

At each day, we first fetched every domain name that has been registered on the same day, the day before and two days before and stored them in a temporary list. Then, for each list we collected the features of the domain name on the day of the classification. Collecting the features for the domain names takes less than three minutes. Depending on the age of the domain name, we applied the different classifiers to identify newly registered phishing domains. We describe this process with an example for the date 2015-06-14:

1. Collect domains that have been registered on 2015-06-14, 2015-06-13 and 2015-06-12.
2. Get features for each of the domains on 2015-06-14.
3. Apply the decision tree for the registration day on domains from 2015-06-14 and the decision tree for one and two day old domains on 2015-06-13 and 2015-06-12.

Often, we have seen that domain names which had less than 20 queries were not malicious even though they have been classified as such. Therefore, we apply a filter before step 3, where every domain name that has less than 20 queries is neglected.

	False Positive Rate		
	Standard	Optimised	Improvement
Day 0	0,055 %	0,048 %	12,7 %
Day 1	0,181 %	0,152 %	16,0 %
Day 2	0,114 %	0,104 %	8,8 %
Average	0,117 %	0,101 %	12,5 %

Table 5.3: SIDekICk-New - False positive rate for the standard and optimised classifier

Results In the evaluation period, every domain reported by *Netcraft* has been detected. 9 domains have been detected on the day of the report and one domain has been detected one day earlier. Additionally, 13 phishing domains have been detected that have not been found by *Netcraft*. Besides phishing domains, the classifier also found domains that, were used for bogus webshops in order to sell fake sport shoes or domains forwarded to scamming websites were paid surveys and other online scams were promoted. In total, 33 malicious domains were reported with an average of 1,06 domains per day and 0,35 domains per classification. The fact that we found twice as many phishing domains than reported by *Netcraft* indicates that there might be even more domains left undetected. Especially phishing campaigns that target a very specific and small group of users might not be detected by *SIDekICk-New* because of their low number of queries. However, 75 % of the global phishing campaigns are targeting one of 10 very popular Internet-services like *PayPal*. These phishing campaigns reach many users and therefore are most likely detected by our classifier. Thus, we assume that even in the worst case scenario the false negative rate is below 25 %.

On the other side, 235 domain names were detected that were very likely false positives. For every classification of a domain set, 2,35 domain names were falsely detected as malicious. The total false positive rate is 0,34 % (235 of 69.067 domain names). The false positive rate differs among the days of classification (see Table 5.3). The decision tree for the day of the registration has the lowest number of false positives, the tree for the first day the highest number.

Optimisation During the evaluation, we observed that some domains that erroneously have been classified as malicious left quarantine recently. Therefore, we assume that these domains were registered by domainers which caused an increase in queries. We introduced another filter after step 3 and excluded every domain that has left quarantine in the last two days. Thereby, only 200 domains were mistakenly classified as phishs and we were able to reduce the total FP rate by 14,9 % to now 0,29 % (200 of 69.067 domain names).

5.4.2 SIDekICk-Comp

Process The second modules is evaluated for a period of 7 days from 2015-06-08 to 2015-06-14. In this time, *Netcraft* reported 96 phishing domain names that have been older than 7 days.

At the end of a day, we collect the features for every domain name that received more than 50 queries. This takes around 60 minutes for 200.000 domain names. After the features are collected, the classifier is iterating through every domain name and returns the result of the classification. This takes around 3 minutes for 200.000 domain names (0,9 ms per domain name).

Results Evaluating the results of *SIDekICk-Comp* is difficult. In total 14.242 unique domains have been classified as malicious. However, when comparing the domain names that have been classified as malicious with the domains that have been reported by *Netcraft*, only 10 of 96 have been reported correctly (10,4 %). *Netcraft* only focuses on phishing domains but because also compromised domains used by exploit kits share similar characteristics, we assume that among the classified domain names also domain names of these kind are listed. In order to get a better understanding of the set of supposedly malicious domains, we analysed the most queried domains with *virustotal*. As a result 50 of 14.242 domains (0,35 %) were reported by at least one of the scanners used by *virustotal*. From their experience with false positives, Čermák et al. (2014) defined that at least four scanners of *virustotal* must detect a domain name before they consider a domain as malicious. We considered this threshold to achieve a mores rigorous evaluation such that only 8 of 14.242 (0,06 %) are certainly malicious domain names. We had to focus on the 500 most queried domains because of rate limitations of the *virustotal* API.

An overview of the results can be seen in Table 5.4. It shows the number of analysed domain names, the number of domains that have been reported by *Netcraft* that day, the number of domains that have been classified as benign and as malicious. Also, it is listed how many of the reported domains by *Netcraft* appeared among the domains that were classified as phishing. The last column shows the number of domains that have been classified by *virustotal* as malicious as well.

Even if we consider a high number of domain names that are infected but are not reported, the classifier still has reported very likely many false positives. We looked at some of the reported domains manually and have seen for example websites of a local Dutch shop for pedicure with over 200 queries per day and many queries from Peru and Mexico. This behaviour looks very suspicious but is not enough to determine whether the domain is actually malicious or not. To validate this assumption it would be necessary to find for example the hidden URL that hosts the malicious code. This cannot be observed from the vantage point of an TLD registry.

Date	Analysed	Rep.	Benign	Mal. (reported)	*virustotal* (>3)
06-08	332.121	16	331.630	491 (2)	10 (2)
06-09	243.930	20	243.875	55 (0)	0
06-10	247.212	11	236.530	10.682 (0)	6 (1)
06-11	261.193	10	259.797	1.396 (3)	10 (2)
06-12	218.684	18	217.566	1.118 (0)	7 (2)
06-13	207.427	15	207.126	301 (1)	8 (0)
06-14	207.821	16	207.622	199 (4)	9 (1)
Total	1.718.388	96	1.704.143	14.242 (10)	50 (8)

Table 5.4: SIDekICk-Comp - Detection of potential malicious domains

In total, 41 unique domains have been listed by *virustotal*. Figure 5.5 shows the shares of the most common countries for domains that have been classified as malicious by *SIDekICk-Comp* and by *virustotal*. It can be seen, that both sets of domains have a geographic distribution that varies from what we observe for benign domains from our training set. Most of the queries from the first set come from Spain (13,95 %) and most of the queries from the second set come from China (18,7 %). Both sets share a high query growth and a medium number of queries (see Figure 5.6). The number of queries varies stronger for domains that have been classified by *SIDekICk-Comp* than for domains classified by *virustotal*.

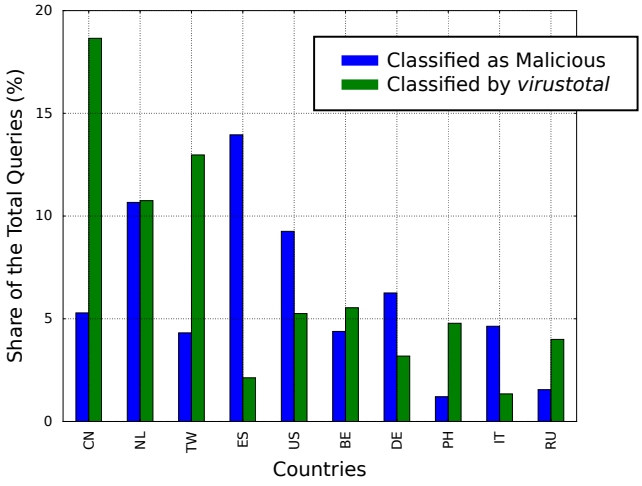

Figure 5.5: *SIDekICk-Comp* - Shares of the geographic location of querying resolvers for domains that have been classified by *SIDekICk-Comp* and domains that have been detected by *virustotal*.

The similarity between the domains classified by *virustotal* and the domains classified by *SIDekICk-Comp* shows that we have selected the right features to detect old, compromised domains but we still miss features to

narrow down the results and to decrease the high number of false positives.

In the next chapter we summarise our findings in this thesis and discus what would be necessary in order to make *SIDekICk* part of an initiative to actively fight malicious domain names in *.nl* and other zones.

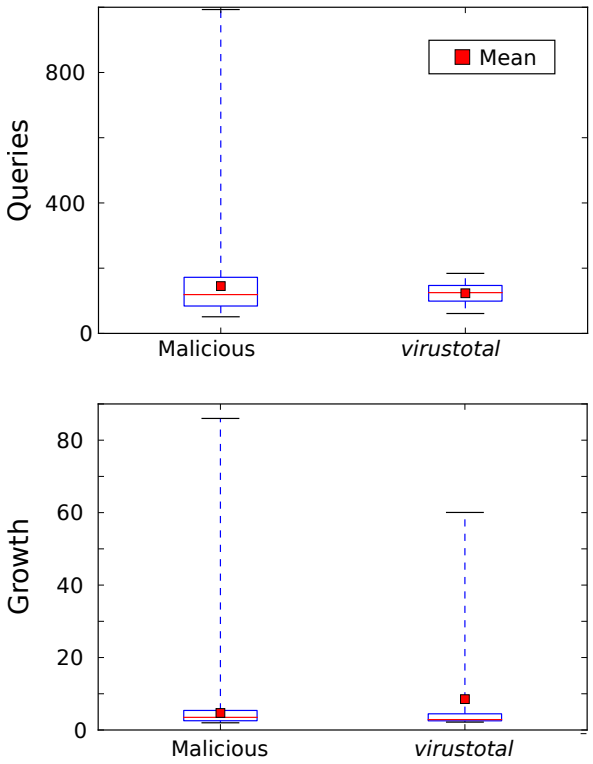

Figure 5.6: *SIDekICk-Comp* - Attributes of the domains classified as malicious and domains classified by *virustotal* as boxplots (without outliers).

Chapter 6

Conclusion

In this thesis we focused on malicious domain names in the *.nl* ccTLD. We provided a characterisation of known malicious *.nl* domain names based on previous research and confirmed that behaviour of malicious domains in gTLDs like *.com* and *.net* can be observed for *.nl* as well. Beyond this, we explained how domains that are leaving quarantine can have characteristics that can be similar to newly registered malicious domains and how the observation of small resolvers can be used to detect previously unknown botnet domains. The characteristics of quarantined domains were used in combination with features of DNS query frequency and geographic location of querying resolvers to develop a prototype called *SIDekICk*, which is able to detect newly registered malicious domains with a high precision and a low false positive rate. A second module of *SIDekICk* lists domains that are potentially compromised based on the daily analysis of every *.nl* domain name that has been resolved. We have shown that, although the classifier uses features that are characteristic for malicious domains, more features are needed in order to improve its precision.

6.1 Limitations

The developed prototype still has a few limitations and is relying on certain variables that have an impact on its performance.

First, although the false positive rate of *SIDekICk-New* is below one percent, on average 6 domain names are still falsely classified as malicious per day. If the phish is hosted on the main site, then these domains can be verified manually. However, if the phish is hidden in a sub-directory, manual inspection becomes more difficult. In *SIDekICk-Comp*, the false positive rate is high and in our training period over 2.000 domains were classified as malicious per day on average. This makes a manual classification not efficient such that further improvements of the classifier are necessary.

The precision of the classifiers relies on the accuracy of the features that

are collected for each domain name. If the geographic location of multiple IP addresses of large resolvers are not determined correctly, then *SIDekICk* might classify some domains erroneously as malicious.

The limited data set from only one name server might have an impact on the precision of the classifier as well. Assuming that a large ISP sends usually queries to one of the name servers which are not monitored by ENTRADA, but suddenly sends most of the DNS queries to the server whose queries are monitored, then domains will receive rapidly more queries, and therefore have high growth and high peak. This again might cause a wrong classification.

Culprits can avoid detection if they mainly target Dutch users, such that the geographic deviation stays low. If they would be able to avoid a rapid increase of queries we probably would not detect their domains as well.

6.2 Future Work

In order to cope with the previously mentioned limitations and to verify the results, several improvements can be applied in the future.

One way to improve precision of the classifiers could be to analyse the web-server software and content management system (CMS) of a suspicious domain. Wisniewski (2015) has discovered in his research that many compromised domains run old vulnerable versions of CMSs like Wordpress or Joomla. Also, so far the geographic deviation is only measured on daily basis. Observing changes in the location of the resolvers and taking the CMS into account might be helpful indicators whether a domain name is compromised.

In order to validate if a suspicious domain is actually used for malicious purposes, it can be sometimes possible to find the botnet C&C administration panel or the website with the malicious script on the website. Sood (2014) is using search engines like Google to find suspicious URLs for domain names. Automatising this might be challenging because of API rate limiting and the variety of URL patterns. Additionally, search engines might be useful to reduce the number of false positives. During this project, we often searched manually for a domain name on web-search engines in order to collect more information about a suspicious domain name. There, we sometimes observed that the search returned a high number of results within the last days. This was often a sign that the domain was benign and the increase of queries and high geographic deviation was rather caused by popular content on the website than malicious activities. Hence, counting the number of recent search-engine results could be a way to reduce the false positive rate. Again, rate limiting and limits of the search engine's API might cause difficulties.

So far, *SIDekICk* operates on a daily basis. By reducing the observation

epochs to half a day or a few hours, malicious domains could be detected even earlier which would be a bigger advantage over services like *Netcraft*.

For further validation of *SIDekICk* it would be of interest to apply it in other ccTLDs as well. Thereby, we would be able to evaluate if we can generalise our findings. Also, a cooperation with IT-security firms could allow us to validate our suspicions for some domain names.

Finally, it is one thing to detect malicious domains, but another to actually stop the malicious activities. This requires an organisational approach which is sketched out in the next section.

6.3 Post Malicious Domain Name Detection

Detecting a malicious domain name is only one step towards the ultimate goal of stopping the malign activity, identifying the responsible culprit and taking actions that such a misuse cannot happen again easily. Therefore, this section describes measures that can be taken by registries as soon as a domain name has been classified as malicious. Additionally, we describe how to proceed in case the registry has a suspicion that a domain name is involved in malicious activities but cannot verify this suspicion on its own.

6.3.1 Following the Chain of Responsibility

In case a third party has a complaint about content that is hosted on a *.nl* domain, SIDN has published a general chain of responsibility that describes which entity has to be contacted in order to take down the controversial content[1]:

1. The provider of the content
2. The provider of the website (registrant)
3. The firm that hosts the website
4. The provider of the internet access (registrar)
5. The registry (SIDN)

In case SIDN itself detects a malicious domain name, the same chain can be followed. However, we need to differentiate between a domain name that has been registered by a culprit directly and a domain name that is infected such that it takes part in malicious activities.

[1] `www.sidn.nl/a/nl-domain-name/complaining-about-the-content-of-a-website`

Newly Registered Domains If a domain name is registered for malicious purposes, then there are very likely no or false contact information provided on the website and domain registration information of the registrant are possibly fake as well. Therefore, the first two steps of the chain of responsibility can be skipped directly. As long as the web content is hosted at legitimate web hosting firms, the server or web-space is probably bought with fraudulent information as well. Thus, these firms have an interest in cancelling the contract for this domain as soon as possible, because there is a high chance that they might not get paid for their service and although they are not legally responsible for the content hosted on their servers most of the time (Stop Badware, 2011). If the domain is hosted at a so called bullet proof hoster, the hoster will very likely not respond to complaints. The domain name itself is probably registered with fake information as well which is an incentive for registrars to cancel the registration. Finally, SIDN has still the possibility to remove the domain from their zone file and thereby increase the barrier for accessing the website through DNS. SIDN has strict notice and take down (NTD) procedures and considers the take down of a domain name as *"a last resort*" in the fight against abuse (SIDN, 2014a).

Compromised Domains If malicious content is hosted on a compromised website, contact details of the registrant are usually correct, such that the website owner can be contacted directly. If the registrant has the technical knowledge to clean up the site from the malicious content, then the problem should be solved. Moore and Clayton (2009b) have shown that contacting the administrator of a website can be an efficient measure to take down phishing content. In case the registrant does not have the technical knowledge or does not respond to the complaint, then the hosting company can be contacted. A survey among people whose website got compromised showed that occasionally hosting companies actively or after notification remove malicious content from the compromised server (Stop Badware, 2012). If web hosting companies are notified about malicious content on one of their customer's websites, then 50 % of the firms do not react at all. In case they do respond, they reply withing 48 hours (Canali et al., 2013). For example, they suspend the account of the customer with the consequence that the benign service running on the server is interrupted as well or remove the malicious files. The notification of the authors included the URL that referred to the malicious content. If a complaint at the web hosting company is not successful, a registrar might take actions. However, suspending the account of the registrant has the consequence that benign content is not reachable as well. NTD carried out by SIDN is only an option in very serious cases.

6.3.2 Alternative Approaches

If we would try to take down a domain detected by *SIDekICk*, then we would face several issues while following the chain of responsibilities.

First, following this process is rather time consuming. If we detect a newly registered phishing domain, then we need to act within hours to effectively reduce the impact of the campaign (Aaron and Rasmussen, 2015). Second, we cannot block compromised domains that have been classified by *SIDekICk-Comp* due to the high false positive rate and the high chance that benign content might be blocked. Third, even if we would be almost certain that a domain is compromised, we often cannot provide the actual URL to the malicious content such that the informed web hosting company is probably less likely to react to the request. Finally, even if we would be able to successfully take down malicious content on a compromised website, the chance that a website gets reinfected is still around 20% (Moore and Clayton, 2009a).

The first issue can be solved rather easily. Submitting the detected phishing website manually to the *Netcraft* feed or to the Google Safe Browsing initiative[2] is an efficient way to reach a broad user base on the Internet. Many modern browsers rely on these services and block the content automatically as soon as the domain is listed. Taking down suspected compromised websites is a greater challenge, especially if we are not sure if the domain is actually infected.

One solution is to improve the precision of the classifier as described in Section 6.2. Another one is to rely on third parties to examine the suspicious domains further. For example, registries could collaborate with developers of intrusion detection systems and firewalls or anti-virus vendors. If for example a firewall detects traffic which is directed to one of the suspicious domains, then a rule could be triggered that puts files, which are downloaded from this domain, first into quarantine for further examination. Anti-virus vendors could use the list of suspicious domains as another indicator whether a client is infected. Also, ISPs could observe the traffic to these suspicious domains. If many clients request a previously unknown URL, then malicious content might be hosted on this site. In this approach, the privacy of the customers of the ISPs needs to be taken into account.

So far, SIDN does not have guidelines how to proceed with domains that have been detected by the research and operations teams but decide individually if and which actions should be taken.

[2] `https://www.google.com/safebrowsing/report_phish/`

6.4 Epilogue

Fighting malicious domain names is a community effort. We have shown that registries can contribute to this effort due to their broad and holistic view over their zone. We can expect that the impact of malicious domain names can be reduced if every registry would be able to actively fight abuse in their zones . However, DNS is a complex system. It involves many actors and stakeholders and only by involving as many of them in this fight a more secure Internet can be achieved in the end.

Bibliography

Aaron, G. and Rasmussen, R. (2015). Global phishing survey: Trends and domain name use in 2h2014. *APWG Industry Advisory*, 2H2014.

Antonakakis, M., Perdisci, R., Dagon, D., Lee, W., and Feamster, N. (2010). Building a dynamic reputation system for dns. In *USENIX security symposium*, pages 273–290.

Antonakakis, M., Perdisci, R., Lee, W., Vasiloglou II, N., and Dagon, D. (2011). Detecting malware domains at the upper dns hierarchy. In *USENIX Security Symposium*, pages 16–32.

Antonakakis, M., Perdisci, R., Nadji, Y., Vasiloglou II, N., Abu-Nimeh, S., Lee, W., and Dagon, D. (2012). From throw-away traffic to bots: Detecting the rise of dga-based malware. In *USENIX Security Symposium*, pages 491–506.

AsSadhan, B., Moura, J. M., Lapsley, D., Jones, C., and Strayer, W. T. (2009). Detecting botnets using command and control traffic. In *Network Computing and Applications, 2009. NCA 2009. Eighth IEEE International Symposium on*, pages 156–162. IEEE.

Association, D. D. C. (2015). Dutch data center report 2015. `http://www.dutchdatacenters.nl/uploads/4/4/0/6/44067373/dutch_data_center_report_2015.pdf`.

Biasini, N. (2015). Domain shadowing goes nuclear: A story in failed sophistication. `http://blogs.cisco.com/security/talos/nuclear-sophistication`. Retreived 2015-06-23.

Biasini, N. and Esler, J. (2015). Threat spotlight: Angler lurking in the domain shadows. `http://blogs.cisco.com/security/talos/angler-domain-shadowing`. Retreived 2015-03-31.

Bilge, L., Kirda, E., Kruegel, C., and Balduzzi, M. (2011). Exposure: Finding malicious domains using passive dns analysis. In *Proceedings of the Network and Distributed System Security Symposium, NDSS 2011*.

Binsalleeh, H., Ormerod, T., Boukhtouta, A., Sinha, P., Youssef, A., Debbabi, M., and Wang, L. (2010). On the analysis of the zeus botnet crimeware toolkit. In *Privacy Security and Trust (PST), 2010 Eighth Annual International Conference on*, pages 31–38. IEEE.

Brook, C. (2014). Latest instance of pony botnet pilfers 200k, 700k credentials. `https://threatpost.com/latest-instance-of-pony-botnet-pilfers-200k-700k-credentials/104463`. Retreived 2015-06-23.

Canali, D., Balzarotti, D., and Francillon, A. (2013). The role of web hosting providers in detecting compromised websites. In *Proceedings of the 22nd international conference on World Wide Web*, pages 177–188. International World Wide Web Conferences Steering Committee.

Čermák, M., Čeleda, P., and Vykopal, J. (2014). Detection of dns traffic anomalies in large networks. In *Advances in Communication Networking*, pages 215–226. Springer.

Chaney, A. J.-B. and Blei, D. M. (2012). Visualizing topic models. In *International AAAI Conference on Social Media and Weblogs, ICWSM*.

Chen, C.-M., Ou, Y.-H., and Tsai, Y.-C. (2010). Web botnet detection based on flow information. In *Computer Symposium (ICS), 2010 International*, pages 381–384. IEEE.

Chen, J. and Li, B. (2015). Evolution of exploit kits. `http://www.trendmicro.com/cloud-content/us/pdfs/security-intelligence/white-papers/wp-evolution-of-exploit-kits.pdf`. Trendmicro Whitepaper.

Choi, H., Lee, H., and Kim, H. (2009). Botgad: detecting botnets by capturing group activities in network traffic. In *Proceedings of the Fourth International ICST Conference on COMmunication System softWAre and middlewaRE*, page 2. ACM.

Cooper, G. and Herskovits, E. (1990). A bayesian method for constructing bayesian belief networks from databases. In *Proceedings of the Conference on Uncertainty in AI*, pages 86–94.

Davuth, N. and Sung-Ryul, K. (2013). Classification of malicious domain names using support vector machine and bi-gram method. *International Journal of Security and Its Applications*, 7(1):51–58.

Decker, A., Sancho, D., Kharouni, L., Goncharov, M., and McArdle, R. (2009). Pushdo/cutwail: a study of the pushdo/cutwail botnet. *Trend Micro Technical Report. TechRepublic*.

Dietrich, C., Rossow, C., Freiling, F., Bos, H., Van Steen, M., and Pohlmann, N. (2011). On botnets that use dns for command and control. In *Computer Network Defense (EC2ND), 2011 Seventh European Conference on*, pages 9–16.

Eshete, B. and Venkatakrishnan, V. (2014). Webwinnow: Leveraging exploit kit workflows to detect malicious urls. In *Proceedings of the 4th ACM conference on Data and application security and privacy*, pages 305–312. ACM.

Feily, M., Shahrestani, A., and Ramadass, S. (2009). A survey of botnet and botnet detection. In *Emerging Security Information, Systems and Technologies, 2009. SECURWARE'09. Third International Conference on*, pages 268–273. IEEE.

Fette, I., Sadeh, N., and Tomasic, A. (2007). Learning to detect phishing emails. In *Proceedings of the 16th international conference on World Wide Web*, pages 649–656. ACM.

Frosch, T., Kührer, M., and Holz, T. (2013). Predentifier: Detecting botnet c&c domains from passive dns data. *Advances in IT Early Warning, Fraunhofer Verlag*.

Futai, Z., Siyu, Z., and Weixiong, R. (2013). Hybrid detection and tracking of fast-flux botnet on domain name system traffic. *Communications, China*, 10(11):81–94.

Grier, C., Ballard, L., Caballero, J., Chachra, N., Dietrich, C. J., Levchenko, K., Mavrommatis, P., McCoy, D., Nappa, A., Pitsillidis, A., et al. (2012). Manufacturing compromise: the emergence of exploit-as-a-service. In *Proceedings of the 2012 ACM conference on Computer and communications security*, pages 821–832. ACM.

Gu, G., Perdisci, R., Zhang, J., Lee, W., et al. (2008a). Botminer: Clustering analysis of network traffic for protocol-and structure-independent botnet detection. In *USENIX Security Symposium*, pages 139–154.

Gu, G., Zhang, J., and Lee, W. (2008b). Botsniffer: Detecting botnet command and control channels in network traffic. In *Proceedings of the 15th Annual Network and Distributed System Security Symposium*.

Guyon, I., Boser, B., and Vapnik, V. (1993). Automatic capacity tuning of very large vc-dimension classifiers. In *Advances in Neural Information Processing Systems*, pages 147–155. Morgan Kaufmann.

Hao, S., Feamster, N., and Pandrangi, R. (2010). An internet-wide view into dns lookup patterns. *School of Computer Science, Georgia Tech, Tech. Rep*.

Hao, S., Feamster, N., and Pandrangi, R. (2011). Monitoring the initial dns behavior of malicious domains. In *Proceedings of the 2011 ACM SIGCOMM conference on Internet measurement conference*, pages 269–278. ACM.

Hesselman, C., Jansen, J., Wullink, M., Vink, K., and Simon, M. (2014). A privacy framework for dns big data applications. *Privacy & Informatie*, 6.

Holz, T., Gorecki, C., Rieck, K., and Freiling, F. C. (2008). Measuring and detecting fast-flux service networks. In *Proceedings of the Network and Distributed System Security Symposium, NDSS*.

Howard, F. (2012). Hacked go daddy sites infecting users with ransomware. `https://nakedsecurity.sophos.com/2012/11/23/hacked-go-daddy-ransomware/`. Retrieved 2015-01-13.

Huang, S.-Y., Mao, C.-H., and Lee, H.-M. (2010). Fast-flux service network detection based on spatial snapshot mechanism for delay-free detection. In *Proceedings of the 5th ACM Symposium on Information, Computer and Communications Security*, pages 101–111. ACM.

Jiang, N., Cao, J., Jin, Y., Li, L., and Zhang, Z.-L. (2010). Identifying suspicious activities through dns failure graph analysis. In *Network Protocols (ICNP), 2010 18th IEEE International Conference on*, pages 144–153. IEEE.

Kaspersky Lab (2015). A single ddos attack can cost a company more than 400,000 usd. `http://www.kaspersky.com/about/news/business/2015/A-single-DDoS-attack-can-cost-a-company-more-than-400000-dollar`. Retreived 2015-07-13.

Kheir, N., Tran, F., Caron, P., and Deschamps, N. (2014). Mentor: Positive dns reputation to skim-off benign domains in botnet c&c blacklists. In *ICT Systems Security and Privacy Protection*, pages 1–14. Springer.

Kimberly (2014). Cve-2013-2729 and andromeda 2.9 - a massive hsbc themed email campaign - andromeda botnet. `http://stopmalvertising.com/spam-scams/cve-2013-2729-and-andromeda-2.9-a-massive-hsbc-themed-email-campaign/andromeda-botnet.html`. Retreived 2015-03-31.

Kruse, P. (2014). The rovnix reincarnation. `https://www.csis.dk/en/csis/news/4472/`. Retreived 2015-01-23.

Lee, J., Kwon, J., Shin, H.-J., and Lee, H. (2010). Tracking multiple c&c botnets by analyzing dns traffic. In *Secure Network Protocols (NPSec), 2010 6th IEEE Workshop on*, pages 67–72. IEEE.

Lee, J. and Lee, H. (2014). Gmad: Graph-based malware activity detection by {DNS} traffic analysis. *Computer Communications*, 49(0):33 – 47.

Li, W., Xie, S., Lui, J., and Zhu, X. (2013). A detection method for botnet based on behavior features. In *Proceedings of the 2nd International Conference On Systems Engineering and Modeling*.

Lloyd, S. (2015). Nominet research blog: A study in strange traffic. `http://research.nominet.org.uk/post/112122385676/a-study-in-strange-traffic`. Retreived 2015-07-01.

Mahjoub, D., Reuille, T., and Toonk, A. (2014). Catching malware en masse: Dns and ip style. In *Black Hat USA 2014 Proceedings*.

Martinez-Bea, S., Castillo-Perez, S., and Garcia-Alfaro, J. (2013). Real-time malicious fast-flux detection using dns and bot related features. In *PST*, pages 369–372.

Mazzariello, C. (2008). Irc traffic analysis for botnet detection. In *Information Assurance and Security, 2008. ISIAS'08. Fourth International Conference on*, pages 318–323. IEEE.

Mimoso, M. (2014). Matsnu botnet dga discovers power of words. `https://threatpost.com/matsnu-botnet-dga-discovers-power-of-words/109426`. Retreived 2015-01-23.

Mockapetris, P. (1987a). Rfc 1034: Domain names-concepts and facilities. `ftp://ftp.isi.edu/in-notes/rfc1034.txt`.

Mockapetris, P. (1987b). Rfc 1035: Domain namesimplementation and specification. `http://www.ietf.org/rfc/rfc1035.txt`.

Moore, T. and Clayton, R. (2009a). Evil searching: Compromise and recompromise of internet hosts for phishing. In *Financial Cryptography and Data Security*, pages 256–272. Springer.

Moore, T. and Clayton, R. (2009b). The impact of incentives on notice and take-down. In *Managing Information Risk and the Economics of Security*, pages 199–223. Springer.

Morales, J. A., Al-Bataineh, A., Xu, S., and Sandhu, R. (2009). Analyzing dns activities of bot processes. In *Malicious and Unwanted Software (MALWARE), 2009 4th International Conference on*, pages 98–103. IEEE.

Murphy, K. (1998). A brief introduction to graphical models and bayesian networks. `http://www.cs.ubc.ca/~murphyk/Bayes/bnintro.html`. Retrieved 2015-01-21.

Nagaraja, S., Mittal, P., Hong, C.-Y., Caesar, M., and Borisov, N. (2010). Botgrep: Finding p2p bots with structured graph analysis. In *USENIX Security Symposium*, pages 95–110.

Nazario, J. (2012). Measuring botnet populations. `http://www.arbornetworks.com/asert/2012/05/measuring-botnet-populations/`. Retrieved 2015-01-10.

Nazario, J. and Holz, T. (2008). As the net churns: Fast-flux botnet observations. In *Malicious and Unwanted Software, 2008. MALWARE 2008. 3rd International Conference on*, pages 24–31. IEEE.

Passerini, E., Paleari, R., Martignoni, L., and Bruschi, D. (2008). Fluxor: Detecting and monitoring fast-flux service networks. In *Detection of intrusions and malware, and vulnerability assessment*, pages 186–206. Springer.

Pelleg, D., Moore, A. W., et al. (2000). X-means: Extending k-means with efficient estimation of the number of clusters. In *ICML*, pages 727–734.

Perdisci, R., Corona, I., Dagon, D., and Lee, W. (2009). Detecting Malicious Flux Service Networks through Passive Analysis of Recursive DNS Traces. In *Annual Computer Security Applications Conference*, pages 311–320.

Poese, I., Uhlig, S., Kaafar, M. A., Donnet, B., and Gueye, B. (2011). Ip geolocation databases: Unreliable? *SIGCOMM Comput. Commun. Rev.*, 41(2):53–56.

Porras, P. (2009). Inside risks reflections on conficker. *Communications of the ACM*, 52(10):23–24.

Porras, P., Saidi, H., and Yegneswaran, V. (2009a). An analysis of conficker's logic and rendezvous points. *Computer Science Laboratory, SRI International, Tech. Rep.*

Porras, P., Saïdi, H., and Yegneswaran, V. (2009b). A foray into conficker's logic and rendezvous points. In *USENIX Workshop on Large-Scale Exploits and Emergent Threats*.

Ramachandran, A., Feamster, N., and Dagon, D. (2006). Revealing botnet membership using dnsbl counter-intelligence. *Proc. 2nd USENIX Steps to Reducing Unwanted Traffic on the Internet*, pages 49–54.

Ramage, D. (2007). Hidden markov models fundamentals. `http://cs229.stanford.edu/section/cs229-hmm.pdf`. Stanford University. CS229 Section Notes.

Rascagnres, P. (2015). The andromeda/gamarue botnet is on the rise again. `https://blog.gdatasoftware.com/blog/article/`

the-andromedagamarue-botnet-is-on-the-rise-again.html. Retreived 2015-03-31.

Riden, J. (2008). How fast-flux service networks work. http://honeynet.org/node/132. Retrieved 2015-01-12.

Rodríguez-Gómez, R. A., Maciá-Fernández, G., and García-Teodoro, P. (2013). Survey and taxonomy of botnet research through life-cycle. *ACM Computing Surveys (CSUR)*, 45(4):45.

Schiavoni, S., Maggi, F., Cavallaro, L., and Zanero, S. (2014). Phoenix: Dga-based botnet tracking and intelligence. In *Detection of Intrusions and Malware, and Vulnerability Assessment*, pages 192–211. Springer.

Scikit-learn (2015a). Scikit-learn documentation 1.10: Decision trees. http://scikit-learn.org/stable/modules/tree.html. Retreived 2015-07-08.

Scikit-learn (2015b). Scikit-learn documentation 1.4: Support vector machines. http://scikit-learn.org/stable/modules/svm.html. Retreived 2015-07-08.

SECURE64 (2014). Water torture: A slow drip dns ddos attack. https://blog.secure64.com/?p=377. Retreived 2015-07-17.

SIDN (2014a). Notice and take down code for .nl domain names. https://www.sidn.nl/dotAsset/b2139048-3a43-43d1-9ed9-b732b7c2fe29.pdf. Retreived 2015-07-15.

SIDN (2014b). Sidn statistics. https://www.sidn.nl/a/knowledge-and-development/statistics. Retreived 2014-01-19.

Sood, A. (2014). Exploiting fundamental weaknesses in botnet command and control panels. In *Black Hat USA 2014 Proceedings*.

Soumenkov, I. (2012). Flashfake mac os x botnet confirmed. https://securelist.com/blog/incidents/32661/flashfake-mac-os-x-botnet-confirmed-25/. Retreived 2015-06-24.

Stalmans, E., Hunter, S. O., and Irwin, B. (2012). Geo-spatial autocorrelation as a metric for the detection of fast-flux botnet domains. In *Information Security for South Africa (ISSA), 2012*, pages 1–7. IEEE.

Statsoft (1995). Support vector machines. http://www.statsoft.com/Textbook/Support-Vector-Machines. Retreived 2015-01-19.

Stone-Gross, B., Cova, M., Cavallaro, L., Gilbert, B., Szydlowski, M., Kemmerer, R., Kruegel, C., and Vigna, G. (2009). Your botnet is my botnet: analysis of a botnet takeover. In *Proceedings of the 16th ACM conference on Computer and communications security*, pages 635–647. ACM.

Stone-Gross, B., Cova, M., Gilbert, B., Kemmerer, R., Kruegel, C., and Vigna, G. (2011). Analysis of a botnet takeover. *Security & Privacy, IEEE*, 9(1):64–72.

Stop Badware (2011). Web hosting provider liability for malicious content. `http://www.nist.gov/itl/upload/StopBadware_Web-Hosting-Provider-Liability-for-Malicious-Content.pdf`. Whitepaper, Retreived 2015-07-15.

Stop Badware (2012). Compromised websites - an owners perspective. `https://www.stopbadware.org/files/compromised-websites-an-owners-perspective.pdf`. Retreived 2015-07-15.

Verisign (2014). The verisign domain report. *The Domain Name Industry Brief*, 11.

Verisign (2015). The domain name industry brief. *The Domain Name Industry Brief*, 12.

Villamarín-Salomón, R. and Brustoloni, J. C. (2008). Identifying botnets using anomaly detection techniques applied to dns traffic. In *Consumer Communications and Networking Conference, 2008. CCNC 2008. 5th IEEE*, pages 476–481. IEEE.

Villamarín-Salomón, R. and Brustoloni, J. C. (2009). Bayesian bot detection based on dns traffic similarity. In *Proceedings of the 2009 ACM symposium on Applied Computing*, pages 2035–2041. ACM.

Wisniewski, C. (2009). Aftertaste - the domain tasting era has quietened to a whisper. `https://nakedsecurity.sophos.com/2009/08/15/guest-blog-aftertaste-domain-tasting-era-quietened-whisper/`. Retreived 2015-07-20.

Wisniewski, C. (2015). When penguins attack - linuxs role in the malware ecosystem. In *Talk at the BSides Boston*.

Wullink, M. (2015). Dns big data bij sidn labs. `https://www.sidnlabs.nl/laatste-berichten/nieuwsdetail/article/dns-big-data-bij-sidn-labs/`. Retreived 2015-01-21.

Yarochkin, F., Kropotov, V., Huang, Y., Ni, G.-K., Kuo, S.-Y., and Chen, I.-Y. (2013). Investigating dns traffic anomalies for malicious activities. In

Dependable Systems and Networks Workshop (DSN-W), 2013 43rd Annual IEEE/IFIP Conference on, pages 1–7. IEEE.

Zeydan, H., Selamat, A., and Salleh, M. (2014). Survey of anti-phishing tools with detection capabilities. In *Biometrics and Security Technologies (ISBAST), 2014 International Symposium on*, pages 214–219.

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

Zhao, D., Traore, I., Sayed, B., Lu, W., Saad, S., Ghorbani, A., and Garant, D. (2013). Botnet detection based on traffic behavior analysis and flow intervals. *Computers & Security*, 39:2–16.

Zhu, Z., Yegneswaran, V., and Chen, Y. (2009). Using failure information analysis to detect enterprise zombies. In *Security and Privacy in Communication Networks*, pages 185–206. Springer.