COMPUTERS & SECURITY 77 (2018) 138–161

Available online at www.sciencedirect.com

ScienceDirect

journal homepage: www.elsevier.com/locate/cose

Computers & Security

DomainChroma: Building actionable threat intelligence from malicious domain names☆

Daiki Chiba[a,*], *Mitsuaki Akiyama*[a], *Takeshi Yagi*[a], *Kunio Hato*[a], *Tatsuya Mori*[b], *Shigeki Goto*[b]

[a] *NTT Secure Platform Laboratories, Tokyo, Japan*
[b] *Waseda University, Tokyo, Japan*

ARTICLE INFO

Article history:
Received 18 September 2017
Revised 27 February 2018
Accepted 29 March 2018
Available online 6 April 2018

Keywords:
Malicious domain name
Categorization
Actionable threat intelligence
Defense point
Domain blacklist
Abuse report

ABSTRACT

Since the 1980s, domain names and the domain name system (DNS) have been used and abused. Although legitimate Internet users rely on domain names as indispensable infrastructures for using the Internet, attackers use or abuse them as reliable, instantaneous, and distributed attack infrastructures. However, there is a lack of complete understanding of such domain-name abuses and methods for coping with them.

In this study, we designed and implemented a unified analysis system combining current defense solutions to build actionable threat intelligence from malicious domain names. The basic concept underlying our system is malicious domain name *chromatography*. Our analysis system can distinguish among mixtures of malicious domain names for websites. On the basis of this concept, we do not create a hodgepodge of current solutions but design separation of abused domain names and offer actionable threat intelligence or defense information by considering the characteristics of malicious domain names as well as the possible defense solutions and points of defense. Finally, we evaluated our analysis system and defense-information output using a large real dataset to show the effectiveness and validity of our system.

1. Introduction

Domain names have been an integral part of today's Internet since the 1980s (Mockapetris, 1983a; 1983b; Vixie, 2007). While the Internet cannot virtually function without domain names, cyber attackers also use domain names and the domain name system (DNS) as a reliable, instantaneous, and distributed infrastructure for conducting attacks. For example, attackers register similar domain names to legitimate services or popular brands to deceive users into downloading malware or unwanted programs. Another common example is the so-called command and control (C&C), wherein attackers use domain names as rendezvous points of malware-infected hosts to control them and launch other attacks such as denial-of-service (DoS) and spam email.

Countermeasures, such as detection and filtering of malicious domain names owned/used by attackers, have been

☆ This paper is the extended version of the paper presented at IEEE COMPSAC 2017 (Chiba et al., 2017).

* Corresponding author.
E-mail address: daiki.chiba@ieee.org (D. Chiba).

https://doi.org/10.1016/j.cose.2018.03.013

studied and implemented for many years (Antonakakis et al., 2010; 2011; 2012; Bilge et al., 2011; Felegyhazi et al., 2010; Hao et al., 2016; Ma et al., 2009; Rahbarinia et al., 2015; Sato et al., 2010). Nevertheless, abuse of new domain names has continued and remains a significant threat. Moreover, there is no single common defense solution against domain-name abuses because each malicious domain name has different characteristics. If we fail to choose the right countermeasure for each malicious domain name, the countermeasure will be basically ineffective in practice. For example, some malicious domain names are created by abusing legitimate services including online advertising, web hosting, and the dynamic DNS. If we filter domain names used in such legitimate services, we may prevent users from accessing legitimate services and disrupt legitimate businesses. In fact, a legitimate dynamic DNS provider suffered from such a situation (Microsoft, 0000; NoI, 0000), i.e., their domain name was suspended by a false choice of defenses. Other malicious domain names are purposely set up using an algorithm called the domain generation algorithm (DGA). If we hesitate to filter these domain names, we will not be able to decrease the threat of cyber attacks.

A key challenge is to determine the optimal defense solution for each malicious domain name. This paper is intended to reveal *what, where, how*, and *until when* countermeasures need to be taken against such malicious domain names. We focus on the relationships between domain-name categories and practical defense solutions to determine how best to use detected malicious domain-name information. In reality, there is a significant distance from simply detecting malicious domain names to using them for making the Internet safer. In particular, malicious domain names differ significantly depending on their characteristics such as their hierarchical structure, back-end services offering them, and operational situations. Thus, we need to consider these characteristics for each malicious domain name and determine the best defense solution to prevent the filtering of any legitimate services or businesses. Therefore, we should first categorize or classify malicious domain names according to their characteristics then determine the defense solution suitable for each domain name to build actionable threat intelligence.

In this study, we designed and implemented a unified analysis system combining current defense solutions for obtaining practical and actionable threat intelligence from malicious domain names. The concept behind our system is referred to as malicious domain name *chromatography*, which is used for the separation of mixtures composed of various types of malicious domain names for websites. On the basis of this concept, we do not create a hodgepodge of current solutions but design separation of malicious domain names and offer defense information by considering the characteristics of the malicious domain names as well as the possible defense solutions and points of defense.

Our main contributions are summarized as follows.

- We developed a taxonomy of malicious domain names and provide systematized knowledge of the latest defense solutions with points of defense.
- We designed and implemented an analytical system, which systematically determines the optimal defense solution for each malicious domain name.
- We evaluated our analysis pipeline and defense-information output using a large real dataset to show both the effectiveness and validity of our system. In particular, we show that over 70% of malicious domain names require only DNS-level defense with no collateral damage of legitimate accesses.

This paper is an extended version of our conference paper (Chiba et al., 2017). In this new paper, we include a new method and output, new analyses, provide new insights into our proposed system, and discuss the landscape of attacks against domain names. Our new contributions are summarized as follows.

- We developed a method for determining the expiration date for the defense information created with our proposed system for obtaining effective defense solutions (Section 3.5).
- We propose to use our system for creating abuse reports to provide owners or administrators with proper defense information (Section 3.6).
- We newly conducted additional analyses including classification accuracy of defined categories (Section 4.2), detailed evaluation of our system's output (Section 4.3), comparison of collateral damage (Section 4.4), and validation of the expiration date (Section 4.5).
- We provide completely new experimental results based on newly subscribed 25 types of commercial and public blacklists, including over 1.6 million domain names (Section 4.6).
- We describe the landscape of attacks against domain names while surveying the current state-of-the-art defense solutions against individual attacks (Section 5).

The rest of this paper is organized as follows. In Section 2, we show a taxonomy of malicious domain names and consider the possible defense solutions and points of defense. We discuss our analysis system DomainChroma in Section 3. We give a detailed explanation of datasets and the evaluation results in Section 4. In Section 5, we survey and describe the landscape of attacks and defenses against domain names in detail. Finally, we conclude this paper in Section 6.

2. Chromatography of malicious domain names

We define the concept of malicious domain name *chromatography*. Traditionally applied in chemistry, chromatography separates a mixture into its components based on their different chemical characteristics. Our chromatography handles mixtures composed of various types of malicious domain names for websites. We explore the separation of malicious domain names and offer actionable threat intelligence or defense information by considering both the characteristics of the malicious domain names and possible points of defense. Fig. 1 illustrates how this chromatography concept can be used to analyze malicious domain names. The mixture containing various categories of malicious domain names is separated into the categories presented later in Section 2.1.

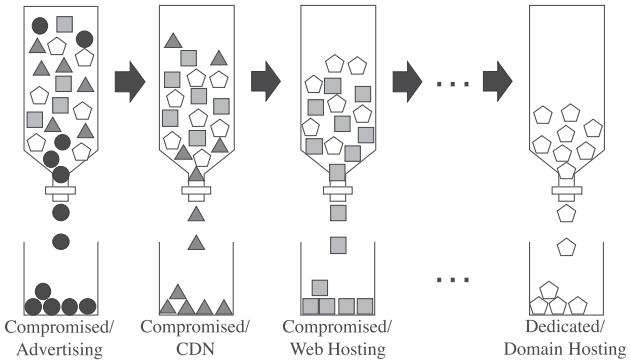

Fig. 1 – Categorizing malicious domain names by using chromatography-based concept.

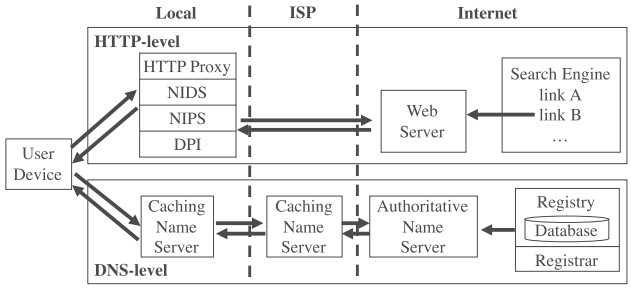

Fig. 2 – Our target points of defense against malicious domain names.

2.1. *Characteristics of malicious domain names*

In most cyber attacks, domain names are used to deliver malicious content (e.g., exploit content and malware) and to command/control malware-infected hosts because domain names and the DNS are easy to use, reliable, and become distributed systems. We investigated the characteristics of malicious domain names to further classify them into two categories, i.e., *compromised* and *dedicated*.

Compromised. This category contains malicious domain names abusing legitimate services. We consider this category because such malicious domain names are originally intended to offer legitimate services to Internet users, and we should not simply stop such domain names and filter accesses to them. We explore the compromised category and discuss how to classify it in Section 3.2.1.

Dedicated. This category contains malicious domain names prepared exclusively for malicious purposes. We consider this category because malicious domain names differ significantly relative to compromised domain names. Based on this idea, we should clearly distinguish dedicated malicious domain names from compromised domain names in terms of providing mitigations or countermeasures. We discuss the details of this dedicated category and illustrate how to classify it in Section 3.2.2.

2.2. *Points of defense against malicious domain names*

We summarize possible points of defense and corresponding defense methods against malicious domain names in terms of building a realistic remedy strategy. Specifically, we divide these points into two levels: *HTTP-level* and *DNS-level*, as shown in Fig. 2.

HTTP-level points of defense. This level includes three components that are involved in web communication using HTTP/HTTPS, i.e., security appliances, web servers, and search engines. Although this paper primarily focuses on domain names and the DNS, we also consider HTTP-level defenses because there is a close connection between HTTP and the DNS. For example, most HTTP communication uses DNS-name resolution; thus, we can defend against malicious HTTP communication at both the DNS and HTTP levels. The following three components can only defend against attacks using malicious domain names if attacks use HTTP, e.g., distributing exploit content or malware using websites, hosting phishing websites, and operating malware-infected hosts using C&C.

Security appliances include HTTP proxies, network intrusion detection/prevention systems (NIDS/NIPS), and deep packet inspection (DPI). They can filter HTTP communication pointing to malicious domain names.

Web servers on the Internet can be critical points of defense if they serve malicious content. The defense on web servers is straightforward, i.e., deleting corresponding malicious content on the web servers. To deploy this defense, abuse reports can be sent to content owners, server administrators (Li et al., 2016b), and national or regional computer emergency response team (CERT) organizations.

Search engines are one of the most frequently used web applications. Most users access websites based on search results generated by a search engine. If search engines display a link to malicious content, users are susceptible to cyber attacks. Thus, search engines should filter links that point to malicious content.

DNS-level points of defense This level includes three major components in the DNS, i.e., caching name servers, authoritative name servers, and domain registries/registrars.

Caching name servers are generally deployed in local area networks or Internet service provider (ISP) networks. The defense in caching name servers involves primarily filtering user access to malicious domain names based on domain name blacklists.

Authoritative name servers are deployed in each DNS zone, primarily at the effective second-level domain (e2LD) level. An e2LD is the smallest unit of a domain name that can be registered by Internet users. The e2LD part can be easily extracted from any domain name by using the Public Suffix List (Mozilla foundation, 0000). The defenses in authoritative name servers include filtering and updating zone data. Filtering can be used to block DNS queries pointing to a blacklisted domain name to prevent users from accessing a malicious domain name. Updating zone data involves deleting a malicious domain name record so that the domain name cannot be resolved.

Domain registries/registrars manage domain name registrations. The registry manages the registration of domain names within large DNS zones, such as top-level domains (TLDs). The registrar is a service provider that connects the registries to manage domain databases. Malicious domain names can be defended by anti-abuse actions on the master

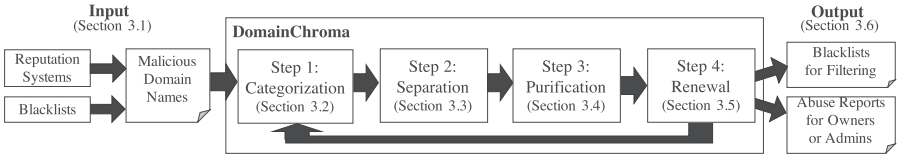

Fig. 3 – Overview of our analysis system DomainChroma.

database of each registry such as deleting a domain name. For example, VeriSign, Inc., which is a `.com` and `.net` registrar, can take actions by complying with court orders and law enforcement (Ver, 0000), and Afilias Limited, which is a `.org` and `.info` registrar, can delete DNS records based on abuse reports (Afi, 0000).

3. Analysis system: DomainChroma

We designed an analysis system called DomainChroma that takes into account both characteristics of malicious domain names discussed in Section 2.1 and each point of defense discussed in Section 2.2. Our system was designed to systematize the knowledge of state-of-the-art defense techniques, enabling detection of malicious domain names and optimal defense actions. Specifically, we attempt to reveal *what, where, how* and *until when* countermeasures need to be taken against malicious domain names, thus securing domain names and the DNS for legitimate Internet users. To this end, we implemented the *chromatography* concept introduced in Section 2. Fig. 3 is an overview of our analysis system DomainChroma. DomainChroma consists of four steps: *categorization, separation, purification*, and *renewal*. The order of these steps is designed to reflect the demand in defense operations against malicious domain names. Note that distinguishing malicious domain names from a mixture of malicious and *legitimate* domain names is out of the scope of our paper. This paper focuses on deciding *actions* based on already detected malicious domain names. The step-by-step details of each step are presented in the following sections.

3.1. *Input: malicious domain names*

The DomainChroma input is expected to contain malicious domain names provided by current domain-reputation systems (Antonakakis et al., 2010; 2012; Bilge et al., 2011) or any kind of domain name blacklist, with a low probability of obvious legitimate domain names. Such malicious domain names are engaged in various types of cyber attacks (e.g., drive-by download, malware download and C&C, and phishing).

As explained in Section 2.1, it is difficult to effectively install defenses AFTER obtaining such malicious domain names because such domain names do NOT mean that we can simply stop or filter them. We should be careful of the risk of collateral damage caused by excessive filtering of each malicious domain name. To this end, DomainChroma automatically analyzes the input malicious domain names and provides the best defense solution for each malicious domain name to reduce the burden of administrators.

At this point, we know *what* domain names to be targeted for actions, but *where, how*, and *until when* these actions should be implemented are unknown. DomainChroma proceeds through the following steps to determine the actions for the optimal defense against malicious domain names.

3.2. *Step 1: Categorization*

The first step of DomainChroma is categorization. The input malicious domain names are categorized into *compromised* or *dedicated* since these two categories have different characteristics in terms of deciding countermeasures, as discussed in Section 2.1.

3.2.1. *Compromised*

We define compromised malicious domain names as being composed of the following three sub-categories, i.e., advertising, content delivery network (CDN), and web hosting. This section explains the definitions of these sub-categories and how to identify them in detail.

Advertising. Online advertisements are commonly used in most websites to generate revenue. Basically, such advertisements match content publishers and advertisers to maintain advertising ecosystems. Online advertising is not intended to abuse domain names nor engage in cyber attacks. However, cyber attackers have used this ecosystem as an attack vector to reach target users effectively. For example, Zarras et al. showed that 1% of online advertisements are used to lead users to malicious content (Zarras et al., 2014). In addition, Xing et al. found that common web-browser extensions deliver malware via the advertising ecosystem and have affected more than 600,000 users (Xing et al., 2015). If we detect malicious domain names from a specific advertising provider, we should consider the ecosystem and take appropriate countermeasures against specific malicious content. Thus, we categorize advertising domain names so as not to filter legitimate advertisements excessively. To identify domain names in advertising ecosystems, we leverage the advertising/tracking servers provided by hpHosts (hpH, 0000) and the EasyList (Eas, 0000) provided by AdBlock Plus (Adb, 0000) to identify domain names engaged in advertising ecosystems.

CDN. A CDN delivers web content to end users with a distributed and efficient infrastructure. Typically, it is used to reduce the latency of accessing web content (CDN, 0000). The content of websites using a CDN is copied to multiple distributed CDN servers so that the users can access the content from a nearby server rather than the original web server. A CDN is essentially used by legitimate users or companies; however, attackers have used CDNs as a reliable infrastructure to distribute malicious content (Chen et al., 2016a). Thus, we should filter specific content or the URLs rather than the

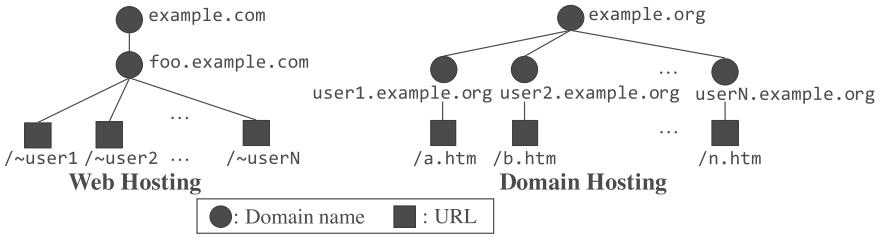

Fig. 4 – Structure of web hosting and domain hosting.

CDN domain names to prevent users from accessing malicious content while maintaining legitimate services. To do this, we should identify which domain names are used by the CDNs. Specifically, we collect IP address ranges used by popular CDNs (e.g., Amazon CloudFront (Amazon CloudFront, 0000), CloudFlare (CloudFlare, 0000), and Fastly (Fastly, 0000)) to detect domain names using CDNs. In addition, we classify CDN domain names based on the typical characteristics of canonical name (CNAME) records (Web, 0000), (Web, 0000). For example, CNAME records can include `cdn` in the domain name; thus, we classify CDN domain names based on such rules.

Web hosting. We define web hosting as involving domain names that use shared hosting services, including cloud services and file-sharing services. Since web hosting is an economical option for users to host a website, the number of web hosting services is increasing dramatically. Cyber attackers use web hosting services as an attack infrastructure, e.g., to host malicious websites or malware (Akiyama et al., 2011; Canali et al., 2013; Stokes et al., 2010). In this paper, we define web hosting as domain names that have multiple owner URLs (Fig. 4, left). We consider two methods of collecting domain names using web hosting services. One method uses a known web hosting provider list (Web, 0000) to specify web hosting-related domain names. The other collecting method uses a search-engine API (Microsoft Corporation, 0000). We query the API to extract URLs under the target domain names and crawl each URL to heuristically detect whether the target domain name uses web hosting services. For example, if a user can create an individual directory (e.g., `/~user1`) under the target domain name (e.g., `foo.example.com`), we consider the domain name is in web hosting.

3.2.2. *Dedicated*

We define dedicated malicious domain names as being composed of the following nine sub-categories, i.e., DGA, re-registration, sinkholing, parking, typosquatting, no-URLs, dynamic DNS, gratuitous, and domain hosting. We define these sub-categories and how to classify them below.

DGA. A DGA dynamically produces domain names primarily used as rendezvous points between attackers and victims or malware-infected hosts. A domain name generated using this algorithm is called an automatically generated domain (AGD) (Plohmann et al., 2016). Domain generation algorithms generate a huge number of distinct AGDs then use only a small subset of generated domain names for their actual malicious activities, such as C&C communication. Attackers use a DGA to make blacklisting or taking down C&C infrastructure infeasible. Automatically generated domains are only used for malicious purposes, and we should filter AGDs to specify malware-infected hosts and prevent users from cyber attacks. To identify AGDs, we leverage the linguistic features of AGDs (Bilge et al., 2011; Schiavoni et al., 2014; Yadav et al., 2010) and take a machine learning approach. Specifically, we extract linguistic features, train a machine-learning classifier, and detect AGDs. Our linguistic features are composed of n-grams, entropy, the length of domain names, and consonant ratio. We apply a supervised machine-learning algorithm using a random forest (Breiman, 2001) that relies on the majority vote of many decision trees with randomly selected features to train the classifier and separate AGDs from non-AGDs. The details of the data used to create the classifier and the evaluation results are described in Section 4.2.

Re-registration. This sub-category contains maliciously re-registered domain names that were originally legitimate domain names. Generally, a domain name has a valid period of one or more years. If a domain owner or registrant does not renew the domain registration, the domain will expire and be available for re-registration. For a more detailed explanation of the re-registration process, the reader can refer to recent studies (Hao et al., 2016; Lauinger et al., 2016; Lever et al., 2016). Expired domain names, particularly popular domain names, tend to be targeted and immediately re-registered by attackers for malicious purposes, such as phishing attacks. To detect such re-registered malicious domain names, we use crawling/parsing WHOIS data and detect re-registration events. When parsing WHOIS data, we use a method similar to that used in a previous study (Liu et al., 2015), i.e., we statistically label each domain name's WHOIS information such as the WHOIS server, name server, creation date, update date, and expiration date. To detect re-registration events, we use heuristic rules based on our hypothesis that attackers cannot easily re-register a domain name and simultaneously maintain name servers and creation date. Thus, if we detect a change in name servers and creation date in our time-series WHOIS data, we consider the domain name as re-registered.

Sinkholing. Security researchers and organizations often use a countermeasure called sinkholing to take control of malware C&C domain names to change DNS records to forward C&C traffic from malware-infected hosts to their controlled DNS servers to render C&C harmless (Chen et al., 2016b; Kührer et al., 2014; Rahbarinia et al., 2013; Rossow et al., 2013). In terms of defending users or cleaning malware-infected hosts, we should identify sinkholed malicious domain names because such domain names will not be used for legitimate services. To identify such sinkholed domain names, we take

two steps, i.e., we collect name server records for a target domain name and match the record with known sinkholing information. We actively send DNS queries to domain names to collect corresponding name server records in a similar way to that in a previous study (Kountouras et al., 2016). Then, we match collected name server records with known name server records used in a sinkholing operation. This way works well because legitimate security researchers and organizations tend to set their dedicated name server records for sinkholing to indicate that they purposely operate sinkholed domain names (Kührer et al., 2014).

Parking. Parking is a service used to monetize currently unused domain names that display advertisements. Cyber attackers tend to use such parking services to monetize malicious traffic resulting from malware infection or phishing attacks (Alrwais et al., 2014; Li et al., 2013; Vissers et al., 2015). Parked domain names used for malicious purposes should be filtered to hamper monetization related to cyber attacks. To identify such parked domain names, we use recently published parking-specific information (Vissers et al., 2015). Specifically, we take a similar approach to the previously described sinkholing identification, i.e., we first collect corresponding name server records for each domain name then match those records to known name-server records used in parking services. This approach works well because one needs to set specific name server records of one's domain name to use parking services.

Typosquatting. Typosquatting is generally defined as an attack technique to register similar domain names to popular or legitimate services (Agten et al., 2015; Khan et al., 2015; Szurdi et al., 2014; Wang et al., 2006). The aim is to monetize traffic generated from a typing error or conduct phishing attacks. Such typosquatting domain names are almost always created for non-legitimate purposes; thus, we should detect and filter such domain names and the traffic directed to them. To detect such malicious domain names engaged in typosquatting, we designed a typosquatting classifier. This classifier involves three steps, i.e., extracting possible origin domain names, filtering, and detecting. When extracting origin-domain-name candidates, we consider five typo models for an input domain name (e.g., `example.com`): addition (e.g., `exxample.com`), deletion (e.g., `xample.com`), substitution (e.g., `ezample.com`), transposition (e.g., `xeample.com`), and supplemental (e.g., `ex.ample.com`). Then, we filter the candidates based on the following two heuristics. One is whether the candidate domain name is within fat finger distance from the input malicious domain name. The fat finger distance means adjacent letters on a US QWERTY keyboard where a typo may occur. The other heuristic is whether the candidate domain name is listed in a popular domain name ranking list, such as Alexa Topsites (Ale, 0000). Attackers tend to target more popular domain names; thus, this filtering is effective for finding true origin domain names. Finally, we detect typosquatting domain names based on features, such as domain name length, usage of NXDOMAIN wildcarding, and name server records.

No-URLs. We define no-URLs as domain names that are considered only in the DNS protocol and have no URLs under the domain name. The reason we must define this subcategory is that our objective is not only detecting malicious domain names but also offering optimal defensive information for current realistic network environments and operators, as discussed in Section 2. For example, a malicious domain name that has no URLs can immediately be filtered in a caching name server in a network because there is no risk of excessively filtering legitimate websites. To identify this subcategory, we use a search engine API to check whether the domain name has any URLs. This is similar to the method used to detect the web hosting sub-category described previously.

Dynamic DNS. This sub-category contains domain names used on dynamic DNS services. Dynamic DNS services allow Internet users to register subdomains under their specific domain names and resolve the names of subdomains and IP addresses. Dynamic DNS updates IP addresses that correspond to a domain name in real time (Vixie et al., 1997). Such dynamic DNS services were originally developed for legitimate users with dynamic IP addresses to access servers remotely. However, due to their low cost and high availability, attackers tend to abuse such services to conduct cyber attacks (Chen et al., 2014; Li et al., 2013; Nelms et al., 2013; Rahbarinia et al., 2016; 2015; Vadrevu et al., 2013). For example, attackers use such services to change the domain names used for C&C and modify the IP addresses corresponding to domain names to evade IP address blacklists or reputation systems. As mentioned previously, the dynamic DNS is used by legitimate users and abused by attackers. Thus, when considering defenses against malicious domain names that use the dynamic DNS, we should consider this characteristic and take care not to stop dynamic DNS services. To this end, we detect such dynamic DNS domain names based on pre-defined dynamic DNS domain name lists (DNS-BH - Malware Domain Blocklist, 0000).

Gratuitous. This sub-category contains domain names created by gratuitous domain-registration services (Fre, 0000), (Fre, 0000; Free DNS, 0000). For example, a gratuitously available domain name can be registered under some TLDs, such as `.tk`, `.ml`, and `.gq` or some e2LDs, such as `.flu.cc` and `.igg.biz`. Some gratuitous domain-registration services also offer gratuitous URL redirection services. Such services are easily abused by cyber attackers to create malicious domain names (Li et al., 2013; Rahbarinia et al., 2016; 2015). We should understand such service characteristics and consider countermeasures against cyber attacks. To identify gratuitous domain names, we take an approach similar to sinkholing and parking identification, i.e., we first collect corresponding name server records for each domain name then match those records with known name-server records used in gratuitous domain-registration services (DNS-BH - Malware Domain Blocklist, 0000; DNS-BH - Malware Domain Blocklist, 0000).

Domain hosting. Domain hosting is very similar to the previously described web hosting sub-category. In this paper, the difference between web hosting and domain hosting is the structure of domain names and URLs, as shown in Fig. 4. In a web-hosting case, an individual directory can be created (e.g.,/∼`user1`). On the other hand, in a domain-hosting case, subdomains or fully qualified domain names (FQDNs) can be created under the hosting service's domain names (e.g., `user1.example.org`). We should recognize these two patterns because we consider optimal countermeasures for each domain name to avoid filtering legitimate accesses. To

Table 1 – Corresponding categories in each step of DomainChroma.

Step 1: Categorization	Step 2: Separation	Step 3: Purification	Step 4: Renewal
Compromised: Advertising	HTTP-level	–	–
Compromised: CDN	HTTP-level	–	–
Compromised: Web hosting	HTTP-level	–	–
Dedicated: DGA	DNS-level	e2LD-level	Short-term
Dedicated: Re-registration	DNS-level	e2LD-level	Long-term
Dedicated: Sinkholing	DNS-level	e2LD-level	Long-term
Dedicated: Parking	DNS-level	e2LD-level	Long-term
Dedicated: Typosquatting	DNS-level	e2LD-level	Long-term
Dedicated: No-URLs	DNS-level	e2LD-level	Long-term
Dedicated: Dynamic DNS	DNS-level	FQDN-level	Short-term
Dedicated: Gratuitous	DNS-level	FQDN-level	Short-term
Dedicated: Domain hosting	DNS-level	FQDN-level	Long-term

classify domain hosting, we take an approach similar to the web hosting sub-category, i.e., we use known domain hosting patterns and a search engine API. When querying a search-engine API, we extract both URLs under a target domain name and subdomains under a e2LD of a target domain name to detect whether the target domain name is in a domain-hosting situation.

3.3. *Step 2: Separation of mixtures*

The second step of DomainChroma is separation of mixtures composed of malicious domain names. Once the input domain names are categorized in step 1, we can determine *where* the actions against each input domain name should be performed. We use a conditional procedure based on our pre-defined rules. These rules assign the points of defense to the corresponding categories, as shown in Table 1. Specifically, the input domain names are separated into two groups; one *requiring HTTP-level defenses* the other *requiring DNS-level defenses*. These groups correspond to the points of defense summarized in Section 2.2.

Domain names requiring HTTP-level defenses fall into the *compromised* category, namely, *advertising*, *CDN*, and *web hosting*, as defined in Section 3.2.1. These domain names are defined as compromised because they originally referred to legitimate services. Thus, within this group, our actions should not target domain names alone but should use additional information such as URLs pointing to specific malicious content or files.

Conversely, domain names requiring DNS-level defenses fall into the *dedicated* category, namely, *DGA*, *re-registration*, *sinkholing*, *parking*, *typosquatting*, *no-URLs*, *dynamic DNS*, *gratuitous*, and *domain hosting*, as defined in Section 3.2.2. These domain names are defined as dedicated because they are exclusively prepared for malicious purposes and can be directly targeted at DNS-level points of defense.

From the category/sub-category identification result of each input domain name and our rules defined in Table 1, we can decide *where* to apply defense solutions for the domain name. If a domain name falls into multiple sub-categories in both HTTP and DNS levels (e.g., web hosting and typosquatting), we assign it to the HTTP-level to reduce the risk of collateral damage caused by excessive filtering of legitimate communication. In an organization, the assignment will depend on the organization's operational policy. Our conditional procedure is easily tunable for this purpose.

3.4. *Step 3: Purification*

Step 3, *purification*, determines *how* each domain name should be used for the optimal defense. Similarly to step 2, we match defense strategies to categories through our pre-defined rules in Table 1. Specifically, we purify the domain names requiring DNS-level defenses or further separate them into two subgroups: those *requiring FQDN-level defenses* and those *requiring e2LD-level defenses*.

We stipulate that domain names in sub-categories *dynamic DNS*, *gratuitous*, and *domain hosting* require FQDN-level defenses, whereas those in sub-categories *DGA*, *re-registration*, *sinkholing*, *parking*, *typesquatting*, and *no-URLs* require e2LD-level defenses. Domain names in the former group are defined as requiring FQDN-level defenses because their hierarchical structure means that users can create subdomains or FQDNs under the e2LDs owned by the providers of the three sub-categories in this group, as explained in Section 3.2.2. Domain names in the latter group (which includes all other categories requiring DNS-level defenses in Step 2), are defined as requiring e2LD-level defenses because they are almost certainly registered with malice. Within the subgroup requiring e2LD-level defenses, we extract and process the e2LD parts of the input domain names, which can be more effectively filtered than the FQDNs. When a domain name matches multiple sub-categories in both FQDN and e2LD levels (e.g., dynamic DNS and DGA), we assign it to the FQDN-level to reduce the risk of collateral damage.

3.5. *Step 4: Renewal*

The fourth and final step of DomainChroma is *renewal*, whereas the previous two steps determine *where* and *how* we should apply defense solutions, respectively, this step determines *until when* the domain name information should be used for defense purposes. This step sets an *expiration date* for the defense information in each malicious domain name. The expiration date allows for changes in the malicious domain names and subsequent updates in the defense information.

As in steps 1, 2, and 3, category matching is done through a conditional procedure based on our pre-defined rules in Table 1. Specifically, the expiration dates of the DNS-level domain names identified in Step 1 are roughly classified as *short-term* and *long-term*.

Domain names in categories *DGA, dynamic DNS*, and *gratuitous* are assigned short-term expiration dates because attackers can systematically set up numerous distinct domain names in these categories at low cost, as explained in Section 3.2.2. For the DGA category, we only consider time-dependent DGAs, meaning that generated domain names are only valid during certain periods. For example, Conficker.C, which is one of the most famous DGAs, generates and expires 50,000 domain names per day (Plohmann et al., 2016). From the operational point of view, the defense information should consist of meaningful data and there is no need to keep invalid and expired data. On the other hand, domain names in the other categories requiring DNS-level defenses, namely, *re-registration, sinkholing, parking, typosquatting, no-URLs*, and *domain hosting*, are assigned long-term expiration dates. The recommended short-term expiration date set by the operators at each point of defense shown in Section 2.2 is less than one week. However, the desired short-term expiration date will depend on the operational situations or difficulties at each point of defense. In this case, we must classify the domain names requiring short-term updates to obtain effective defense solutions. When a domain name falls into multiple sub-categories in both short-term and long-term expiration dates (e.g., gratuitous and typosquatting), we assign it to the short-term expiration date to keep defense information meaningful.

3.6. Output: Practical defense information

DomainChroma outputs practical defense information from input malicious domain names for each point of defense defined in Section 2.2. Specifically, DomainChroma converts input malicious domain names into actionable threat intelligence so that the operators can see if it is safe to use each malicious domain name as defense information such as filtering blacklists or abuse reports for owners or administrators. Table 2 maps the defense information to the corresponding points of defense, as detailed in the following sections.

Blacklists for filtering. The defense information in blacklists filters the user's accesses to malicious domain names at both HTTP-level and DNS-level points of defense.

The HTTP-level points of defense include security appliances, web servers, and search engines, as explained in Section 2.2. Security appliances, which mainly monitor the HTTP protocol, can identify URLs requiring HTTP-level defenses. As explained in Section 3.3, such URLs are created using the additional information of specific malicious content or files. Blacklists for filtering should not be applied to web servers because the servers are the targets of filtering. The URLs referred by security appliances can also be referred by search engines, which can then filter any link pointing to malicious content.

DNS-level points of defense include caching name servers, authoritative name servers, and the domain registry/registrar, as explained in Section 2.2. At caching name servers in a local organization or ISP, blacklists can prevent users from accessing malicious domain names. The blacklists used in caching name servers include both FQDN and e2LD lists respectively corresponding to the domain names requiring FQDN-level and e2LD-level defenses output in Step 3 of DomainChroma. To minimize collateral damage of legitimate accesses, DomainChroma selects only the malicious domain names that are defendable at the DNS-level. Authoritative name servers can also filter answers to DNS queries pointing to blacklisted domain names, preventing their accesses by users. However, because such servers are deployed at each DNS zone, the efficiency of blacklisting is much lower for such servers than for caching name servers. At the domain registry/registrar, filtering blacklists are irrelevant because the domain name database is directly managed by the registry.

Abuse reports. Abuse reports provide additional defense information at both HTTP-level and DNS-level points of defense. Abuse reports request owners or administrators of web content or domain names to correct the problems caused by malicious content owned/managed by them. Generally, abuse reports can be sent by contacting the abuse contact extracted from the WHOIS data of the domain name or by requesting national or regional CERTs (Li et al., 2016a; Stock et al., 2016).

HTTP-level points of defense include security appliances, web servers, and search engines. Abuse reports are irrelevant to security appliances and search engines, which lack original content, but can be sent to content owners and server administrators of abused web servers requesting actions to remove the malicious content.

DNS-level points of defense include caching name servers, authoritative name servers, and domain registries and registrars. Abuse reports are irrelevant to caching name servers because these servers merely cache the DNS resolutions; they do not manage domain names. Authoritative name servers can update their zone data by referencing abuse reports; for example, they can delete the record of a malicious domain name. Although this action ensures obliteration of the specific domain name, the abuse reports must be sent to every authoritative name server corresponding to that domain name. For this reason, authoritative name servers are assessed as *partially effective* in Table 2. Domain registries/registrars can also reference abuse reports; therefore, they perform an anti-abuse action on the master database of each registry. For example, the master database can delete the domain name. As of February 2018, there are 1,235 domain registries (ICANN, 0000). Therefore, sending abuse reports to domain registries is relatively straightforward.

4. Evaluation

We implemented and evaluated DomainChroma on real datasets containing numerous domain names. The evaluation included the classification accuracy of each category/subcategory, output of DomainChroma, comparisons of collateral damage, validity of the expiration dates, and large-scale deployment.

Table 2 – Effectiveness of DomainChroma output at each point of defense.

		DNS-Level Defense			HTTP-Level Defense		
		Caching Name Server (Local or ISP)	Authoritative Name Server	Domain Registry and Registrar	Local Security Appliance	Web Server	Search Engine
Blacklists for Filtering	Domain List (e2LD)	●	◐	○	○	○	○
	Domain List (FQDN)	●	◐	○	○	○	○
	URL List	○	○	○	●	○	●
Abuse Reports for Owners or Admins	Domain List (e2LD)	○	◐	●	○	○	○
	Domain List (FQDN)	○	◐	○	○	○	○
	URL List	○	○	○	○	●	○

●: Effective, ◐: Partially Effective, ○: Not Effective

Table 3 – Dataset of malicious domain names.

Type	Period	# FQDNs
Honeyclient-Exploit	2015-03-01–2015-10-07	537
Honeyclient-Malware	2015-03-01–2015-10-07	68
Sandbox-Malware	2015-03-01–2015-10-07	775
Sandbox-C&C	2015-03-01–2015-10-07	8473
Pro-C&C	2015-03-01–2015-03-29	97
Pro-Phishing	2015-03-01–2015-03-29	78,221
Total		88,171

4.1. *Dataset*

Before evaluating DomainChroma, we prepared three types of datasets: an input dataset of malicious domain names, datasets related to these malicious domain names, and a dataset of domain categories/sub-categories.

The dataset of malicious domain names is presented in Table 3. As explained in Section 3.1, the expected input to DomainChroma is a dataset of malicious domain names provided by domain-reputation systems or any domain name blacklist. We used the six types of malicious domain names listed in Table 3. Note that benign or legitimate domain names were not included in the dataset for our evaluation because simply distinguishing malicious domain names from a mixture of malicious and *legitimate* domain names is out of the scope of our paper. We focused on deciding *actions* based on already detected malicious domain names. The malicious domain names were composed of *truly* malicious domain names confirmed by a client-based honeypot (honeyclient), sandbox system, and commercial and professional services provided by a security vendor. The *Honeyclient-Exploit* and *Honeyclient-Malware* types contained malicious domain names related to drive-by download attacks detected by our honeyclient, which regularly crawled public blacklists (hpH, 0000; Mal, 0000), some commercial blacklists, and legitimate websites (Ale, 0000; DMO, 0000) from March 2015 to October 2015. More precisely, the malicious domain names collected in Honeyclient-Exploit were distributing exploit content during drive-by download attacks. Honeyclient-Malware was composed of malicious domain names responsible for distributing malware samples. The malicious domain names in *Sandbox-Malware* and *Sandbox-C&C* were observed in a sandbox system running malware samples. These samples were randomly downloaded from VirusTotal (Vir, 0000) daily and consisted of newly submitted (within 24 h) malicious executable files used in Microsoft Windows. Specifically, the malicious domain names in Sandbox-Malware were connected by malware downloader samples, enabling the download of other malware samples. Sandbox-C&C contained the C&C servers' domain names detected in the sandbox. The malicious domain names in *Pro-C&C* and *Pro-Phishing* were captured from C&C and phishing websites in March 2015 by a commercial and professional security service.

The second dataset included datasets related to the malicious domain names in the first dataset. Specifically, we used the passive DNS database (DNSDB) (Farsight Security, Inc., 0000) corresponding to the domain names listed in Table 3. The DNSDB stored the logs collected from a large set of caching name servers operated by multiple organizations and enabled us to further investigate the domain name usage, such as the first- and last-seen timestamps, a list of resolved IP addresses, and changes in the name server records. We also crawled and parsed the WHOIS data corresponding to the domain names listed in Table 3. As explained in Section 3.2.2, we extracted the WHOIS server, name server, creation date,

Table 4 – Categorized domain names.

Category	Data source	#Rules/Patterns	#Matched FQDNs
Compromised: Advertising	hpHosts (hpH, 0000) & EasyList (Eas, 0000)	–	324
Compromised: CDN	DNSDB (Farsight Security, Inc., 0000) (A/CNAME)	118 patterns	19
Compromised: Web hosting	Search engine results	–	23,804
Dedicated: DGA	DGArchive (Plohmann et al., 2016)	28 DGA families	12,895
Dedicated: Re-registration	WHOIS Results	–	64
Dedicated: Sinkholing	DNSDB (Farsight Security, Inc., 0000) (NS)	19 NS records	78
Dedicated: Parking	DNSDB (Farsight Security, Inc., 0000) (NS)	16 NS records	1379
Dedicated: Typosquatting	DNSDB (Farsight Security, Inc., 0000) (A/NS) & WHOIS	5 typo models	91
Dedicated: No-URLs	Search engine results	–	64,367
Dedicated: Dynamic DNS	Dynamic DNS lists (DNS-BH - Malware Domain Blocklist, 0000)	743 patterns	60
Dedicated: Gratuitous	DNSDB (Farsight Security, Inc., 0000) (NS)	4 NS records	138
Dedicated: Domain hosting	Search engine results	–	33,388

update date, and expiration date of the domain names. Moreover, the URLs under each domain name in Table 3 were extracted and crawled using an external search engine (Microsoft Corporation, 0000).

The third dataset contained the domain name categories/sub-categories defined in Section 3.2. To re-implement and evaluate the category-classification technique, we included domain names with labeled categories in the category dataset. Our category-labeled domain names are summarized in Table 4 and described in detail below. The advertising domain names were sourced from the pre-defined advertising services' domain list (Eas, 0000; hpH, 0000). The CDN domain names were detected by applying the detection rules or patterns described in Section 3.2.1. The web-hosting domain names in the search-engine results were heuristically decided from their URL structures and content. For the DGA sub-category, we leveraged the domain names created by 28 distinct DGA families, which were kindly provided by the DGArchive project (DGA, 0000). Note that we only used the time-dependent DGA family, as explained in Section 3.5. The re-registration domain names were detected in our WHOIS data and DNSDB results by our heuristic rules presented in Section 3.2.2. Sinkholing and parking domain names were identified in the DNSDB results by their known specific name server records, as explained in Section 3.2.2. The typosquatting domain names were detected in the DNSDB results by using the machine-learning algorithm presented in Section 3.2.2. Domain names in the no-URLs sub-category were detected in the above search-engine results. Dynamic DNS and gratuitous domain names were identified by their pre-defined patterns in the DNSDB results, as shown in Section 3.2.2. Finally, the domain names in the domain-hosting sub-category were heuristically detected in the search-engine results, as explained in Section 3.2.2.

4.2. *Classification accuracy of each category/sub-category*

Before adopting the category classifiers in DomainChroma, we first evaluated the classification accuracy of each category/sub-category using the third dataset introduced in Section 4.1. As an accuracy measure, we computed the number of falsely classified domain names. The advertising sub-category contained only advertising and tracking domain names, and all domain names were correctly matched. The domain names in the CDN, sinkholing, parking, dynamic DNS, and gratuitous sub-categories were determined by reliable patterns or rules. We manually confirmed that all domain names in each of these sub-categories were correctly classified. In the re-registration and typosquatting sub-categories, the absence of false matches was confirmed by manually inspecting the WHOIS data corresponding to the domain names in each sub-category. The domain names in the web-hosting, domain-hosting, and no-URLs sub-categories relied on their URL and domain structures in reliable search-engine results. We confirmed that no URLs or domain names were falsely detected by these processes.

The domain names in the DGA sub-category were selected by the machine-learning classifier introduced in Section 3.2.2. To evaluate this classifier, we applied 10-fold cross-validation on the training data. The training data included a DGA dataset compiled from some of the DGArchive (DGA, 0000) domain names, and a non-DGA dataset containing legitimate domain names extracted from Alexa Topsites (Ale, 0000). The evaluation criteria were precision and recall. Precision defines the number ratio of the actual DGA domain names to the domain names detected as DGA by the machine-learning classifier. The recall is the number ratio of the correctly detected DGA domain names to the actual DGA domain names. The precision and recall in the cross validation were 0.971 and 0.964, respectively, confirming the high accuracy of our machine-learning classifier.

4.3. *Output of DomainChroma*

The six types of malicious domain names shown in Table 3 were input to DomainChroma, and the outputs were evaluated. This evaluation revealed the relationships between our defined domain-name categories/sub-categories and the various attack types (e.g., drive-by download, malware download and C&C, and phishing).

Table 5 lists the domain names in *each* category detected with our individual classifiers. Note that some of the domain names belong to multiple categories. We now present some noteworthy cases from the table. Advertising is most commonly used in Pro-Phishing and Sandbox–C&C. We confirmed that advertising with gratuitous web hosting and tracking

Table 5 – Individual category-detection results.

Dataset	#Advertising	#CDN	#Web hosting	#DGA	#Re-registration	#Sinkholing	#Parking	#Typosquatting	#No-URLs	#Dynamic DNS	#Gratuitous	#Domain hosting
Honeyclient-Exploit	0	0	16	54	0	0	3	0	521	8	36	132
Honeyclient-Malware	0	0	2	0	0	0	2	0	66	0	0	9
Sandbox-Malware	3	18	86	45	0	0	82	0	689	1	1	208
Sandbox-C&C	55	0	1895	1619	4	72	600	20	6578	14	55	3382
Pro-C&C	2	0	28	10	0	6	12	0	69	1	4	41
Pro-Phishing	264	1	21,777	11,167	60	0	680	71	56,444	36	42	29,616
Total	324	19	23,804	12,895	64	78	1379	91	64,367	60	138	33,388

Table 6 – Final category-detection results by DomainChroma.

Dataset	#Advertising	#CDN	#Web hosting	#DGA	#Re-registration	#Sinkholing	#Parking	#Typosquatting	#No-URLs	#Dynamic DNS	#Gratuitous	#Domain hosting	#Total FQDNs
Honeyclient-Exploit	0	0	16	29	0	0	3	0	337	8	36	108	537
Honeyclient-Malware	0	0	2	0	0	0	2	0	57	0	0	7	68
Sandbox-Malware	3	17	86	31	0	0	77	0	441	1	1	118	775
Sandbox-C&C	55	0	1884	1299	4	20	424	13	3255	14	51	1454	8473
Pro-C&C	2	0	28	7	0	2	6	0	37	1	4	10	97
Pro-Phishing	264	1	21,766	9036	37	0	527	21	38,968	29	35	7537	78,221
Total	324	18	23,782	10,402	41	22	1039	34	43,095	53	127	9234	88,171

Table 7 – Summary of output of DomainChroma.

Dataset	# FQDNs [DNS-level] [e2LD-level]	# FQDNs [DNS-level] [FQDN-level]	# FQDNs [HTTP-level]	# Total input FQDNs
Honeyclient-Exploit	369 (68.7%)	152 (28.3%)	16 (3.0%)	537
Honeyclient-Malware	59 (86.8%)	7 (10.3%)	2 (2.9%)	68
Sandbox-Malware	549 (70.8%)	120 (15.5%)	106 (13.7%)	775
Sandbox-C&C	5015 (59.2%)	1519 (17.9%)	1939 (22.9%)	8473
Pro-C&C	52 (53.6%)	15 (15.5%)	30 (30.9%)	97
Pro-Phishing	48,589 (62.1%)	7601 (9.7%)	22,031 (28.2%)	78,221
Total	54,633 (62.0%)	9414 (10.7%)	24,124 (27.4%)	88,171

services are exploited in phishing websites. The CDN is largely used in Sandbox-Malware and provides a distribution platform for malicious file objects. Web-hosting services provide flexible and reliable cloud services; thus, are exploited by all types of attackers. A DGA is generally thought to be used in C&C, but we confirmed the presence of JavaScript-based DGAs in websites created by exploit kits. We also confirmed DGA-specific characteristics in phishing sites with continuously generated domain names. Some of the re-registered phishing domain names were originally linked to legitimate services. The C&C domain names tend to be targeted in the sinkholing and botnet takedown operations of security organizations (Nadji et al., 2013a; 2013b). Therefore, sinkholing is used only in C&C. Parking, which provides an income source, is used in all types of attacks. Typosquatting is used in Sandbox-C&C and Pro-Phishing; a domain name similar to a legitimate service and domain names targeting typographic errors of famous online payment services were observed in C&C. The no-URLs category includes many domain names in all types of attacks. The prevalence of attacks in this category can be explained by the domain names that do not use URLs or HTTP/HTTPS, and by the URLs that cannot be crawled by search-engine providers. For example, phishing URLs can be spread only by spam email that cannot be reached by search engines. For no specific reason, domain names in the dynamic DNS and gratuitous categories are used in all types of attacks, except Honeyclient-Malware. Domain-hosting services are also used in all types of attacks for the reasons described under web hosting.

Table 6 shows the final assigned categories by DomainChroma. As explained in Sections 3.3–3.5, DomainChroma prioritizes *HTTP-level*, *FQDN-level*, and *Short-term* to reduce the risk of collateral damage and to keep defense information meaningful. Thus, contrary to Table 5, every domain name is assigned to one category in Table 6.

Table 7 summarizes the output of DomainChroma; namely, the identified points of defense and levels of defense information of the input domain names. The points of defense are HTTP-level (security appliances, web servers, and search engines) and DNS-level (caching name servers, authoritative name servers, and domain registries/registrars), as explained in Section 2.2. The level of defense information is the required granularity level (e2LD-level or FQDN-level). From the DomainChroma output, we can provide blacklists to points of defense and send abuse reports to corresponding organizations in real operations. As mentioned in Section 3.3, if a domain name belongs to multiple sub-categories in both HTTP and DNS levels, DomainChroma selects the HTTP-level to reduce the risk of collateral damage of legitimate accesses. Among the FQDNs, 24,124 (27.4% of the input FQDNs) required HTTP-level defenses, 54,633 (62.0% of the input FQDNs) required DNS-level defenses with e2LD-level domain information, and 9414 (10.7% of the input FQDNs) required DNS-level defenses with FQDN-level domain information. These results indicate that when applying the output of DomainChroma, over 70% of the domain names engaged in various types of cyber attacks could be effectively defended only at DNS-level points of defense. This result is surprising. In the Honeyclient-Exploit and Honeyclient-Malware cases, which correspond to drive-by download attacks, only approximately 3% of the input FQDNs required HTTP-level defenses, although both types were web-based attacks targeting web browsers and their plugins. This finding can possibly be explained by the activities of recent attackers who tend to prepare new dedicated domain names for hosting their exploit kits and malware samples in drive-by download attacks. In the malware activities Sandbox-Malware, Sandbox-C&C, and Pro-C&C, up to 86% of the input FQDNs required DNS-level defenses. In the pro-phishing case, which corresponds to web-phishing attacks, around 70% of the input FQDNs required DNS-level defenses, and only 9.7% required FQDN-level defense information. The low percentage of inputs requiring FQDN-level information is attributed to the self-registration and self-use of dedicated e2LDs in most recent phishing sites.

4.4. *Comparison of collateral damage*

This evaluation investigated the risk of collateral damage when applying the DNS-level blacklist output from DomainChroma. The collateral damage was defined as the number of distinct legitimate domain names or URLs that would be falsely filtered by the DNS-level blacklists. The evaluation proceeded in two steps: generating the blacklists and investigating their collateral damage.

First, we generated DNS-level blacklists for filtering at the e2LD and FQDN levels by using DomainChroma. As described in the previous section, we input the domain names in Table 3 to DomainChroma. The blacklists were created on October 8, 2015, which succeeded the collection period of the input domain names. For comparison, we created two types of additional blacklists: *DomainAll (e2LD-level)* and *DomainAll (FQDN-level)*. The DomainAll (e2LD-level) blacklist

Table 8 – Collateral-damage evaluation.

	DomainChroma	DomainAll (e2LD-level)	DomainAll (FQDN-level)
# Blacklists (e2LD-level)	28,840	56,406	0
# Blacklists (FQDN-level)	9414	0	88,171
# Collateral Damaged Domains	0	333,665	30,966
# Collateral Damaged URLs	0	1,331,808	613,631

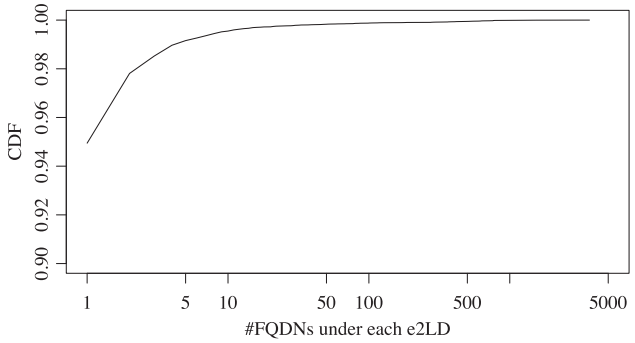

Fig. 5 – CDF of the number of FQDNs under each blacklisted e2LD in DomainChroma.

contained all distinct e2LDs (e.g., `example.com`) directly extracted from the input FQDNs (e.g., www.example.com). The DomainAll (FQDN-level) blacklist contained all input FQDNs, meaning that no domain names were excluded from the input. Table 8 shows the number of domain names in each of the above-described blacklists. Note that DomainChroma contains fewer blacklisted domain names (e2LD-level) than stated in Table 7 because the table includes duplications at the e2LD-level. Our analysis regarding the duplication and its distribution shows that the mean number of FQDNs under each blacklisted e2LD is 1.89 and the max number of those is 3678. Fig. 5 provides a log-scale CDF of the number of FQDNs under each blacklisted e2LD in DomainChroma. We find that 94.9% of blacklisted e2LDs have only one FQDN under each e2LD and 99.2% of them have less than or equal to five FQDNs under each 2LD, showing that a small number of e2LDs aggregated many FQDNs. We confirmed that such e2LDs aggregated many FQDNs related to similar phishing websites, malware distribution websites, and C&C.

Second, we investigated the collateral damage incurred by the three blacklists. To this end, we leveraged the DNSDB and search-engine results, as described in Section 4.1. For a fair evaluation, we used only the DNSDB and search-engine information obtained after the creation date (October 8, 2015) of the blacklists. Specifically, we extracted the legitimate domain names under each blacklisted domain name from the DNSDB results and the legitimate URLs containing each blacklisted domain name from the search-engine results. If these legitimate domain names and URLs are filtered by a blacklisted domain name, collateral damage will occur. In the above investigations, we identified and filtered out the malicious domain names and URLs in the DNSDB and search-engine results, respectively, by referring to multiple public and commercial blacklists including VirusTotal (Vir, 0000) and Google Safe Browsing (Goo, 0000) dated to November 2016. Note that there is the possibility of missed malicious domain names and URLs by these blacklists due to their coverage. However, we manually sampled and examined that the remaining legitimate domain names and URLs were truly legitimate at that time. Table 8 summarizes the number of collaterally damaged domain names and URLs after applying the above three blacklists. The blacklists generated by DomainChroma incurred no collateral damage; that is, no legitimate domain names or URLs were falsely filtered by DomainChroma. Recall that DomainChroma was designed to prevent such situations by analyzing the characteristics of the domain names and points of defense. On the other hand, the DomainAll (e2LD-level) and DomainAll (FQDN-level) blacklists incurred huge collateral damage because these blacklists contained many domain names that require different levels of defenses. Specifically, DomainAll (e2LD-level) contained domain names that require FQDN-level and HTTP-level defenses, and DomainAll (FQDN-level) contained domain names that require HTTP-level defenses. These results confirmed a significant gap between the simple detection of malicious domain names and their use in defense solutions. To determine the defense solutions, we require the characteristics of each malicious domain name. These results also indicate that, by combining various techniques and considering both defense solutions and points of defense, DomainChroma generates optimal blacklists that do not inconvenience legitimate users.

4.5. *Validity of the renewal settings*

This section validates the renewal settings determined in Step 4 of DomainChroma presented in Section 3.5. To this end, we re-leveraged the DNSDB results from a different perspective. Specifically, we analyzed the lifespans of the malicious domain names shown in Table 3, where the lifespan defines the period from the first-seen to the last-seen timestamp. The lifespan is equivalent to the period in which the domain name has at least one corresponding IP address. For this purpose, we applied a survival-analysis method based on the Kaplan–Meier estimator (Kaplan and Meier, 1958), which is widely used for lifespan estimation in this field (Gañán et al., 2015; Lauinger et al., 2016; Noroozian et al., 2016). Fig. 6 summarizes the results of the survival analysis. The survival probability means that the domain name retains at least one IP address after the elapsed number of days. The dynamic DNS (DDNS), DGA, and gratuitous categories, for which we set short-term expiration dates, exhibited much lower survival probabilities (lifespans) than the other DNS-level categories and HTTP-level

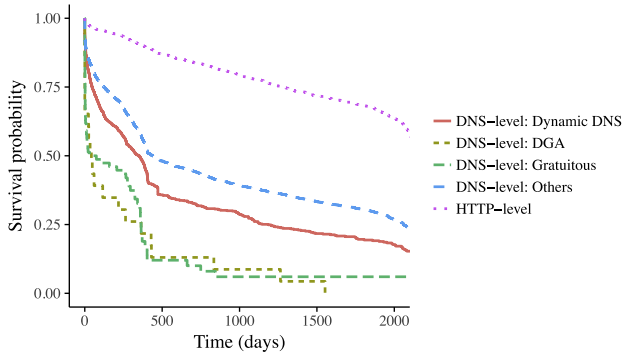

Fig. 6 – Survival analysis of domain names in each category (*x*-axis (time) is number of days elapsed since first-seen time stamp.)

categories (advertising, CDN, and web hosting), for which we set long-term expiration dates (cf. Table 1). This result is unsurprising because Step 4 of DomainChroma captures the domain names with short lifespans, as stated in Section 3.5. Further lifespan analysis revealed that over 50% of the domain names in the dynamic DNS and gratuitous categories existed for less than 30 days. This evaluation confirms that DomainChroma appropriately determines the renewal or expiration date of each domain name.

4.6. *Large-scale deployment*

So far we have evaluated the basic usefulness of DomainChroma. This section evaluates its capabilities in a large-scale deployment. Specifically, we newly subscribed to 25 types of commercial and public blacklists, including over 1.6 million domain names, to evaluate DomainChroma in terms of immediate applicability to real-world, large-scale, and state-of-the-art blacklists.

4.6.1. *Subscribed blacklists*

The multiple blacklists used in this evaluation are shown in Table 9. The blacklists labeled *Spamhaus* were provided by the Spamhaus project (The Spamhaus Project Ltd., 0000) in November 2017. *Spamhaus-Spam* contained spam-related domain names (e.g., spam payload, spam sources, and spam senders). *Spamhaus-Phish* consisted of phishing domain names. *Spamhaus-Malware* listed malware-related domain names such as malware distribution websites. *Spamhaus-C&C* was composed of malicious domain names used for botnet C&C. *Spamhaus-AbusedSpam* included abused spam domain names that are abused by spammers through cyber attacks such as exploits and hacking. *Spamhaus-AbusedRedirector* contained abused spammed redirector domain names. *Spamhaus-AbusedPhish*, *Spamhaus-AbusedMalware*, and *Spamhaus-AbusedC&C* consisted of abused phishing, malware, and C&C domain names, respectively. The blacklists labeled *Uribl* were provided by URIBL (URI, 0000) in November 2017. The *Uribl-Black* contained malicious domain names belonging to or used by spammers. The *Uribl-Grey*, *Uribl-Red*, and *Uribl-Gold* consisted of less malicious domain names than *Uribl-Black* since URIBL used more experimental methods to detect them (URI, 0000). The blacklists labeled *Hphosts* were obtained from hpHosts (hpH, 0000) in October 2017. The *Hphosts-EMD* was composed of malicious websites engaged in malware distribution. The *Hphosts-FSA* listed malicious websites engaged in selling/distribution of bogus or fraudulent applications/services. The *Hphosts-PHA* included malicious websites of illegal pharmacy activities. The *Hphosts-PSH* listed phishing websites. The *Hphosts-PUP* consisted of malicious websites engaged in distributing potentially unwanted programs (PUPs). The *Hphosts-WRZ* included malicious websites engaged in selling and distributing cracked software called warez. *Phishtank* and *Openphish* consisted of phishing sites obtained from PhishTank (Phi, 0000) and OpenPhish (Ope, 0000) in October 2017, respectively. *Malwaredomains* were obtained from DNS-BH (Mal, 0000) and contained malicious domain names that were known to be used to distribute malware and spyware in October 2017. *Ransomware* consisted of malicious domain names used for ransomware botnet C&C provided by Ransomware Tracker (Ran, 0000) in October 2017. *Sans* was composed of suspicious domain names provided by SANS Internet Storm Center (SANS Internet Storm Center, 0000) in October 2017. *Zeus* included zeus botnet C&C provided by Zeus Tracker (Zeu, 0000) in October 2017.

We analyzed the overlap between the subscribed blacklists, showing that only 2.4% of domain names were overlapped in total. Table 10 shows the detailed overlap result. Note that we aggregated blacklists provided by the same provider, namely Spamhaus, Uribl, and Hphosts, since there were no overlaps in them. The largest overlap (30,872 FQDNs) was between Spamhaus and Uribl, the second largest one (8019 FQDNs) was between Hphosts and Malwaredomains, the third largest

Table 9 – Input blacklists in large-scale deployment.

Blacklist	Date	# FQDNs
Spamhaus-Spam	2017-11-10	466,235
Spamhaus-Phish	2017-11-10	5690
Spamhaus-Malware	2017-11-10	681
Spamhaus-C&C	2017-11-10	51,775
Spamhaus-AbusedSpam	2017-11-10	1222
Spamhaus-AbusedRedirector	2017-11-10	93
Spamhaus-AbusedPhish	2017-11-10	81
Spamhaus-AbusedMalware	2017-11-10	120
Spamhaus-AbusedC&C	2017-11-10	4434
Uribl-Black	2017-11-3	97,683
Uribl-Grey	2017-11-3	194
Uribl-Red	2017-11-3	1228
Uribl-Gold	2017-11-3	199,908
Hphosts-EMD	2017-10-29	185,475
Hphosts-FSA	2017-10-29	294,233
Hphosts-PHA	2017-10-29	34,870
Hphosts-PSH	2017-10-29	224,511
Hphosts-PUP	2017-10-28	13,127
Hphosts-WRZ	2017-10-19	3588
Phishtank	2017-10-30	11,841
Openphish	2017-10-30	2529
Malwaredomains	2017-10-27	21,130
Ransomware	2017-10-30	1883
Sans	2017-10-30	3464
Zeus	2017-10-30	55
Total		1,626,050

Table 10 – Overlap between subscribed blacklists.

	Spamhaus	Uribl	Hphosts	Phishtank	OpenPhish	Malwaredomains	Ransomware	Sans	Zeus
Spamhaus	–	30,872	5025	43	11	414	30	55	2
Uribl	30,872	–	2200	5	5	101	0	0	0
Hphosts	5025	2200	–	4947	1655	8019	280	480	30
Phishtank	43	5	4947	–	728	648	0	0	0
Openphish	11	5	1655	728	–	488	0	0	0
Malwaredomains	414	101	8019	648	488	–	24	40	3
Ransomware	30	0	280	0	0	24	–	1881	0
Sans	55	0	480	0	0	40	1881	–	47
Zeus	2	0	30	0	0	3	0	47	–

one (5025 FQDNs) was between Spamhaus and Hphosts. Most of the overlaps were domain names created by DGAs. These providers were assumed to blacklist the same DGAs since some DGAs were already reverse-engineered (Plohmann et al., 2016).

4.6.2. *Categories assigned*

Table 11 lists the final assigned categories by DomainChroma in the large-scale deployment. Every domain name is assigned to one category based on our defined priorities explained in Sections 3.3–3.5. We now present some noteworthy cases from the table. The advertising category is most commonly assigned to the Phishtank blacklist. Some phishing websites in Phishtank exploited advertising with gratuitous web hosting and tracking services. The CDN category is mostly matched in Hphosts-EMD and Hphosts-FSA and used as a platform to distribute malware or fraudulent applications. Web-hosting services generally provide flexible and reliable services; thus, are heavily used by all types of blacklists. For DGAs, we confirmed the C&C AGD in Spamhaus-C&C and DGA-like continuously generated disposable domain names mainly in Spamhaus-Spam, Uribl-Gold, and Hphosts-FSA. Some re-registered domain names, which are originally used for legitimate services, are found mostly in Uribl-Gold and Spamhaus-Spam. The sinkholing category is assigned to some blacklisted domain names in Spamhaus-Spam and Spamhaus-C&C. The parking category is largely found in Spamhaus-Spam. Typosquatting is heavily used in Spamhaus-Spam and Hphosts-EMD to disguise the domain name as a legitimate service to deceive users into clicking the link in spam message or malicious websites. The no-URLs category includes many domain names in all types of blacklists because there are domain names that do not use public URLs reached by search-engine crawlers. The dynamic DNS category is mostly used in Hphosts-EMD to create short-lived disposable domain names to distribute malicious files. Similarly to dynamic DNS, domain names in the gratuitous category are detected in Spamhaus-Spam and Hphosts-EMD. For no specific reason, domain names in the domain hosting category are used in all types of blacklists.

4.6.3. *Output of DomainChroma*

Table 12 summarizes the output of DomainChroma in the large-scale deployment; namely, the identified points of defense and levels of defense information of the input 25 types of subscribed blacklists. The points of defense are HTTP-level and DNS-level, as explained in Section 2.2. The level of defense information is the required granularity level (e2LD-level or FQDN-level). In total, 204,098 (12.6% of the input FQDNs) required HTTP-level defenses, 918,475 (56.5% of the input FQDNs) required DNS-level defenses with e2LD-level domain information, and 503,477 (31.0% of the input FQDNs) required DNS-level defenses with FQDN-level domain information. These results indicate that 87.5% of the domain names extracted from various types of blacklists could be effectively defended only at DNS-level points of defense. However, there are still domain names requiring HTTP-level defenses detected by DomainChroma, meaning that simple use of such blacklists is not an optimized defense solution and causes collateral damage.

Now we explore the result in each blacklist shown in Table 12 in terms of the numbers of domain names requiring HTTP-level defenses. The results in blacklists labeled *Spamhaus* were divided into two groups. One group consisted of four Spamhaus blacklists: *Spamhaus-Spam, Spamhaus-Phish, Spamhaus-Malware*, and *Spamhaus-C&C*. The percentages of domain names requiring HTTP-level defenses in this group are relatively low and range from 0.5 to 15.0%. The other group consisted of the other five Spamhaus blacklists: *Spamhaus-AbusedSpam, Spamhaus-AbusedRedirector, Spamhaus-AbusedPhish, Spamhaus-AbusedMalware*, and *Spamhaus-AbusedC&C*. The percentages of those are very high and range from 57.5 to 75.7%. Even though Spamhaus DBL (The Spamhaus Project Ltd., 0000) is originally intended to use as DNS blacklists, the quality of blacklisted domain names varies by each blacklist. For blacklists labeled *Uribl*, the percentages of domain names requiring HTTP-level defenses also varies greatly: 7.5% in *Uribl-Black*, 68.6% in *Uribl-Grey*, 42.9% in *Uribl-Red*, and 7.4% in *Uribl-Gold*. In blacklists labeled *Hphosts*, the percentages of HTTP-level defenses range from 10.1 to 25.8%. Specifically, blacklists for web-based threats, such as *Hphosts-PSH* and *Hphosts-WRZ*, require HTTP-level defense more than other Hphosts blacklists. For blacklists intended to block phishing attacks, namely *Phishtank* and *Openphish*, over 66% of blacklisted domain names require HTTP-level defenses. In *Malwaredomains*, 29.8% of blacklisted domain names require HTTP-level defenses. Only 0.1% of domain names require HTTP-level defenses in the *Ransomware* blacklist. For the *Sans* blacklist, we detected that 6.5% of domain names require HTTP-level defenses. In the *Zeus*

Table 11 – Category-detection results in large-scale deployment.

Blacklist	#Advertising	#CDN	#Web hosting	#DGA	#Re-registration	#Sinkholing	#Parking	#Typosquatting	#No-URLs	#Dynamic DNS	#Gratuitous	#Domain hosting	#Total FQDNs
Spamhaus-Spam	13	0	40,528	76,266	42	199	95,165	441	208,866	0	8873	35,842	466,235
Spamhaus-Phish	1	0	852	753	2	0	108	2	2199	1	729	1043	5690
Spamhaus-Malware	1	0	35	64	0	7	2	0	224	2	6	340	681
Spamhaus-C&C	1	0	271	44,088	0	386	48	4	4895	0	77	2005	51,775
Spamhaus-AbusedSpam	0	0	703	71	0	0	0	2	423	0	7	16	1222
Spamhaus-AbusedRedirector	2	0	64	0	0	0	3	1	19	0	0	4	93
Spamhaus-AbusedPhish	0	0	59	8	0	0	0	0	8	1	0	5	81
Spamhaus-AbusedMalware	0	0	77	8	0	0	0	0	25	0	0	10	120
Spamhaus-AbusedC&C	0	0	3357	289	0	0	1	8	727	0	3	49	4434
Uribl-Black	8	0	7294	14,999	15	0	402	40	45,734	2	4419	24,770	97,683
Uribl-Grey	15	0	118	7	0	0	4	0	28	0	0	22	194
Uribl-Red	2	0	525	74	0	0	7	2	392	0	7	219	1228
Uribl-Gold	5	0	14,731	48,742	72	6	1318	69	43,287	0	32	91,646	199,908
Hphosts-EMD	0	62	25,306	26,943	25	91	5577	275	42,945	2638	7222	74,391	185,475
Hphosts-FSA	0	53	39,921	39,904	31	10	6730	108	93,627	80	1934	111,835	294,233
Hphosts-PHA	0	0	5940	4640	18	0	286	2	6248	57	221	17,458	34,870
Hphosts-PSH	0	3	45,730	14,083	4	2	5909	112	52,245	115	2831	103,477	224,511
Hphosts-PUP	0	20	1304	3815	0	0	26	1	7451	0	16	494	13,127
Hphosts-WRZ	0	0	924	105	14	0	284	5	769	3	8	1476	3588
Phishtank	25	4	7944	528	13	0	102	5	2952	2	125	141	11,841
Openphish	1	0	1669	119	0	0	1	1	546	1	70	121	2529
Malwaredomains	5	0	6290	2123	7	11	601	11	5153	37	869	6023	21,130
Ransomware	0	0	2	1231	0	3	1	0	75	0	0	571	1883
Sans	0	0	224	1481	0	5	9	0	624	35	51	1035	3464
Zeus	0	0	9	3	0	0	1	0	32	0	7	3	55
Total	79	142	203,877	280,344	243	720	116,585	1089	519,494	2974	27,507	472,996	1,626,050

Table 12 – Output of DomainChroma in large-scale deployment.

Blacklist	# FQDNs [DNS-level] [e2LD-level]	# FQDNs [DNS-level] [FQDN-level]	# FQDNs [HTTP-level]	# Total FQDNs
Spamhaus-Spam	380,979 (81.7%)	44,715 (9.6%)	40,541 (8.7%)	466,235
Spamhaus-Phish	3064 (53.8%)	1773 (31.2%)	853 (15.0%)	5690
Spamhaus-Malware	297 (43.6%)	348 (51.1%)	36 (5.3%)	681
Spamhaus-C&C	49,421 (95.5%)	2082 (4.0%)	272 (0.5%)	51,775
Spamhaus-AbusedSpam	496 (40.6%)	23 (1.9%)	703 (57.5%)	1222
Spamhaus-AbusedRedirector	23 (24.7%)	4 (4.3%)	66 (71.0%)	93
Spamhaus-AbusedPhish	16 (19.8%)	6 (7.4%)	59 (72.8%)	81
Spamhaus-AbusedMalware	33 (27.5%)	10 (8.3%)	77 (64.2%)	120
Spamhaus-AbusedC&C	1025 (23.1%)	52 (1.2%)	3357 (75.7%)	4434
Uribl-Black	61,190 (62.6%)	29,191 (29.9%)	7302 (7.5%)	97,683
Uribl-Grey	39 (20.1%)	22 (11.3%)	133 (68.6%)	194
Uribl-Red	475 (38.7%)	226 (18.4%)	527 (42.9%)	1228
Uribl-Gold	93,494 (46.8%)	91,678 (45.9%)	14,736 (7.4%)	199,908
Hphosts-EMD	75,856 (40.9%)	84,251 (45.4%)	25,368 (13.7%)	185,475
Hphosts-FSA	140,410 (47.7%)	113,849 (38.7%)	39,974 (13.6%)	294,233
Hphosts-PHA	11,194 (32.1%)	17,736 (50.9%)	5940 (17.0%)	34,870
Hphosts-PSH	72,355 (32.2%)	106,423 (47.4%)	45,733 (20.4%)	224,511
Hphosts-PUP	11,293 (86.0%)	510 (3.9%)	1324 (10.1%)	13,127
Hphosts-WRZ	1177 (32.8%)	1487 (41.4%)	924 (25.8%)	3588
Phishtank	3600 (30.4%)	268 (2.3%)	7973 (67.3%)	11,841
Openphish	667 (26.4%)	192 (7.6%)	1670 (66.0%)	2529
Malwaredomains	7906 (37.4%)	6929 (32.8%)	6295 (29.8%)	21,130
Ransomware	1310 (69.6%)	571 (30.3%)	2 (0.1%)	1883
Sans	2119 (61.2%)	1121 (32.4%)	224 (6.5%)	3464
Zeus	36 (65.5%)	10 (18.2%)	9 (16.4%)	55
Total	918,475 (56.5%)	503,477 (31.0%)	204,098 (12.6%)	1,626,050

blacklist, 16.4% of domain names require HTTP-level defenses and cause collateral damage.

In summary, there are many domain name blacklists against various threats. However, the above results indicate that the quality varies by each blacklist or its provider. DomainChroma can be used to analyze such quality differences between multiple blacklists and decide the best defense solution for each malicious domain name to reduce the risk of collateral damage.

4.6.4. *Generating optimal DNS-level blacklists*

We generated DNS-level blacklists for filtering at the e2LD and FQDN levels by using DomainChroma's output discussed in Section 4.6.3. Specifically, we input the blacklists shown in Table 9 to DomainChroma. Table 13 lists the number of blacklisted domain names requiring e2LD-level defenses, FQDN-level defenses, and both e2LD and FQDN levels. A total of 1,283,178 (78.9% of the input FQDNs from various blacklists) were extracted as DNS-level blacklists from multiple blacklists. That is, DomainChroma enables us to aggregate the FQDNs that require e2LD-level defenses and exclude those that cause collateral damage.

Since the generated DNS-level blacklist is based on the DomainChroma's analysis results illustrated in Section 4.6.3, the number/percentage of the domain names differ significantly in each input blacklist. Specifically, over 90% of domain names in *Spamhaus-Spam, Spamhaus-Malware, Spamhaus-C&C, Uribl-Black, Uribl-Gold*, and *Ransomware* are directly used for DNS-level blacklists generated using DomainChroma. On the other hand, only less than 30% of domain names in *Spamhaus-AbusedRedirector, Spamhaus-AbusedPhish, Spamhaus-AbusedC&C*, and *Phishtank* are used for the generated blacklists.

Regarding aggregating the FQDNs that require e2LD-level defenses, a total of 918,475 FQDNs (shown in Table 12) were aggregated into 779,701 e2LDs (shown in Table 13). We confirmed that many malicious FQDNs were successfully aggregated, especially in Hphosts-FSA (fraudulent applications/services), Hphosts-EMD (malware distribution), and Hphosts-PSH (phishing). In these cases, some dedicated e2LDs were used to host many malicious subdomains or websites.

Regarding mitigating collateral damage, a total of 204,098 FQDNs (shown in Table 12) were excluded in the generated blacklist. We confirmed that many FQDNs requiring HTTP-level defenses were appropriately excluded, especially in Hphosts-PSH, Spamhaus-Spam, Hphosts-FSA, and Hphosts-EMD.

In summary, these results indicate that DomainChroma is useful for automatically maintaining meaningful DNS-level blacklists to reduce the burden of administrators.

5. Related work

Domain names have been used over the past 30 years, and there have been various types of attacks against both the system or implementation of the DNS and domain names. In this section, we classify various attacks against domain names and

Table 13 – DNS-level blacklists generated by DomainChroma in large-scale deployment.

Blacklist	# Domains [e2LD-level]	# Domains [FQDN-level]	# Domains [e2LD/FQDN-level]	# Total FQDNs
Spamhaus-Spam	380,946	44,715	425,661 (91.3%)	466,235
Spamhaus-Phish	3064	1773	4837 (85.0%)	5690
Spamhaus-Malware	297	348	645 (94.7%)	681
Spamhaus-C&C	49,421	2082	51,503 (99.5%)	51,775
Spamhaus-AbusedSpam	496	23	519 (42.5%)	1222
Spamhaus-AbusedRedirector	23	4	27 (29.0%)	93
Spamhaus-AbusedPhish	16	6	22 (27.2%)	81
Spamhaus-AbusedMalware	33	10	43 (35.8%)	120
Spamhaus-AbusedC&C	1025	52	1077 (24.3%)	4434
Uribl-Black	61,190	29,191	90,381 (92.5%)	97,683
Uribl-Grey	39	22	61 (31.4%)	194
Uribl-Red	475	226	701 (57.1%)	1228
Uribl-Gold	93,199	91,678	184,877 (92.5%)	199,908
Hphosts-EMD	39,377	84,251	123,628 (66.7%)	185,475
Hphosts-FSA	81,068	113,849	194,917 (66.2%)	294,233
Hphosts-PHA	4945	17,736	22,681 (65.0%)	34,870
Hphosts-PSH	43,044	106,423	149,467 (66.6%)	224,511
Hphosts-PUP	6349	510	6859 (52.3%)	13,127
Hphosts-WRZ	658	1487	2145 (59.8%)	3588
Phishtank	3044	268	3312 (28.0%)	11,841
Openphish	620	192	812 (32.1%)	2529
Malwaredomains	7478	6929	14,407 (68.2%)	21,130
Ransomware	1128	571	1699 (90.2%)	1883
Sans	1730	1121	2851 (82.3%)	3464
Zeus	36	10	46 (83.6%)	55
Total	779,701	503,477	1,283,178 (78.9%)	1,626,050

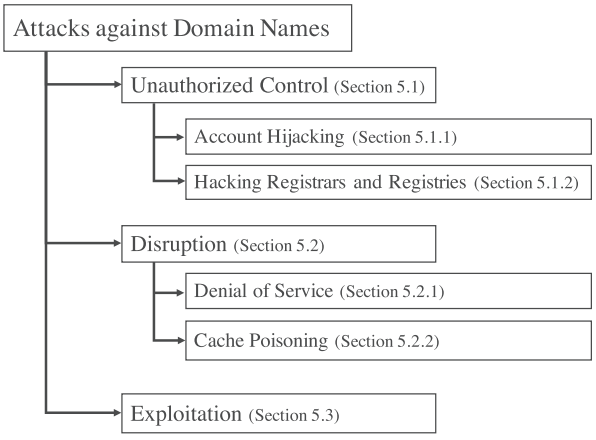

Fig. 7 – Taxonomy of attacks against domain names.

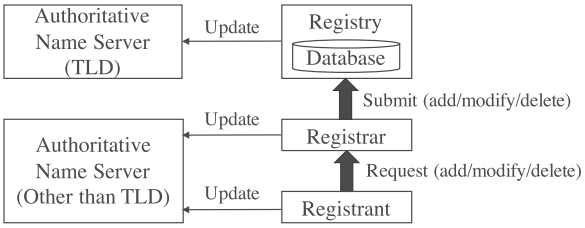

Fig. 8 – Domain-registration process.

review defense mechanisms or related work for each attack. We grouped such attacks into three major categories in terms of attack characteristics, i.e., unauthorized control, disruption, and exploitation. Fig. 7 shows the taxonomy of the categories we define in the following sections.

5.1. *Unauthorized control*

Unauthorized control is an attack against the domain-name-registration process and registered data that changes domain name information. Fig. 8 shows a simplified domain-name-registration process (ICANN WHOIS, 0000). This process involves a registry, registrar, and registrant (Hao et al., 2013; Hoffman et al., 2015). The registry manages the registration of domain names within large DNS zones, such as top-level domains (TLDs), and operates corresponding authoritative name servers. The registrar is a service provider that connects registries and registrants and manages databases in registries through the Extensible Provisioning Protocol (EPP) (Hollenbeck, 2009). The registrant is an individual or organization who wishes to register a domain name. Unauthorized control attacks can be divided into two subcategories related to attacker targets, i.e., *account hijacking* and *hacking registrars and registries*.

5.1.1. *Account hijacking*

Attackers target accounts to gain privileges to control the domain-registration process. Specifically, targets include registrant accounts used to log into the registrar system to change domain registrations and registrar accounts to change the registry database directly. For example, the registrant

account of a well-known company offering an identity management platform was hijacked in 2014. The name server record of the company's domain name was changed to the attacker's domain name (Gigya Inc., 0000). To change the registry database, an attacker targets a registrar's account using social engineering of the registrar's account-recovery function to change the target company's domain name configuration (OpenDNS, 0000). Potential defenses include domain-name-specific countermeasures, such as a registry lock service (ICA, 0000), that offers additional levels of authentication between the registry and registrar, and general security countermeasures, such as protecting passwords and strong passwords.

5.1.2. *Hacking registrars and registries*

Some attackers also target registrar and registry systems to directly control the domain-name-registration process. For example, using an SQL-injection attack against a TLD registry system, one can gain access to the management console of the registry and obtain customer e-mail addresses and passwords (Bitdefender, 0000). In another SQL-injection attack against a TLD registry, attackers publicly leaked a database that contained more than 10,000 accounts (Eha, 0000). In 2015, a command injection attack targeted a registrar's servers to change the records of a well-known company (Ars, 0000). The basic defenses against registrar and registry hacking, including SQL injections and command injections, are not specific to domain name ecosystems. They are the same as those used to protect general web systems or services, i.e., updating software, escaping characters that have a special meaning in SQL, and introducing an appropriate web-application firewall.

5.2. *Disruption*

Disruption is defined as an attack against DNS protocols or implementations. Whereas unauthorized control attacks target DNS registration data, disruption attacks target DNS components. There are two types of disruption attacks, i.e., *denial of service (DoS)* and *cache poisoning*.

5.2.1. *Denial of service*

There are three DNS-related DoS attacks, i.e., distributed DoS (DDoS) attacks against name servers, water torture DDoS attacks, and DNS reflector attacks. Distributed DoS attacks against authoritative name servers are simple but powerful ways to disrupt DNS services. In October 2016, a large U.S.-based DNS-service provider that provides authoritative name servers to large companies experienced a large-scale DDoS attack using a botnet composed of multiple infected Internet-of-Things devices (Dyn, 0000). Water torture attacks, also known as pseudo-random subdomain attacks, are DNS protocol-specific attacks against authoritative name servers that overload servers and prevent legitimate users from accessing the target servers (Secure64, 0000). For example, an attacker uses a botnet to command infected hosts to query nonexistent subdomains of legitimate domain names, such as `qazwsxedc.example.com`. Such queries can cause caching name servers and authoritative name servers to crash. Denial of service reflection attacks, including DNS amplification attacks, exploit UDP-based DNS packets to implement attacks against any server connected to the Internet (Rossow, 2014). Specifically, an attacker using this attack can send DNS queries whose source IP addresses are spoofed to a target's IP address to so-called amplifiers. The amplifiers answer the DNS queries and send a large number of DNS responses to the target. Amplifiers are typically composed of open DNS resolvers that respond to all DNS queries for any domain name (Zhang et al., 2014). One possible mitigation technique involves taking down botnets or open DNS resolvers so as not to be abused in DoS attacks. Another mitigation technique involves applying source-address validation filtering by Internet service providers (ISPs) (Beverly et al., 2009).

5.2.2. *Cache poisoning*

Cache poisoning attacks insert false domain name data in a caching name server (Perdisci et al., 2009). In this attack, an attacker sends spoofed DNS responses to a caching name server name server to cause the server to answer false or malicious IP addresses for the queried domain names. Most implementations of caching name servers have introduced randomizing the source port of DNS queries to prevent traditional cache poisoning. Kaminsky demonstrated a more efficient cache-poisoning method to evade randomization (CER, 0000). Currently, introducing DNS security extensions can prevent most such poisoning attacks and is the best solution (Arends et al., 2005).

5.3. *Exploitation*

Exploitation is defined as an abuse of domain names for conducting cyber attacks. *Unauthorized control* (Section 5.1) and *disruption* (Section 5.2) attacks target DNS-registration data and DNS servers, whereas exploitation attacks target domain names. While *unauthorized control* and *disruption* have corresponding defense/mitigation solutions, and it is clear that *who should do what* against both attack categories, there are no single or unified solutions against *exploitation*.

Detecting various malicious domain names is a common countermeasure against exploitation attacks. Previous studies mainly focused on the difference between malicious and legitimate domain names and detecting malicious domain names. However, such studies did NOT provide information of optimal countermeasures against detected malicious domain names. Our DomainChroma can use the malicious domain names detected by previous studies to build actionable threat intelligence. Specifically, DomainChroma can reveal *what, where, how*, and *until when* countermeasures need to be taken against such malicious domain names. We surveyed previous studies on detecting malicious domain names and divided them into two groups, i.e., *domain behavior* and *user behavior*.

Domain behavior. Previous studies focused on different domain name behavior, such as lexical features, registration patterns, and usage patterns, to detect malicious domain names. For example, Ma et al. detected malicious domain names and URLs based on their lexical structure (Ma et al., 2009). Felegyazhi et al. focused on using WHOIS information to detect malicious domain names (Felegyhazi et al., 2010). Notos was the first domain-reputation system to detect malicious domain names that share similar patterns in terms of IP address and domain-name usage (Antonakakis et al., 2010). Chiba et al.

leveraged time-series features of domain-name usage and network-based features of IP addresses and domain names to detect malicious domain names (Chiba et al., 2016). Predator was recently proposed by Hao et al. to detect malicious domain names when they are registered (Hao et al., 2016). Predator focuses on registration information obtained from a TLD registry.

User behavior. Other studies focused on different user behavior, such as combinations of queried domain names and DNS traffic patterns. For example, Sato et al. relied on DNS queries from multiple malware-infected devices to find malicious domain names (Sato et al., 2010). Exposure was proposed to find malicious domain names based on the time-series changes in DNS traffic or queries (Bilge et al., 2011). Antonakakis et al. proposed Kopis, which uses the characteristics of user behavior observed in authoritative name servers to detect malicious domain names (Antonakakis et al., 2011). Antonakakis et al. also proposed Pleiades, which focuses on DNS queries to non-existent domain names in caching name servers to detect malicious C&C domain names (Antonakakis et al., 2012). Segugio was proposed to detect C&C domain names from DNS traffic patterns in large ISP networks (Rahbarinia et al., 2015).

Understanding ecosystems that support malicious domain names is another countermeasure against exploitation attacks. As introduced in Section 3.2, there are many types of ecosystems and corresponding studies. However, such studies only considered individual ecosystems and did NOT provide systematic categorization of malicious domain names based on such ecosystems. DomainChroma was designed using the information revealed in previous studies to decide optimal actions to be taken. We reviewed previous studies on understanding or analyzing such ecosystems supporting malicious domain names.

Advertising. Cyber attackers have used online advertising ecosystems as attack vectors to reach target users effectively. Zarras et al. revealed that 1% of online advertisements are used to lead users to malicious content (Zarras et al., 2014). Xing et al. found that common web-browser extensions deliver malware via the advertising ecosystem and have affected more than 600,000 users (Xing et al., 2015).

CDN. Attackers have abused CDNs as a reliable and efficient infrastructure to distribute malicious content. Lever et al. analyzed how malware uses CDNs using domain names collected from dynamic malware analysis (Lever et al., 2017). Stringhini et al. showed that some malware operators use CDNs exclusively to deliver malware samples (Stringhini et al., 2017).

Hosting. Previous studies (Akiyama et al., 2011; Canali et al., 2013; Stokes et al., 2010) revealed that attackers use hosting services, such as cloud and file-sharing, to host malicious file objects since web hosting is an economical option for any user on the Internet.

DGA. Attackers use a DGA to generate a huge number of distinct AGDs then use only a small subset of generated domain names as C&C communication to evade countermeasures. Previous studies (Bilge et al., 2011; Schiavoni et al., 2014; Yadav et al., 2010) analyzed DGA-specific linguistic features for detecting/classifying DGAs. Plohmann et al. conducted a comprehensive measurement study of 43 DGA-based malware families and proposed a taxonomy for DGA algorithms (Plohmann et al., 2016).

Re-registration. Expired domain names, particularly popular domain names, tend to be targeted and immediately re-registered by attackers for malicious purposes, such as phishing attacks. Hao et al. analyzed the characteristics of the re-registration process of malicious domain names to develop a domain reputation system (Hao et al., 2016). Lever et al. revealed that re-registered domain names inherit the residual trust associated with the domain names' prior usage and identified many maliciously re-registered domain names resolved by malware samples (Lever et al., 2016). Lauinger et al. focused on domain names about to be deleted and analyzed WHOIS data for a survival analysis of re-registrations (Lauinger et al., 2016).

Sinkholing. Security researchers and organizations often use a sinkholing technique to take control of malware C&C domain names. Rahbarinia et al. proposed a system to detect IP addresses corresponding to sinkholing operations by leveraging passive DNS data (Rahbarinia et al., 2013). Kührer et al. proposed a method of identifying sinkholed domain names by using graph-based approaches that use the relationships between sinkholed domain names and IP addresses (Kührer et al., 2014).

Parking. Cyber attackers tend to use parking services to monetize malicious traffic. Alrwais et al. analyzed parking services through a unique observation of a monetization process to confirm the presence of click fraud and traffic spam (Alrwais et al., 2014). Vissers et al. analyzed the ecosystem of domain parking in terms of consequences of accessing parked domain names and showed that users landing on domain parking websites are exposed to various types of malware and scams (Vissers et al., 2015).

Typosquatting. Typosquatting is an attack technique to register similar domain names to popular or legitimate services to monetize traffic generated from a typing error. Szurdi et al. conducted a study of typosquatting domain registrations and found that even less popular domain names are targeted by this attack (Szurdi et al., 2014). Agten et al. presented a content-based and longitudinal study of typosquatting domain names to reveal that 95% of popular domain names are actively targeted by attackers (Agten et al., 2015).

Dynamic DNS. Attackers tend to abuse dynamic DNS services to conduct cyber attacks due to their low cost and high availability. Lever et al. showed that about 32% of all malware samples in their dataset queried at least one dynamic DNS domain name (Lever et al., 2017).

Gratuitous. Some domain-registration services offer gratuitously available domain names under some TLDs and e2LDs. Previous studies showed that such services are easily abused by cyber attackers to create malicious domain names (Li et al., 2013; Rahbarinia et al., 2016; 2015).

6. Conclusion

We designed and implemented a unified analysis system called DomainChroma to reveal *what*, *where*, *how*, and *until when* countermeasures should be taken against malicious

domain names for websites. The concept of malicious domain name *chromatography*, which was defined here to mean the separation of mixtures composed of various types of malicious domain names, was applied to DomainChroma. Based on this concept and systematized knowledge, DomainChroma builds actionable threat intelligence from today's malicious domain names without incurring collateral damage of legitimate services. We evaluated DomainChroma using a large real dataset to show that over 70% of domain names require only DNS-level defense with no collateral damage of legitimate accesses. Moreover, we illustrated the quality differences between multiple commercial and public domain name blacklists, including over 1.6 million domain names, to show the effectiveness of DomainChroma in terms of generating optimal DNS-level blacklists. We showed that DomainChroma automatically aggregated 139 thousand FQDNs into e2LDs and excluded 204 thousand FQDNs that caused collateral damage. We hope that the knowledge and results in this paper can be used to improve both the techniques and operations in DNS-level and HTTP-level points of defense to defend against attacks using domain names and the DNS in the future.

REFERENCES

Adblock Plus, 2017. https://adblockplus.org/.

Afilias Domain Anti-Abuse Policy, 2014. https://www.afilias.info/sites/afilias.info/files/Afilias%20Domain%20Anti-Abuse%20Policy.pdf

Agten P, Joosen W, Piessens F, Nikiforakis N. Seven months' worth of mistakes: a longitudinal study of typosquatting abuse. Proceedings of the twenty-second annual network and distributed system security symposium (NDSS'15), 2015.

Akiyama M, Yagi T, Itoh M. Searching structural neighborhood of malicious URLs to improve blacklisting. Proceedings of the eleventh annual IEEE/IPSJ international symposium on applications and the internet (SAINT'11); 2011. p. 1–10. doi: 10.1109/SAINT.2011.11.

Alexa Top Sites, 2017. http://www.alexa.com/topsites

Alrwais SA, Yuan K, Alowaisheq E, Li Z, Wang X. Understanding the dark side of domain parking. Proceedings of the twenty-third USENIX security symposium; 2014. p. 207–22.

Amazon CloudFront. Locations and IP address ranges of CloudFront edge servers, 2017. https://docs.aws.amazon.com/AmazonCloudFront/latest/DeveloperGuide/LocationsOfEdgeServers.html.

Antonakakis M, Perdisci R, Dagon D, Lee W, Feamster N. Building a dynamic reputation system for DNS. Proceedings of the nineteenth USENIX security symposium, 2010.

Antonakakis M, Perdisci R, Lee W, Vasiloglou II N, Dagon D. Detecting malware domains at the upper DNS hierarchy. Proceedings of the twentieth USENIX security symposium, 2011.

Antonakakis M, Perdisci R, Nadji Y, Vasiloglou N, Abu-Nimeh S, Lee W, Dagon D. From throw-away traffic to bots: detecting the rise of DGA-based malware. Proceedings of the twenty-first USENIX security symposium, 2012.

Arends R., Austein R., Larson M., Massey D., Rose S. DNS security introduction and requirements. RFC 4033, RFC. (Proposed Standard). http://www.ietf.org/rfc/rfc4033.txt; 2005.

Beverly R, Berger AW, Hyun Y, kc claffy. Understanding the efficacy of deployed internet source address validation filtering. Proceedings of the ninth ACM internet measurement Conference (IMC'09); 2009. p. 356–69.

Bilge L, Kirda E, Kruegel C, Balduzzi M. EXPOSURE: finding malicious domains using passive DNS analysis. Proceedings of the network and distributed system security symposium (NDSS'11), 2011.

Bitdefender. Turkmenistan TLD leaks domain data, unencrypted passwords, 2013. https://www.hotforsecurity.com/blog/turkmenistan-tld-leaks-domain-data-unencrypted-passwords-5153.html.

Breiman L. Random forests. Mach Learn 2001;45(1):5–32 doi:10.1023/A:1010933404324.

CDNPlanet, 2017. http://www.cdnplanet.com/cdns/.

Canali D, Balzarotti D, Francillon A. The role of web hosting providers in detecting compromised websites. Proceedings of the twenty-second international world wide web conference (WWW'13); 2013. p. 177–88.

Chen J, Zheng X, Duan H, Liang J, Jiang J, Li K, Wan T, Paxson V. Forwarding-loop attacks in content delivery networks. Proceedings of the twenty-third annual network and distributed system security symposium (NDSS'16).

Chen Y, Antonakakis M, Perdisci R, Nadji Y, Dagon D, Lee W. DNS noise: measuring the pervasiveness of disposable domains in modern DNS traffic. Proceedings of the forty-fourth annual IEEE/IFIP international conference on dependable systems and networks (DSN'14); 2014. p. 598–609. doi: 10.1109/DSN.2014.61.

Chen Y, Kintis P, Antonakakis M, Nadji Y, Dagon D, Lee W, Farrell M. Financial lower bounds of online advertising abuse – a four year case study of the TDSS/TDL4 botnet. Proceedings of the thirteenth international Conference on detection of intrusions and malware, and vulnerability assessment (DIMVA'16); 2016b. p. 231–54. doi: 10.1007/978-3-319-40667-1_12.

Chiba D, Akiyama M, Yagi T, Yada T, Mori T, Goto S. DomainChroma: providing optimal countermeasures against malicious domain names. Proceedings of the forty-first IEEE annual computer software and applications conference, COMPSAC 2017, Turin, Italy, July 4–8. IEEE Computer Society, 2017. doi: 10.1109/COMPSAC.2017.112.

Chiba D, Yagi T, Akiyama M, Shibahara T, Yada T, Mori T, Goto S. DomainProfiler: discovering domain names abused in future. Proceedings of the forty-sixth annual IEEE/IFIP international conference on dependable systems and networks (DSN'16); 2016. p. 491–502. doi: 10.1109/DSN.2016.51.

CloudFlare, 2017. CloudFlare IP ranges. https://www.cloudflare.com/ips/.

DGArchive, 2017. https://dgarchive.caad.fkie.fraunhofer.de/.

DMOZ – The Directory of the Web, 2017. http://www.dmoz.org/.

DNS-BH Malware Domain Blocklist, 2017. http://www.malwaredomains.com/.

DNS-BH - Malware Domain Blocklist. Bulk registrars, 2017. http://mirror1.malwaredomains.com/files/bulk_registrars.txt.

DNS-BH - Malware Domain Blocklist. Dynamic DNS, 2017. http://mirror1.malwaredomains.com/files/dynamic_dns.txt.

DNS-BH - Malware Domain Blocklist. Free web hosts, 2017. http://mirror1.malwaredomains.com/files/freewebhosts.txt.

DYN. DYN analysis summary of friday october 21 attack, 2016. http://dyn.com/blog/dyn-analysis-summary-of-friday-october-21-attack/.

EasyList, 2017. https://easylist.to/easylist/easylist.txt.

Farsight Security, Inc. DNSDB, 2017. https://www.dnsdb.info/.

Fastly. Accessing Fastly's IP ranges, 2017. https://docs.fastly.com/guides/securing-communications/accessing-fastlys-ip-ranges.

Felegyhazi M, Kreibich C, Paxson V. On the potential of proactive domain blacklisting. Proceedings of the third USENIX conference on large-scale exploits and emergent threats (LEET'10), 2010.

Free Domains With Full DNS Support, 2017. http://freeavailabledomains.com/en/.

Freenom, 2017. http://www.freenom.com/en/index.html.
Free DNS. Domain registry, 2017. http://freedns.afraid.org/domain/registry/.
Gañán C, Cetin O, van Eeten M. An empirical analysis of zeus C&C lifetime. Proceedings of the tenth ACM symposium on information, computer and communications security (AsiaCCS'15); 2015. p. 97–108.
Gigya Inc. Regarding today's service attack, 2014. http://www.gigya.com/blog/regarding-todays-service-attack/.
Google Safe Browsing, 2017. https://developers.google.com/safe-browsing/.
Hao S, Kantchelian A, Miller B, Paxson V, Feamster N. PREDATOR: proactive recognition and elimination of domain abuse at time-of-registration. Proceedings of the twenty-third ACM conference on computer and communications security (CCS'16); 2016. p. 1568–79.
Hao S, Thomas M, Paxson V, Feamster N, Kreibich C, Grier C, Hollenbeck S. Understanding the domain registration behavior of spammers. Proceedings of the ACM internet measurement conference (IMC'13); 2013. p. 63–76. doi: 10.1145/2504730.2504753.
Hoffman P, Sullivan A, Fujiwara K. DNS terminology. RFC 7719, RFC. (Informational). http://www.ietf.org/rfc/rfc7719.txt; 2015.
Hollenbeck S. Extensible provisioning protocol (EPP). RFC 5730, RFC. (Internet standard). http://www.ietf.org/rfc/rfc5730.txt; 2009.
hpHosts, 2017. http://www.hosts-file.net/.
hpHosts – Ad and Tracking Servers Only, 2017. https://hosts-file.net/ad_servers.txt.
ICANN Registry Request Service – Registry Lock Service, 2009. https://www.icann.org/en/system/files/files/verisign-reglock-request-25jun09-en.pdf.
ICANN. Registry listing, 2012. https://www.icann.org/resources/pages/listing-2012-02-25-en.
ICANN WHOIS. Domain name registration process, 2017. https://whois.icann.org/en/domain-name-registration-process.
Kaplan EL, Meier P. Nonparametric estimation from incomplete observations. J Am Stat Assoc 1958;53(282):457–81.
Khan MT, Huo X, Li Z, Kanich C. Every second counts: quantifying the negative externalities of cybercrime via typosquatting. Proceedings of the IEEE symposium on security and Privacy (SP'15); 2015. p. 135–50. doi: 10.1109/SP.2015.16.
Kountouras A, Kintis P, Lever C, Chen Y, Nadji Y, Dagon D, Antonakakis M, Joffe R. Enabling network security through active DNS datasets. Proceedings of the nineteenth international symposium on research in attacks, intrusions, and defenses (RAID'16); 2016. p. 188–208. doi: 10.1007/978-3-319-45719-2_9.
Kührer M, Rossow C, Holz T. Paint it black: evaluating the effectiveness of malware blacklists. Proceedings of the seventeenth international symposium on research in attacks, intrusions and defenses (RAID'14); 2014. p. 1–21. doi: 10.1007/978-3-319-11379-1_1.
Lauinger T, Onarlioglu K, Chaabane A, Robertson W, Kirda E. WHOIS lost in translation: (mis)understanding domain name expiration and re-registration. Proceedings of the ACM internet measurement conference (IMC'16), 2016.
Lenovo.com. Hijack reportedly pulled off by hack on upstream registrar, 2015. http://arstechnica.com/security/2015/02/lenovo-com-hijack-reportedly-pulled-off-by-hack-on-upstream-registrar/.
Lever C, Kotzias P, Balzarotti D, Caballero J, Antonakakis M. A lustrum of malware network communication: evolution and insights. Proceedings of the IEEE symposium on security and privacy (SP'17); 2017. p. 788–804. doi: 10.1109/SP.2017.59.
Lever C, Walls RJ, Nadji Y, Dagon D, McDaniel P, Antonakakis M. Domain-Z: 28 registrations later measuring the exploitation of residual trust in domains. Proceedings of the IEEE symposium on security and privacy (SP'16); 2016. p. 691–706. doi: 10.1109/SP.2016.47.
Li F, Durumeric Z, Czyz J, Karami M, Bailey M, McCoy D, Savage S, Paxson V. You've got vulnerability: exploring effective vulnerability notifications. Proceedings of the twenty-fifth USENIX security symposium; 2016a. p. 1033–50.
Li F, Ho G, Kuan E, Niu Y, Ballard L, Thomas K, Bursztein E, Paxson V. Remedying web hijacking: notification effectiveness and webmaster comprehension. Proceedings of the twenty-fifth international conference on world wide web (WWW'16); 2016b. p. 1009–19. doi: 10.1145/2872427.2883039.
Li Z, Alrwais SA, Xie Y, Yu F, Wang X. Finding the linchpins of the dark web: a study on topologically dedicated hosts on malicious web infrastructures. Proceedings of the thirty-fourth IEEE symposium on security and privacy (SP'13); 2013. p. 112–26. doi: 10.1109/SP.2013.18.
Liu S, Foster ID, Savage S, Voelker GM, Saul LK. Who is .com?: learning to parse WHOIS records. Proceedings of the ACM internet measurement conference (IMC'15); 2015. p. 369–80. doi: 10.1145/2815675.2815693.
Ma J, Saul LK, Savage S, Voelker GM. Beyond blacklists: learning to detect malicious web sites from suspicious URLs. Proceedings of the fifteenth ACM SIGKDD international conference on knowledge discovery and data mining (KDD'09); 2009. p. 1245–54. doi: 10.1145/1557019.1557153.
Malware Domain List, 2017. http://www.malwaredomainlist.com/.
Microsoft. Microsoft takes on global cybercrime epidemic in tenth malware disruption, 2014. https://blogs.technet.microsoft.com/microsoft_blog/2014/06/30/microsoft-takes-on-global-cybercrime-epidemic-in-tenth-malware-disruption/
Microsoft Corporation. Microsoft cognitive services Bing search engine APIs, 2017. https://azure.microsoft.com/en-us/services/cognitive-services/search/
Mockapetris P. Domain names: concepts and facilities. RFC 882, RFC. http://www.ietf.org/rfc/rfc882.txt; 1983a.
Mockapetris P. Domain names: implementation specification. RFC 883, RFC. http://www.ietf.org/rfc/rfc883.txt; 1983b.
Mozilla Foundation. Public suffix list, 2017. https://publicsuffix.org/list/.
Multiple DNS Implementations Vulnerable to Cache Poisoning, 2008. http://www.kb.cert.org/vuls/id/800113.
Nadji Y, Antonakakis M, Perdisci R, Dagon D, Lee W. beheading hydras: performing effective botnet takedowns. Proceedings of the ACM conference on computer and communications security (CCS'13); 2013a. p. 121–32. doi: 10.1145/2508859.2516749.
Nadji Y, Antonakakis M, Perdisci R, Lee W. connected colors: unveiling the structure of criminal networks. Proceedings of the sixteenth international symposium on research in attacks, intrusions, and defenses (RAID'13); 2013b. p. 390–410. doi: 10.1007/978-3-642-41284-4_20.
Nelms T, Perdisci R, Ahamad M. ExecScent: mining for new C&C domains in live networks with adaptive control protocol templates. Proceedings of the twenty-second USENIX security symposium; 2013. p. 589–604.
No-IP's Formal Statement on Microsoft Takedown, 2014. https://www.noip.com/blog/2014/06/30/ips-formal-statement-microsoft-takedown/
Noroozian A, Korczynski M, Gañán CH, Makita D, Yoshioka K, van Eeten M. Who gets the boot? Analyzing victimization by ddos-as-a-service. Proceedings of the nineteenth international symposium on research in attacks, intrusions, and defenses (RAID'16); 2016. p. 368–89.
OpenDNS. Five things to know about the tesla motors compromise, 2015. https://blog.opendns.com/2015/04/27/five-things-to-know-about-the-tesla-motors-compromise/.

OpenPhish, 2017. https://openphish.com/.
Perdisci R, Antonakakis M, Luo X, Lee W. WSEC DNS: protecting recursive DNS resolvers from poisoning attacks. Proceedings of the IEEE/IFIP international conference on dependable systems and networks (DSN'09); 2009. p. 3–12.
Plohmann D, Yakdan K, Klatt M, Bader J, Gerhards-Padilla E. A comprehensive measurement study of domain generating malware. Proceedings of the twenty-fifth USENIX security symposium; 2016. p. 263–78.
PhishTank, 2017. https://www.phishtank.com/.
Ransomware Tracker, 2017. https://ransomwaretracker.abuse.ch/blocklist/.
Rahbarinia B, Balduzzi M, Perdisci R. Real-time detection of malware downloads via large-scale URL->file->machine graph mining. Proceedings of the eleventh ACM Asia conference on computer and communications security (AsiaCCS'16); 2016. p. 783–94. doi: 10.1145/2897845.2897918.
Rahbarinia B, Perdisci R, Antonakakis M. Segugio: efficient behavior-based tracking of malware-control domains in large ISP networks. Proceedings of the forty-fifth annual IEEE/IFIP international conference on dependable systems and networks (DSN'15); 2015. p. 403–14. doi: 10.1109/DSN.2015.35.
Rahbarinia B, Perdisci R, Antonakakis M, Dagon D. SinkMiner: mining botnet sinkholes for fun and profit. Proceedings of the sixth USENIX workshop on large-scale exploits and emergent threats (LEET'13), 2013.
Rossow C. Amplification hell: revisiting network protocols for DDoS abuse. Proceedings of the twenty-first annual network and distributed system security symposium (NDSS'14), 2014.
Rossow C, Andriesse D, Werner T, Stone-Gross B, Plohmann D, Dietrich CJ, Bos H. Sok: P2PWNED – modeling and evaluating the resilience of peer-to-peer botnets. Proceedings of the IEEE symposium on security and privacy (SP'13); 2013. p. 97–111. doi: 10.1109/SP.2013.17.
SANS Internet Storm Center. Suspicious domains, 2017. https://isc.sans.edu/suspicious_domains.html.
Sato K, Ishibashi K, Toyono T, Miyake N. Extending black domain name list by using co-occurrence relation between DNS queries. Proceedings of the third USENIX conference on large-scale exploits and emergent threats (LEET'10), 2010.
Schiavoni S, Maggi F, Cavallaro L, Zanero S. Phoenix: DGA-based botnet tracking and intelligence. Proceedings of the eleventh conference on detection of intrusions and malware and vulnerability assessment (DIMVA'14); 2014. p. 192–211.
Secure64. Water torture: a slow drip DNS DDoS attack, 2014. https://secure64.com/water-torture-slow-drip-dns-ddos-attack/.
Sri Lankan NIC site (nic.lk) hacked via SQL Injection Vulnerability, 2013. http://www.ehackingnews.com/2013/01/sri-lankan-nic-siteniclk-hacked-via-sql.html.
Stock B, Pellegrino G, Rossow C, Johns M, Backes M. Hey, you have a problem: on the feasibility of large-scale web vulnerability notification. Proceedings of the twenty-fifth USENIX security symposium; 2016. p. 1015–32.
Stokes JW, Andersen R, Seifert C, Chellapilla K. WebCop: Locating neighborhoods of malware on the web. Proceedings of the third USENIX workshop on large-scale exploits and emergent threats (LEET'10), 2010.
Stringhini G, Shen Y, Han Y, Zhang X. Marmite: spreading malicious file reputation through download graphs. Proceedings of the thirty-third annual computer security applications conference (ACSAC'17); 2017. p. 91–102. doi: 10.1145/3134600.3134604.
Szurdi J, Kocso B, Cseh G, Spring J, Felegyhazi M, Kanich C. The long "taile" of typosquatting domain names. Proceedings of the twenty-third USENIX security symposium; 2014. p. 191–206.
The Spamhaus Project Ltd. The domain block list, 2017. https://www.spamhaus.org/dbl/.
URIBL, 2017. http://uribl.com/about.shtml.
Vadrevu P, Rahbarinia B, Perdisci R, Li K, Antonakakis M. Measuring and detecting malware downloads in live network traffic. Proceedings of the eighteenth European symposium on research in computer security (ESORICS'13); 2013. p. 556–73. doi: 10.1007/978-3-642-40203-6_31.
Verisign Anti-Abuse Domain Use Policy, 2011. https://www.icann.org/en/system/files/files/verisign-com-net-name-request-10oct11-en.pdf
VirusTotal, 2017. https://www.virustotal.com/.
Vissers T, Joosen W, Nikiforakis N. Parking sensors: analyzing and detecting parked domains. Proceedings of the twenty-second annual network and distributed system security symposium (NDSS'15), 2015.
Vixie P. DNS complexity. ACM Que 2007;5(3):24–9 doi:10.1145/1242489.1242499.
Vixie P, Thomson S, Rekhter Y, Bound J. Dynamic updates in the domain name system (DNS UPDATE). RFC 2136, RFC. (Proposed Standard), http://www.ietf.org/rfc/rfc2136.txt; 1997.
Wang Y, Beck D, Wang J, Verbowski C, Daniels B. strider typo-patrol: discovery and analysis of systematic typo-squatting. Proceedings of the second workshop on steps to reducing unwanted traffic on the internet (SRUTI'06), 2006.
Web Hosting Statistics & Information, 2017. http://webhosting.info/web-hosting-statistics.
Webpageteset/cdn.h, 2017. https://github.com/WPO-Foundation/webpagetest/blob/master/agent/wpthook/cdn.h.
WebPagetest, 2017. https://www.webpagetest.org/.
Xing X, Meng W, Lee B, Weinsberg U, Sheth A, Perdisci R, Lee W. Understanding malvertising through ad-injecting browser extensions. Proceedings of the twenty-fourth international conference on world wide web (WWW'15); 2015. p. 1286–95. doi: 10.1145/2736277.2741630.
Yadav S, Reddy AKK, Reddy ALN, Ranjan S. Detecting algorithmically generated malicious domain names. Proceedings of the tenth ACM internet measurement conference (IMC'10); 2010. p. 48–61. doi: 10.1145/1879141.1879148.
Zarras A, Kapravelos A, Stringhini G, Holz T, Kruegel C, Vigna G. The dark alleys of madison avenue: understanding malicious advertisements. Proceedings of the acm internet measurement conference (IMC'14); 2014. p. 373–80. doi: 10.1145/2663716.2663719.
Zeus Tracker, 2017. https://zeustracker.abuse.ch/blocklist.php.
Zhang J, Durumeric Z, Bailey M, Liu M, Karir M. On the mismanagement and maliciousness of networks. Proceedings of the twenty-first annual network and distributed system security symposium (NDSS'14), 2014.

Daiki Chiba is currently a researcher at NTT Secure Platform Laboratories, Tokyo, Japan. He received his B.E., M.E., and Ph.D. degrees in computer science from Waseda University in 2011, 2013, and 2017. Since joining Nippon Telegraph and Telephone Corporation (NTT) in 2013, he has been engaged in research on cyber security through data analysis. He is a member of IEEE and IEICE.

Mitsuaki Akiyama received his M.E. and Ph.D. degrees in Information Science from Nara Institute of Science and Technology, Japan in 2007 and 2013. Since joining Nippon Telegraph and Telephone Corporation NTT in 2007, he has been engaged in research and development of network security, especially honeypot and malware analysis. He is now with the Cyber Security Project of NTT Secure Platform Laboratories.

Takeshi Yagi received his B.E. degree in electrical and electronic engineering and his M.E. degree in science and technology from Chiba University, Japan in 2000 and 2002. He also received his Ph.D. degree in information science and technology from Osaka University, Osaka, Japan in 2013. Since joining NTT in 2002, he has been

engaged in research and design of network architecture, traffic engineering, and cyber security. He is now a senior research engineer in the Cyber Security Project of NTT Secure Platform Laboratories. He is a member of IEEE and IEEJ.

Kunio Hato received his B.E. and M.E. degrees in information processing from Tokyo Institute of Technology in 1997 and 1999, respectively. Since joining Nippon Telegraph and Telephone Corporation (NTT) in 1999, he has been engaged in research and development of IP VPNs, wide area Ethernet, and network security systems. He is now a Senior Research Engineer, Supervisor, in Cyber Security Project of NTT Secure Platform Laboratories. He was with the Network Services of NTT communications from 2014 to 2017. He is a member of IEICE.

Tatsuya Mori is currently a professor at Waseda University, Tokyo, Japan. He received B.E. and M.E. degrees in applied physics, and Ph.D. degree in information science from the Waseda University, in 1997, 1999 and 2005, respectively. He joined NTT lab in 1999. Since then, he has been engaged in the research of measurement and analysis of networks and cyber security. From Mar 2007 to Mar 2008, he was a visiting researcher at the University of Wisconsin-Madison. Dr. Mori is a member of ACM, IEEE, IEICE, IPSJ, and USENIX.

Shigeki Goto is a professor at the Department of Computer Science and Engineering, Waseda University, Japan. He received his B.S. and M.S. in Mathematics from the University of Tokyo. Prior to becoming a professor at Waseda University, he has worked for NTT for many years. He also earned his Ph.D. in Information Engineering from the University of Tokyo. He is the president of JPNIC. He is a member of ACM and IEEE, and he was a trustee of Internet Society from 1994 to 1997.