Operational Domain Name Classification: From Automatic Ground Truth Generation to Adaptation to Missing Values

Jan Bayer[1], Ben Chukwuemeka Benjamin[1], Sourena Maroofi[2], Thymen Wabeke[3], Cristian Hesselman[3,4], Andrzej Duda[1], and Maciej Korczyński[1]

[1] Univ. Grenoble Alpes, CNRS, Grenoble INP, LIG
{first.last}@univ-grenoble-alpes.fr
[2] KOR Labs Cybersecurity
[3] SIDN Labs
[4] University of Twente

Abstract. With more than 350 million active domain names and at least 200,000 newly registered domains per day, it is technically and economically challenging for Internet intermediaries involved in domain registration and hosting to monitor them and accurately assess whether they are benign, likely registered with malicious intent, or have been compromised. This observation motivates the design and deployment of automated approaches to support investigators in preventing or effectively mitigating security threats. However, building a domain name classification system suitable for deployment in an operational environment requires meticulous design: from feature engineering and acquiring the underlying data to handling missing values resulting from, for example, data collection errors. The design flaws in some of the existing systems make them unsuitable for such usage despite their high theoretical accuracy. Even worse, they may lead to erroneous decisions, for example, by registrars, such as suspending a benign domain name that has been compromised at the website level, causing collateral damage to the legitimate registrant and website visitors.

In this paper, we propose novel approaches to designing domain name classifiers that overcome the shortcomings of some existing systems. We validate these approaches with a prototype based on the COMAR (COmpromised versus MAliciously Registered domains) system focusing on its careful design, automated and reliable ground truth generation, feature selection, and the analysis of the extent of missing values. First, our classifier takes advantage of automatically generated ground truth based on publicly available domain name registration data. We then generate a large number of machine-learning models, each dedicated to handling a set of missing features: if we need to classify a domain name with a given set of missing values, we use the model without the missing feature set, thus allowing classification based on all other features. We estimate the importance of features using scatter plots and analyze the extent of missing values due to measurement errors.

Finally, we apply the COMAR classifier to unlabeled phishing URLs and find, among other things, that 73% of corresponding domain names are maliciously registered. In comparison, only 27% are benign domains hosting malicious websites. The proposed system has been deployed at two ccTLD registry operators to support their anti-fraud practices.

Keywords: DNS · domain name abuse · classification · phishing · malicious domain registration · compromised websites

1 Introduction

Attackers have traditionally used domain names to spread malware, ensure reliable communication between malicious command-and-control (C&C) servers and botnets using domain generation algorithms (DGAs), or to launch spam or phishing campaigns. A domain name can be registered for a legitimate purpose by a benign registrant or with malicious intent by an attacker. A benign domain name can also be compromised at the hosting, domain, or website level, and involved in malicious activities later in its lifetime.

With more than 350 million active domain names[5] and at least 200 thousand newly registered domains per day,[6] it is technically and economically challenging for top-level domain (TLD) registries and registrars to scrutinize them at the time of registration and accurately assess whether they are benign or likely registered with malicious intent. Furthermore, once a domain name is involved in a malicious activity, and the abusive URL is blacklisted, or reported to the operator's helpdesk, an investigator must gather evidence on whether the domain name is attacker-owned (i.e., registered by a malicious actor) or has been compromised (and possibly how) before deciding on the type of the mitigation action. While a maliciously registered domain name can be suspended, a benign and subsequently hacked domain name generally cannot be blocked because it may cause collateral damage to the harmless domain name owner and regular visitors of legitimate websites available under the benign domain name. Instead, the webmaster or the hosting provider should remove the malicious content (e.g., malware or phishing website) from the server and patch the vulnerable application to prevent future intrusions [48].

The problem of DNS abuse and domain names being a vehicle for delivering malicious content [5, 27, 28, 36, 47] motivates the development and implementation of automated methods to support investigators in assessing domain name maliciousness as well as appropriate and prompt mitigation of security threats. To address these challenges, several research studies proposed domain name reputation systems based on machine learning (ML) to distinguish between benign registrations and the malicious ones [4, 6, 7, 15, 16, 20, 24, 31, 33, 45] as well as compromised domain names and those owned by attackers [10, 29, 34].

However, building a fully automated domain name classification system that can be effectively used and deployed in an operational environment requires

[5] https://www.verisign.com/assets/domain-name-report-Q22022.pdf

[6] https://zonefiles.io

meticulous design: from feature engineering and acquiring the underlying data to handling missing values resulting from measurement and data collection errors. The design flaws of existing classifiers may make them unsuitable in operational environments despite their high theoretical accuracy. Even worse, incorrect classification may lead to misguided decisions by intermediaries such as the suspension of a benign domain name, causing collateral damage to their legitimate users and the painstaking process of reclaiming the domain by its rightful registrant.

Most of the proposed systems make use of privileged, closed, or pay-walled data (e.g., passive DNS, non-public registration information, retail pricing of domain names, or search engine results). Therefore, building such classifiers can be costly or complicated for those involved in DNS operations to assess and mitigate domain name abuse and challenging for researchers to replicate previous scientific results.

Furthermore, since the domain name classifiers often use supervised machine learning methods, researchers strive for high-quality ground truth data to train robust models. Their sources vary from one study to another: some rely on third-party sources such as Google Safe Browsing [24], some others on website or domain popularity ranking lists [4, 24], or blacklists [4, 20, 24], while lacking insight into the underlying proprietary methodology used by their providers. Another approach to obtain high-quality ground truth data is to manually label the dataset [10, 34]. However, it is a time-consuming process requiring expert knowledge and the datasets may quickly become outdated.

Another design issue is related to handling missing values in ground truth and unlabeled data. Previous methods tended to impute missing values using statistical methods (e.g., the mean of a group [13, 31]). Maroofi et al. [34] proposed another approach to deal with missing values: use other available data for selected features (e.g., estimating the domain registration date based on privileged passive DNS). However, not all proposed methods can be applied to different types of features. Moreover, as models are often trained and evaluated on data with a complete feature value vector, the domain names with missing values arising from, for example, measurement errors, may not be classified. The number of unclassified cases may be significant and can affect the operational utility of the deployed domain classifier.

In this paper, we propose novel approaches to designing domain name classifiers that overcome the shortcomings of existing systems. We present a method to automatically generate ground truth based on publicly available domain name registration data. We generate a large number of ML models, each dedicated to handling a set of missing features: if we need to classify a domain name with a given set of missing values, we use the model without the missing feature set thus supporting classification based on all other features. The proposed design principles apply to any domain name classifier.

We validate these approaches with a prototype based on the COMAR (COmpromised versus MAliciously Registered domains) system [34] focusing on its careful design, automated and reliable ground truth generation, feature selection, the analysis of the extent of missing values resulting from measurement errors, and on its extensive evaluation.

We also apply the implemented classifier to 20 months of phishing data, study selected characteristics of the domain names of malicious URLs, and analyze their distribution across different types of TLDs. The system has shown its suitability for efficiently classifying compromised and maliciously registered domain names as two country-code TLD (ccTLD) operators have deployed it to support their DNS anti-abuse practices.

Our contributions can be summarized as follows:

- We develop a novel technique for automatically generating ground-truth data for compromised (benign) and maliciously registered domains. It consists of measuring the mitigation actions on abusive domains by TLD registries, registrars, and hosting providers.
- We propose a visualization method to assess the importance of features using scatter plots and analyze the features most likely to be missing due to measurement errors.
- We propose an approach based on multiple trained models to account for missing values, as opposed to traditional methods based on imputing missing values using statistical methods.
- We apply the COMAR classifier to domain names extracted from phishing URLs and find that while for legacy gTLDs and ccTLDs between 27% and 31% of abused domains are benign but possibly exploited at the website level, the vast majority of new gTLD domain names are maliciously registered.
- As many as 66.1% of the maliciously registered domain names have no specific technology on their homepages. In comparison, 52.2% of compromised domains use more than five different frameworks and plugins to build the website, making them more susceptible to web application attacks.

2 Background and Related Work

Several researchers proposed domain name classifiers to address the problem of domain name abuse [4,6,7,10,15,16,20,24,29,31,33,34,45]. Many studies provided domain name reputation scores indicative of whether they are malicious (i.e., registered by a miscreant for cybercriminal purposes) or benign (i.e., registered by a benign user for legitimate purposes) [4, 6, 7, 15, 16, 20, 24, 31, 33, 45]. Some recent work proposed distinguishing between maliciously registered and benign but compromised domain names [10, 29, 34]. The last type corresponds to the domains taken over by attackers, for example, through vulnerabilities in libraries or frameworks such as content management systems used to build websites. In this section, we identify the key challenges in domain name classification and discuss how the existing methods address them.

2.1 Data and Feature Selection

After formulating the classification problem and the outcomes (i.e., labels), one of the starting points in designing any domain name classifier is the selection of data sources and *features* for distinguishing between two groups of domain names (e.g., benign and malicious).

The primary criterion for selecting data sources is their availability. Datasets used in DNS reputation systems can be either publicly or non-publicly available. The privileged or commercial sources such as the passive DNS data used in the Exposure [6,7], Notos [4], or Predator [20] are only limited to those who have access to such data. Historical data raise a similar problem (e.g., historical WHOIS data used in the takedown of Avalanche [31]). Furthermore, reproducibility and performance validation of the systems relying on non-publicly available data by independent researchers may be difficult or impossible.

On the other hand, systems based on publicly available data sources do not have the problems raised by non-public data sources and can still achieve high accuracy. Moreover, they are more likely adopted by the involved operators, not only DNS intermediaries but also, for example, law enforcement agencies [31].

The Mentor [24] and Domain Classifier [29] systems used public data sources and demonstrated high accuracy. De Silva et al. [10] combined public and non-public (passive DNS data from Farsight [14]) data sources to achieve 97.2% accuracy. COMAR [34] used both publicly and non-publicly available data to distinguish between compromised and maliciously registered domain names and concluded that when removing the non-publicly available passive DNS, it achieves an accuracy of up to 97%.

In Section 3.2, we critically revisit relevant features and select those that do not use privileged or commercial data sources.

a DNS reputation system will work, and how wide and easy it can be adopted by the DNS community. DNS reputation system data sources can be either publicly or non-publicly available data sources. Using non-publicly available data sources such as Passive DNS data acquired from specialized sensors, BGP data, or commercially sold data, etc, as implemented in *Exposure* [7], *Notos* [4], *Predator* [20] has limitations as its usage is limited to only those who have access to such privileged data and also verifying such a system's accuracy may be difficult. On the other hand, using publicly available data sources, are not limited to these challenges as seen with non-publicly available data sources. As claimed by *Mentor* [24], *Domain classifier* [29], *COMAR* [34] *APWG* [3], using these public data sources gave high accuracy scores. Furthermore, *De Silva et al* [10] combined both public and non-public (Passive DNS data from *farsight* [14]) data sources to get the highest score of 97.7%. However, the authors of *COMAR* [34], used both publicly and non-publicly available data and came to a conclusion that when they removed the non-publicly available data sources, they did not lose much accuracy as COMAR claims a 97% accuracy rate.

2.2 Feature Importance

Feature importance refers to techniques that assign a score to input features (e.g., domain name popularity, domain name age, etc.) based on the extent to which they contribute to the prediction of the target variable (e.g., classification of benign versus maliciously registered domain names). Ranking features according to their importance shows which features are irrelevant and can be omitted. It reduces the dimensionality of the model, its complexity, the need to collect data, and makes it possible to estimate the impact of missing features on the system.

In the DNS reputation systems we reviewed, only Hao et al. [20], Maroofi et al. [34], and Le Pochat et al. [31] documented feature importance of the proposed models. Note that even the most important feature, if it is missing from the dataset (and its value cannot be estimated), cannot contribute to the prediction of the target variable. Therefore, we analyze the extent of missing values resulting from measurement errors in Section 3.4, discuss feature importance in Section 3.6, and show how missing values of selected features affect the classification of domains using scatter plots.

2.3 Ground Truth

The reviewed systems use classifiers to distinguish between malicious, compromised, and benign domain names. The quality and quantity of ground truth data largely determines the ability to train and evaluate a classifier correctly. Table 1 shows different approaches to building ground truth datasets used in previous work. Some of them rely on third-party services such as Google Safe Browsing (GSB) [18], PhishLabs [41], McAfee SiteAdvisor [43], or Alexa [2], some leverage publicly available datasets created by other work [8]. Some others create their ground truth datasets by manually labeling domain names.

Table 1: Comparison of ground truth datasets in different DNS reputation systems (T: Total, M: Malicious, B/C: Benign or Compromised).

Proposed system	**Ground truth source**	**T**	**M**	**B/C**
Predator [20]	McAfee SiteAdvisor [43], Spamhaus [44] URIBL [50] internal spam trap	769,464	512,976	256,488
Domain Classifier [29]	PhisLabs [41], DeltaPhish [8]	10,150	9,475	675
De Silva et al. [10]	manual labeling	3,278	1,889	1,389
COMAR [34]	manual labeling	2,329	1,199	1,130
Mentor [24]	GSB [18], `malwaredomains.com`, `malwaredomainlist.com`, Alexa [2]	1,430	930	500
Notos [4]	SURBL [46], Alexa [2]	–	–	–

The advantage of third-party services is their availability, ease of use, and timeliness. However, researchers do not have full insight into the proprietary methods used by third-party vendors to label the data. Therefore, such datasets cannot be fully trusted. For instance, Le Page et al. investigated phishing URLs that, according to PhishLabs, were most likely using compromised domains and found instances of obviously maliciously registered domains [29]. Another approach is to use GSB to generate ground truth data. Since the ultimate purpose of GSB is to protect end users from accessing malicious content regardless of the domain state, the dataset cannot be used 'as is' to label malicious registrations.

With manual labeling, researchers carefully select the source and methodology for such data. However, it requires expert knowledge and a considerable amount of time. In some cases, the labeling process is not trivial (the expert is unable to make a reliable assessment of the maliciousness of a domain name). It can introduce inaccuracies (if the domain is incorrectly labeled) or biases (if such corner cases are skipped and not included in the model). Furthermore, its time-consuming nature often discourages researchers from updating ground truth data and retraining models. A similar problem can arise when using data from previous work—it may be outdated, and thus, it may not include evasion techniques recently used by attackers. Therefore, in Section 3.1, we propose a novel approach to automatically generate ground truth data for such systems. It consists of measuring the mitigation actions on abusive domain names by TLD registries, registrars, and hosting providers and can be applied to different domain classification problems.

2.4 COMAR System

We validate the proposed approaches with a prototype that extends COMAR (COmpromised versus MAliciously Registered domains) [34]—a domain name classification system that distinguishes domain names from blacklisted URLs as compromised or maliciously registered with an accuracy of 97%. It consists of three modules: a data collection module, a feature extraction module, and a classification module. The data collection module acquires data on domain names from phishing and malware delivery blacklists. The feature extraction module extracts 38 features, grouped into seven categories: lexical features, ranking and popularity features, passive DNS features, content-based features, WHOIS and TLD-based features, TLS certificate features, and active DNS features. The classification module uses a trained Logistic Regression model to predict the output class (compromised or maliciously registered domain name).

We have chosen COMAR as it combines new features with those proposed by earlier systems, demonstrates high accuracy, and we have access to its implementation. In contrast to its initial design and performance evaluation, we train multiple models on automatically generated ground truth data to account for missing values and extensively evaluate its performance. Finally, we apply COMAR to unlabeled data and present selected statistics for domain names extracted from phishing URLs over a 20-month period.

3 Methodology

In this section, we discuss in detail the methodology to generate ground truth data automatically, the prototype implementation of the classifier, and the practical approach to overcoming the problem of missing values.

3.1 Automated Generation of Ground Truth

The automated ground truth generation method takes advantage of the type of mitigation actions undertaken by the relevant intermediaries involved in domain registration and hosting. After a domain name is involved in malicious activity and the abusive URL is blacklisted or reported to the operator, the TLD registry or registrar must first collect evidence of whether the domain has been maliciously registered or compromised before deciding on the type of mitigation action. A malicious domain name can be blocked at the DNS level. In contrast, a benign and later hacked domain name cannot be blocked without interrupting benign services related to the domain name. In this case, the webmaster or hosting provider (possibly a reseller) should only remove the malicious content from the server, such as a malware download or a phishing site, and patch the vulnerable application to prevent future intrusions. Based on these generally accepted mitigation practices [11], we design the measurement setup to automatically distinguish between compromised and malicious domains.

Maliciously registered domains. The most common mitigation action for a malicious domain used by registries or registrars consists of removing the domain name from the zone, which makes the domain effectively nonexistent (`NXDOMAIN`). While technically this procedure is sufficient to make the domain name and hosted services inaccessible via the public DNS, it is also essential to prevent re-registration of the domain name at the registry/registrar level. Therefore, it is necessary to change its registration status through the Extensible Provisioning Protocol (EPP) [21] indicating that the domain name is not only taken down but unavailable for any change. This effect is achieved by setting the EPP domain registration status code to `clientHold` (set by the registrar) or `serverHold` (set by the TLD registry) [22].

To generate a list of maliciously registered domain names automatically, we collect registration information using either the Registration Data Access Protocol (RDAP) [38] or WHOIS [9] protocols for domains that appeared in the Anti-Phishing Working Group (APWG) [3] or PhishTank [49] URL blacklists between January 2021 and September 2022. We extract the creation and expiration dates of the domain name, and the EPP status codes. Six months later, we again collect the registration information data for domains expected to be active (i.e., the expiration date is after the date of the second measurement). A recent study shows that the uptime of malicious domain names (i.e., the time between URL blacklisting and mitigating abuse) in all TLDs does not exceed three months [5]. We select the conservative interval between two measurements to six months to ensure that relevant intermediaries have enough time to identify abuse, assess the maliciousness of the registered domain name, and proceed with the appropriate mitigation action. If the EPP status code of the domain name is `clientHold` or `serverHold`, we automatically label such a domain as maliciously registered and conclude that the accredited registrar or TLD registry has suspended the domain name. Note that Alowaisheq et al. [1] excluded domain names with one of the two hold status codes and `pendingDelete`, `redemptionPeriod`, or `autorenewPeriod` in their algorithm for identifying domain delisting. However, we argue that these status codes should not appear before the expiration of benign domains. We analyzed our ground truth data set and found only 2 out of 12,179 records flagged with one of the three status codes alongside the hold status. After a manual investigation of these samples registered at two different registrars, we found that both were maliciously registered and were in `redemptionPeriod` while not yet expired.

Finally, it is important to verify that the domain name creation date for both measurements (i.e., at the time of blacklisting and six months later) remains unchanged to ensure that the mitigation action is related to the activity of the original registrant. If it is not the case, it might be possible that such a domain was blacklisted, removed from the zone, later became available for registration, and re-registered.

Compromised domain names. To generate the ground truth dataset for compromised domain names, we first use browser emulation to collect the content and the title of the index page hosted at the root directory of the apex domain and the corresponding URL reported by APWG and Phistank at the time of blacklisting. For instance, for a given URL `https://a.example.com/_boa/login.html`, we visit the index page of the registered domain name (`https://example.com`) and keep track of the HTTP status code and the title of the webpage. We deliberately choose to visit the index page hosted at the registered domain name rather than at the subdomain level. As the DNS-level take-down actions target registered domain names, our system does not consider content hosted on subdomains relevant for assessing the maliciousness of the registered domain.

Six months later, we fetch the content of the originally blacklisted URL using browser emulation. We only keep URLs returning a `404` HTTP status code, i.e., the pages whose content is not available anymore (it was taken down). Note that the use of browser emulation is important at this stage as some malicious websites use bot evasion techniques [52] and we need to eliminate the URLs that seem to be unavailable as the result of cloaking. During this measurement, we also extract the title of the index page of the registered domain. We choose the domains whose index page is available (HTTP status code `200`) and whose title stayed unchanged for six months. We are aware that such an approach is conservative; however, it can only lead to a decrease in the size of the ground truth dataset. We observed many cases where the content hosted at the URL was taken down at the web hosting level (e.g., webpage indicating the website was not found), but the status code of the HTTP response remained `200`. The combination of these conditions guarantees that the malicious content had been taken down by the webmaster or hosting provider but the domain webpage stayed intact as it represents the benign part of services served under the domain name and thus indicating that the website was compromised.

After excluding public apex domains belonging to legitimate services such as URL shorteners,[7] dynamic DNS, or subdomain providers,[8] we have identified 3,632 compromised and 12,179 maliciously registered domain names. One of the reasons for the imbalance between the two datasets is the conservative approach we have chosen for labeling compromised domains. We have decided to inspect whether the title of each domain homepage has changed within six months, excluding all domains that modified their titles. For instance, we have observed changes in which webpage administrators prepend/append characters to the titles of the benign domain names: `Example - Homepage` became `### Example - Homepage ###` and thus, the page was automatically excluded from the compromised dataset even though the title still contains the original string.

3.2 Feature Selection

We next critically revisit 38 proposed features originally used by the COMAR system [34], exclude features that use privileged or unavailable data and remove irrelevant features.

1. Bing search engine results. As discussed by the authors of the COMAR system, it is a paid service, therefore, we exclude it.
2. Features depending on passive DNS. The access to passive DNS data is privileged and related features proved to have a negligible impact on the performance [34].
3. TLD maliciousness index.[9] It is calculated by Spamhaus [44] and is not available for commercial use.
4. The relationship between the domain name and the hosted content. Original COMAR extracts keywords from the domain name and generates their synonyms using a commercial API. They then determine if the domain name is related to its content based on the occurrence of the keywords and their synonyms in the text of the home page. Since the API is not publicly available at the time of writing, we decided to remove this feature.
5. Quantcast ranking system is not publicly available anymore.
6. TLS certificate price. It is not trivial to distinguish between free and paid certificates since some certificate authorities (e.g., Comodo CA1) offer both paid and free certificates.
7. Presence of a TLS certificate. We exclude the presence of the Transport Layer Security (TLS) certificate from our features since the use of TLS certificates among malicious and benign but compromised domains used in phishing is comparable (see Section 4.1).
8. Valid TLS certificate. For shared hosts, if a certificate is not valid (e.g., wrong host error), we cannot conclude if the malicious actor issued a wrong certificate or if the certificate belongs to another domain on the same (shared)

[7] https://github.com/korlabsio/urlshortener

[8] https://github.com/korlabsio/subdomain_providers

[9] https://www.spamhaus.org/statistics/tlds/

Table 2: List of selected features with their corresponding feature sets.

F	Feature	Description	F-set	Set Name
F1	digit_ratio	Number of digits over the length of the domain name	FS1	Lexical
F2	has_famous_brand_name	If the domain name contains a famous brand name	FS1	Lexical
F3	level_of_subdomain	Number of subdomains in the fully qualified domain name	FS1	Lexical
F4	special_keywords	If there is a special keyword used in the domain name	FS1	Lexical
F5	num_hyphen	Number of hyphens used in the domain name	FS1	Lexical
F6	diff_create_blacklist_time	Difference between domain creation and blacklisting time	FS2	WHOIS
F7	content_length	Content length of the homepage of the domain name	FS3	Content length
F8	has_index_page	Default webserver index page?	FS5	Index page
F9	is_use_redirection	If there is a redirection to another domain	FS4	Home page redirect
F10	is_default_homepage	If there is a default installation of a famous CMS	FS5	Index page
F11	has_vulnerable_tech	If there is a vulnerable technology (e.g., WordPress) used	FS6	Technologies
F12	num_of_tech	Number of distinct libraries used in the homepage	FS6	Technologies
F13	is_self_resolving	The domain name is self resolving	FS7	DNS (Self resolving)
F14	is_in_alexa	If the domain name is in the Alexa list	FS8	Alexa
F15	num_internal_hyperlinks	Number of working internal hyperlinks on the homepage	FS9	Hyperlinks
F16	num_external_hyperlinks	Number of working external hyperlinks on the homepage	FS9	Hyperlinks
F17	num_captured_wayback	Number of saved pages in the Wayback machine	FS10	Wayback machine

hosting service. Therefore, this feature is not suitable for operational deployments.

9. TLD price. The TLD price is not unique among all registrars and resellers, and changes over time. In addition, special offers from registrars or domain resellers can drastically reduce the price for a specific TLD. It is also difficult to collect such data at scale.

Based on this analysis, we remove 13 features and train the model with the remaining 25 features using Logistic Regression. We analyze the coefficients and remove features that are not important for the model. We present the final set of the 17 remaining features in Table 2. Note that the *is_in_alexa* (F14) feature was only available before the termination of service announced by Alexa in May 2022.[10] Since this feature is unavailable for only three of the twenty months of phishing data collected, we keep it and use the method of handling missing values as explained in Section 3.3. However, it can be replaced by the Tranco top sites ranking [30] in future work. For *has_famous_brand_name* (F2), we used the list of target brand names provided by PhishTank. We consider this binary feature to be true if a domain name contains one of these trademarks. We use a similar method to the one proposed by Kintis et al. [25]. However, as our work does not only focus on combo squatting, we consider this feature to be true even for some of the five typosquatting models of Wang et al. [51] (e.g., the value of this feature for domain `facebookk.com` with trademark `facebook` would be true even if it would not be marked as *combosquatted* by the method proposed by Kintis et al. [25]).

Table 3: Percentage of missing values for each feature.

Name	F-set	Missing %
digit_ratio	FS1	0.00%
num_hyphen	FS1	0.00%
special_keywords	FS1	0.00%
level_of_subdomain	FS1	0.00%
has_famous_brand_name	FS1	0.00%
is_self_resolving	FS7	0.00%
num_captured_wayback	FS10	1.42%
is_in_alexa	FS8	15.18%
content_length	FS3	15.22%
num_of_tech	FS6	15.35%
has_vulnerable_tech	FS6	15.35%
diff_create_blacklist_time	FS2	22.29%
is_use_redirection	FS4	26.27%
has_index_page	FS5	26.53%
is_default_homepage	FS5	26.53%
num_internal_hyperlinks	FS9	47.53%
num_external_hyperlinks	FS9	47.53%

3.3 Measuring the Extent of Missing Values

Regardless of the importance of a feature, if it cannot be collected, it cannot contribute to the prediction of the target variable. Therefore, we first evaluate the occurrence of missing values per feature for the unlabeled dataset (see Section 4 for more details). Only 36.5% of domain names have a complete feature vector (i.e., have no missing values). Table 3 shows the percentage of missing values per feature. Some features are always available as they do not depend on any measurements (i.e., lexical features such as domain name digit ratio or the number of hyphens in the domain name) or whose measurements are generally easy to perform such as DNS-related features. However, some features suffer from missing values either due to measurement or parsing errors, or the unavailability of data.

For instance, the difference between domain creation (registration) and domain blacklisting time, also referred to as domain name age [34] (F6), is a feature derived from WHOIS/RDAP data and is missing for 22.29% of domain names. However, for some TLDs (e.g., `.de` TLD), there is no information about the domain registration date in WHOIS. For other TLDs, it is not feasible to collect WHOIS information at scale since either there is no conventional WHOIS server (e.g., for `.gr` TLD) or the access is restricted to authorized IP addresses (e.g., `.es` TLD). Moreover, extracting WHOIS information relies on manual creation of parsing rules and templates for individual registrars, and is by nature lim-

[10] https://support.alexa.com/hc/en-us/articles/4410503838999-We-retired-Alexa-com-on-May-1-2022

ited in scope and susceptible to changes in data representation [32]. The RDAP protocol [38] overcomes the problem of parsing but it is not universally deployed.

While the features related to the page content play an essential role in classification [34], our results show that the values for these features are often missing (up to 47.5% for Hyperlinks). The reason is that data collection related to web content requires significant resources (e.g., browser emulation in our case) and its results highly depend on the page load time and implementation. For instance, poorly maintained websites may result in timeout or measurement errors. Domain redirection is another reason for the missing values of content-related features. URLs that use domain redirection (i.e., HTTP 3XX status code or JavaScript redirection) will load the content of the destination domain name (i.e., different from the original domain). In such cases, we consider these features as missing.

3.4 Handling Missing Values with Multiple Models

Applying simple techniques to handle missing values such as median or mean imputation might generate biased results [12]. Maroofi et al. [34] proposed an imputation method that infers the missing value of one feature from others. For example, a domain name age (i.e., the difference between domain creation and URL blacklisting time) that can be estimated based on the first appearance in the Internet Archive [23], Google Certificate Transparency logs [17], and privileged passive DNS data. However, such an approach cannot be applied to all types of features as many of them are independent of each other (e.g., the number of external hyperlinks cannot be estimated using other features).

To handle missing values, we propose a new approach—we design a multiple-model system that makes use of models trained on different combinations of features. The idea is to generate a large number of models, each dedicated to handling records with missing values of a specific subset of features: if we need to classify a domain name with missing values for features X and Y, we can use the model trained without features X and Y thus allowing classification based on all other features.

Based on our observations of missing values shown in Table 3, we group the features into 10 different sets as illustrated in Table 2. The features in each set are either all available or all missing. For example, the Lexical feature set (FS1) contains 5 features (F1: digit ratio, F2: has famous brand name, F3: level of subdomain, F4: special keywords, F5: number of hyphens). They are always available since they do not rely on active measurements or external third-party services. As soon as the system receives the input URL, it can generate these features. However, the Technologies feature set (F11: has vulnerable technology, F12: number of technologies) heavily depends on the availability of the HTML content of the domain homepage and HTTP headers. Note that our method of grouping features into feature sets is supported by the empirical assessment of missing value rates shown in Table 3, as opposed to Maroofi et al. [34] who grouped features into feature sets only based on their categories.

With eight feature sets with possible missing values (FS2-FS10), we calculate the number of models to be trained using the following formula:

$$number\ of\ models = \sum_{n=0}^{8} \binom{9}{n}, \tag{1}$$

where n is the number of removed feature sets, and 8 is the maximum number of removed feature sets. We create 511 models.

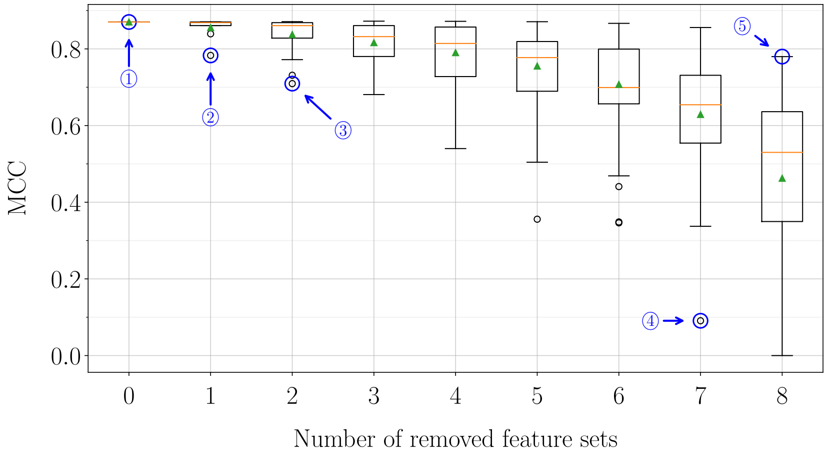
Based on the results of our previous work and the successful implementation of the Logistic Regression method in the operational COMAR system, we use the same classification method. LR uses a combination of weighted input feature values to predict output probabilities, making it easier to interpret (especially when considering multiple models) and assess the maliciousness of registered domains based on the most significant features. Therefore, its interpretability is not only important at the design stage for system tuning and evaluation. It can also play an essential role for operators at helpdesks who need a good understanding of classification results and the underlying models.

Therefore, we train the 511 models using Logistic Regression and use two sources of ground truth data. First, we use automatically generated data using methods described in Section 3.1 and refer to it as *Ground Truth 1* (GT1). This data represents labeled real-world samples of domain names with possibly missing values as quantified and detailed in Section 3.3. As the second source, we use manually labeled data provided to us by Maroofi et al. [34]. This dataset has no missing values and can bring corner cases into the training and testing. We refer to it as *Ground Truth 2* (GT2).

We use these two ground truth data sets in the following way. We train distinct models in 511 iterations. Each iteration represents a subset of feature sets after removing one to eight chosen feature sets at a time. We then use the feature vector of the iteration to train and evaluate each model. For instance, the feature vector of one of the iterations covers the feature sets without WHOIS (FS2), i.e., only FS1 and FS3 to FS10 are used. We train the complete model using the records with a complete feature vector from both GT1 and GT2. For other models with removed feature sets, we use the domain names from GT1 and GT2 in which all values of the remaining features are present.

Training of 511 models takes approximately 20 minutes on a personal computer (Intel Core i5-8265U CPU @ 1.60GHz, 16GB RAM), which may increase the cost compared to systems based on one model only, but overall, it remains low. More importantly, the cost of training is mainly related to the time required to generate ground truth, which is low compared to manual labeling. Therefore, the here-proposed method significantly reduces the overall cost and can label more samples for training. Given the automated approach for ground-truth generation, we could consider regular active learning. However, it would require future work to evaluate if static models exhibit high performance over time.

Fig. 1: Boxplot showing the MCC of models grouped by the number of removed features at a time. Triangles and horizontal lines represent the mean and median of MCC for each group of models, respectively.



3.5 Performance Evaluation

To evaluate the models, we used Stratified K-fold Cross-Validation (SKCV) [26] with $K = 10$. While the standard K-fold Cross-Validation splits the dataset into the training and testing data in each iteration randomly by the predefined ratio, SKCV ensures that each fold keeps the same proportion of classes (i.e., malicious and compromised labels in our case) as in the original distribution. As our ground truth dataset is imbalanced, we have chosen this method to evaluate the models more accurately.

During the following evaluation, we use common metrics to evaluate the performance of models. For details, we refer the reader to Appendix A. Figure 1 shows a boxplot summarizing the distribution of Matthews Correlation Coefficient (MCC) for the 511 models. The MCC of the full model (①) is 0.87 with a 93.67% accuracy. The model without the `WHOIS` feature set (②) is the most significant outlier (MCC: 0.78, FNR: 14.2%) for the models with one removed feature set at a time. This result confirms that the domain age at the time of blacklisting is a strong feature and its absence causes a significant decrease in performance. Similarly, the model without the FS2 and FS10 feature sets (③) is the most significant outlier (MCC: 0.71, FNR: 17.9%) for the group of models with two removed features at a time. If we remove the `WHOIS` (FS2) and `Wayback machine` (FS10) feature sets, the performance of the model is highly impacted. As expected, one of the worst models ④ (MCC: 0.09, FNR: 98%) lacks all previously discussed feature sets (FS2, FS10), but also the remainder of the important features (FS3, FS5, FS6, FS8, and FS9). We discuss the implications of these findings for operational classification later in this section.

Table 4: Distribution of top ten models (combinations of removed feature sets) with the highest coverage of samples in the unlabeled dataset.

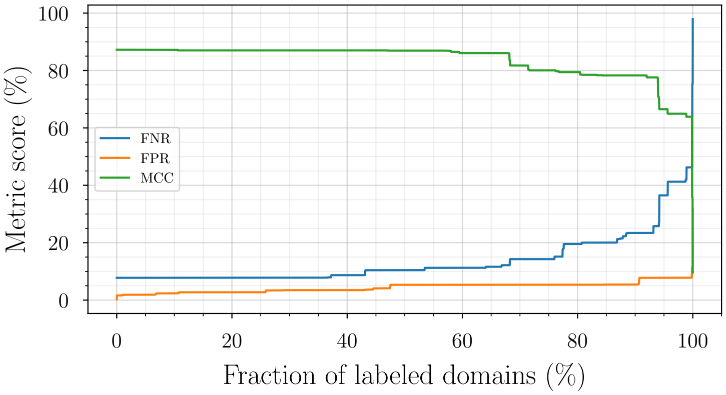
Missing F-sets	Coverage (%)	Model MCC	Model FNR
None	36.5	0.87	0.08
FS4, FS5, FS9	10.5	0.87	0.11
FS9	10.2	0.87	0.10
FS2	7.7	0.78	0.14
FS8	5.8	0.86	0.09
FS3, FS4, FS5, FS6, FS9	4.6	0.80	0.23
FS2, FS4, FS5, FS9	3.7	0.79	0.20
FS2, FS3, FS4, FS5, FS6, FS9	3.1	0.65	0.41
FS3, FS6, FS9	3.1	0.82	0.20
FS8, FS9	2.7	0.86	0.12

Note that even if MCC is a suitable method to evaluate the performance of binary classification with an unbalanced distribution of classes, it is still essential to consider other metrics, such as the false negative and false positive rates. For instance, we carefully monitor the false negative rate to avoid compromised domains incorrectly classified as malicious, which may lead to the blocking of a benign domain causing collateral damage to the legitimate registrant. Therefore, model ⑤ that only uses the domain name age (F6) calculated based on WHOIS has a high MCC (0.78) but it has to be used with caution as its FNR is high (26.1%).

As described in Section 3.3, our system consists of models trained on combinations of incomplete feature vectors and handles the classification of domain names with missing values. Table 4 shows ten of such combinations that appeared the most in our unlabeled dataset. For instance, 10.2% of domains that have missing values for FS9 (`Hyperlinks`) can still be classified using a model with good overall performance (MCC: 0.87, FNR: 10%), similar to the model with no missing values. We observed that 36.5% of domain names in our unlabeled dataset (see Section 4) do not have any missing values and therefore, they can be classified using the complete model with all 10 feature sets present. The remaining 63.5% of domain names have at least one missing value and cannot be classified using a single-model approach (assuming that other methods to handle missing values are not implemented). The 501 remaining models cover 12.1% of samples. These models are necessary for the system to handle missing values resulting from measurements related to each feature set.

A system that can be used by investigators should offer a way of tailoring it to different use cases. If an investigator needs precise classification at the expense of an increased number of remaining domain names that need to be manually verified, she could only use models with good performance metrics such as MCC ≥ 0.85 and FNR $\leq 10\%$ covering 43.2% of unlabeled data. However, if the investigator needs the classification results for informative purposes only (to observe general trends regarding abusive domain names), she can choose

Fig. 2: Empirical cumulative distribution function of performance metrics.

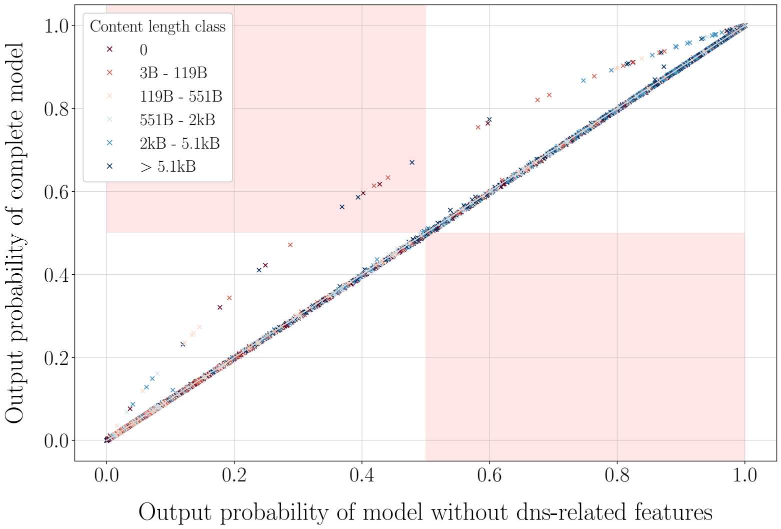


more relaxed requirements (e.g., MCC $\geq$ 0.7 and FNR $\leq$ 20% covering 80.6% of the dataset). Therefore, we propose a systematic approach for selecting the models and metrics based on the desired coverage of unlabeled data and performance, for example, by DNS operators to support their anti-abuse practices. Figure 2 shows the dependency between the chosen metrics and the fraction of automatically classified domains. While FPR stays below 10% for all 511 models, the percentage of automatically labeled domains has a more significant impact on FNR. If investigators want to label 100% of domain names, only 0.01% of domain names will be classified using the worst model with 97.9% FNR. If the coverage of 80% of the dataset is required, the worst model will suffer from 20% FNR. For the results presented in Section 4, we choose models with MCC$\geq$ 0.8 resulting in 76.3% of labeled domain names.

3.6 Feature Importance

Hao et al. [20] and Maroofi et al. [34] assessed the feature importance by excluding feature sets one by one from the system and comparing the calculated metrics of these models. In this section, we present a post hoc method for fine-grained visualization of the feature importance using scatter plots based on similar principles as in their methods. Out of the 511 models, we select those trained with only one feature set missing (9 models as we do not consider lexical features as possibly missing). We choose a sample of 10,000 domain names with no missing values and we classify them first with the complete model and then with 9 models with one removed feature at a time. We present the results for selected feature sets in Figures 3 – 4. Each point in the graph represents one domain name. The y-axis is the predicted probability of a domain being compromised when classified with the complete model. The x-axis is the probability predicted by a model without one of the feature sets. Each point is colored based on the category (content length or domain age). The output probability of points laying

Fig. 3: Scatter plot of probability changes between the full model and the model without DNS-related features (FS7).

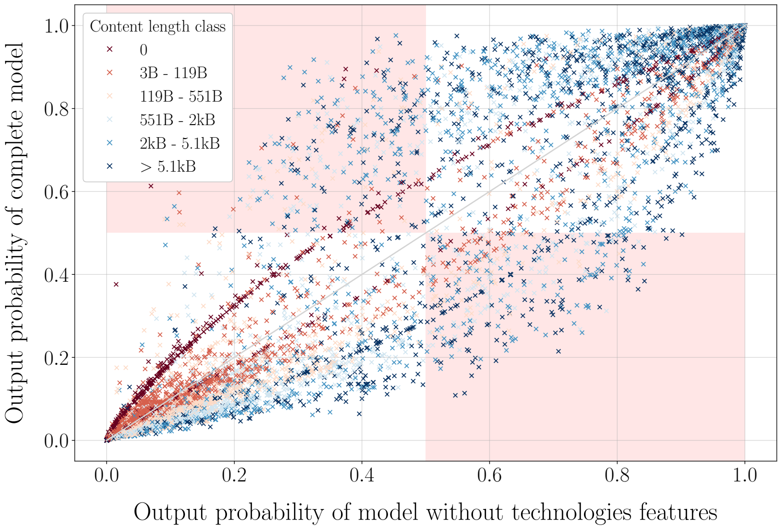


on the line $x = y$ remained unchanged after a feature set elimination. Points for which $x > y$ (increase in output probability), became "more compromised" after feature set removal. Similarly, points for which $x < y$ (decrease in the output probability), became "more malicious". Note that the red zone at the top left and bottom right corner highlight the points that could potentially change labels. For instance, Figure 3 shows that the output probability of a small fraction of domains was impacted by removing the DNS-related features (FS7). Therefore, this feature set does not have a high impact on the classification results.

On the other hand, Figures 4, 10, and 11 (in Appendix B) demonstrate that feature sets FS2 (`WHOIS`), FS8 (`Alexa`), and FS9 (`Hyperlinks`) strongly influence the output of our system as many data points moved horizontally after eliminating a feature set (i.e., became more malicious or more compromised).

However, it is important to note that especially the most important features can lead to misclassification if manipulated by attackers. An attacker may generate long content, deploy multiple technologies on the index page of a registered domain name, or avoid registering domain names with special keywords. However, manipulating COMAR features also requires additional effort and can be costly. Since we based our prototype on the original COMAR system, the individual features share the same characteristics regarding robustness and possible evasion. We refer interested readers to our initial study for a detailed discussion of possible evasion techniques for each feature [34].

Fig. 4: Scatter plot of probability changes between the full model and the model without feature set FS6 (web technologies).



3.7 Ethical Considerations

To collect data, we perform active measurements, particularly browser emulation via HTTP requests and active DNS lookups. Ethical issues that may arise mainly concern the possible overloading of the scanned infrastructure. To address this issue, we limited the number of simultaneous scans to twenty during the browser emulation phase, representing negligible traffic that should not significantly affect web servers. We used the Google public DNS resolver for DNS scans, adhering to the restrictions specified in the official documentation.[11]

4 Classification Results

In this section, we apply the prototype classifier to 218,806 unlabeled unique domain names from APWG [3], OpenPhish [39], and PhishTank [49] URL blacklists collected between January 2021 and September 2022. We study four selected characteristics of the domain names of malicious URLs and analyze their distribution across different types of TLDs.

The overall classification results show that 73% of phishing domain names were registered for malicious purposes, and 27% were classified as registered by benign users but have been compromised. If the domain names were compromised at the hosting rather than at the DNS level, they should not be blocked by TLD registries or registrars.

[11] https://developers.google.com/speed/public-dns/docs/isp

Fig. 5: Top level domain distribution.

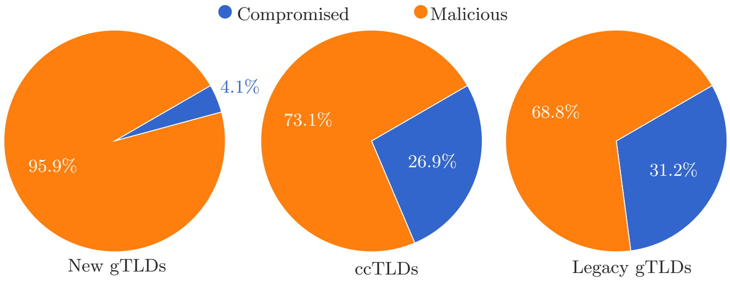


Figure 5 shows that almost 96% of domain names of blacklisted phishing URLs in new generic TLDs (e.g., `.top`, `.pharmacy`, `.xyz`) are likely to be maliciously registered, 69% for legacy gTLDs (e.g., `.com`, `.net`, `.org`), and about 73% for country-code TLDs (e.g., `.br`, `.no`, `.jp`). The question arises: why is the fraction of domains registered for malicious purposes in new gTLDs compared to compromised ones much higher than in ccTLDs and legacy gTLDs? Previous studies [19,28] showed that, in general, for new gTLDs, a relatively large proportion of domain names are either parked or contain no content (DNS or HTTP errors) compared to legacy gTLDs. Intuitively, only domain names containing content are likely to be vulnerable to certain types of exploits and thus can be exploited at the website level. It might be a plausible explanation for why only a tiny fraction of domain names of new gTLDs are likely to be compromised. However, this hypothesis requires systematic future research because no recent studies have conducted such a comparative analysis.

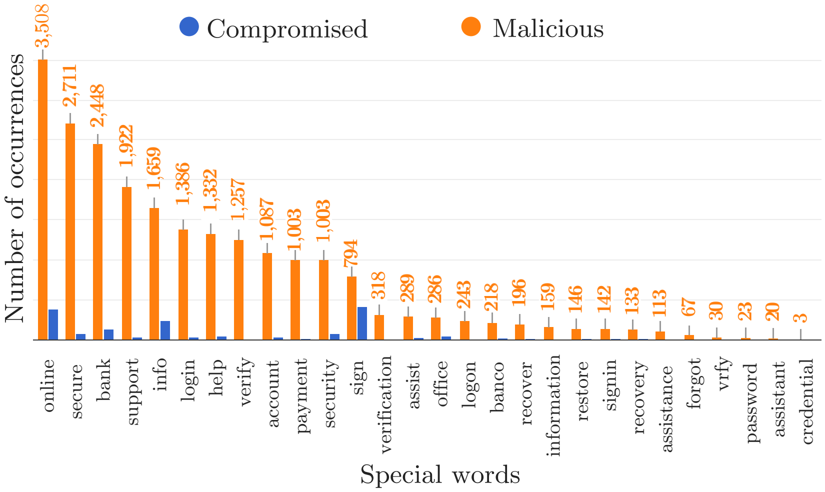
The presented results should be merely seen as trend indicators and may be influenced by the blacklist bias as well as short-term trends in the choices made by attackers. For example, some blacklists may be more effective in detecting maliciously registered domain names (e.g., based on suspicious keywords), while others may be more effective in detecting compromised sites. Some domain registrars, accredited by a TLD registry, may offer low registration prices for a short period to attract new customers. Malicious actors may take advantage of such special offers and register domain names on a large scale, which may affect the observed percentage of compromised and maliciously registered domains.

4.1 Analysis of the Selected Features

We now explain how the compromised and maliciously registered domain names distinguished by our system differ in terms of four selected features: popular terms in domain names, the number of web technologies used, the domain name age, and the use of TLS certificates.

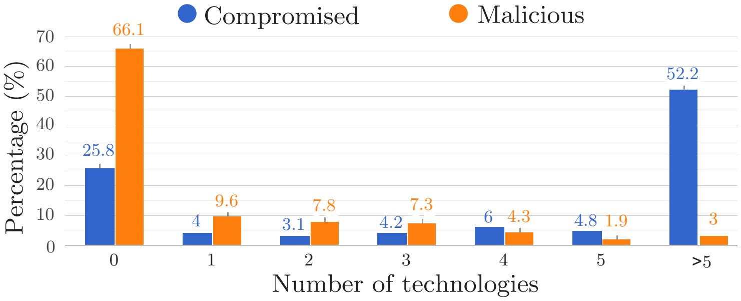
The features indicating that a cybercriminal (rather than a benign user) has registered a domain name include specific keywords such as ‘verification’, ‘payment’, ‘support’, or brand names (e.g., paypal-online-support.com). Figure 6

Fig. 6: Distribution of popular keywords in domain names of compromised and maliciously registered domains.



presents a word frequency analysis of the phishing dataset for both domain names automatically classified as maliciously registered (orange) and those classified as compromised (blue).

Fig. 7: Distribution of the number of technologies between compromised and registered domains.

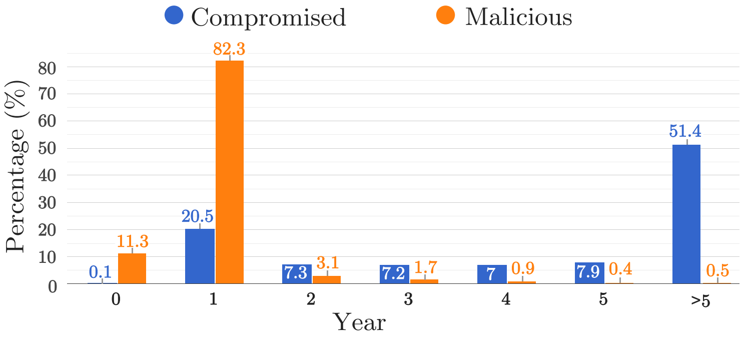


We can observe that cybercriminals tend to incorporate such words into domain names to lure victims into entering their credentials. The most frequently used keywords by malicious actors are ‘online’, ‘secure’, ‘bank’, ‘support, ‘info’, ‘login’, and ‘help’. On the other hand, the domain name of compromised sites rarely contains such specific keywords.

One of the used features is the *number of web technologies* (F12): a count of the JavaScript, Cascading Style Sheets (CSS), or Content Management System (CMS) frameworks and plugins used to build the *homepage* of a registered domain name. The higher number of technologies used for developing a website could reflect the amount of effort and time its designer spent to create a fully-functional website. While this is true for benign (compromised) domain names, malicious actors tend to put little effort into deploying multiple technologies when designing websites on maliciously registered domain names, as it is not critical to the success of phishing attacks. Figure 7 shows the results for compromised and maliciously registered domain names. As many as 52.2% of compromised domains use more than five different (potentially vulnerable) technologies, frameworks, and plugins to build the website. In comparison, 66.1% of the maliciously registered domain names have no specific technology on their homepage. We have noticed that many maliciously registered domains either have no homepage (showing the default directory index served by the web server), redirect to another domain (e.g., the landing page of a phishing attack), or display a custom error message (e.g., forbidden page). Instead, they frequently serve the phishing page either on a URL path or a subdomain level.

The age of a domain name (F6), defined as the time between the registration of the domain name and its appearance on the blacklist, is one of the important features of our classifier. Intuitively, the older the domain name, the more likely it is to have been registered by a benign user but subsequently compromised. On the other hand, cybercriminals tend to use a domain name for malicious activities soon after registration. Figure 8 shows the age of domain names for all TLDs that provide the registration date as part of their WHOIS data: "0" means that registration and blacklisting occurred on the same day, "1" – the difference between the registration date and the blacklisting date is at most one year, and "> 5" means that the difference between the domain registration date and the blacklisting date is greater than five years. For 93.6% of maliciously registered

Fig. 8: Distribution of domain ages between maliciously registered and compromised domain names in percentage.



domain names, the difference between the domain registration date and the blacklisting date is less than a year, and for 11.3% of them, the domains were blacklisted on the same day the domain was registered. For compromised domain names, about 51.4% of them were registered at least six years before being blacklisted. A possible explanation for this phenomenon is that websites hosted on older domain names are more likely to use outdated technologies or content management systems (e.g., vulnerable versions of CMS such as WordPress), making them easier to compromise.

In some sporadic cases, malicious actors may "age" registered domains, waiting weeks or sometimes months before abusing them, or compromise domain names shortly after their registration [34]. However, as our system is fully automated and performs classification based on multiple features (the domain name age is just one of them), it is more resistant to manipulation (e.g., domain aging).

While we explain in Section 3.2 why we avoid using TLS certificate features, we analyze their use by owners of compromised versus maliciously registered domain names. According to a PhishLabs report [40], three quarters of all phishing sites used HTTPS (HTTP over TLS) in 2020 "*to add a layer of legitimacy, better mimic the target site in question, and reduce being flagged or blocked from some browsers.*" However, the report conflates compromised and maliciously registered domain names. Therefore, to establish whether cybercriminals increasingly use TLS certificates, we need to distinguish between compromised and maliciously registered domain names and analyze the use of TLS only in the latter group. Otherwise, it is unclear whether the TLS certificate was issued at the request of a criminal for a maliciously registered domain to enhance the website's credibility or at the request of a legitimate domain owner for a benign domain name that was later compromised and abused by a criminal.

Fig. 9: TLS use by maliciously registered and compromised domain names in percentage.

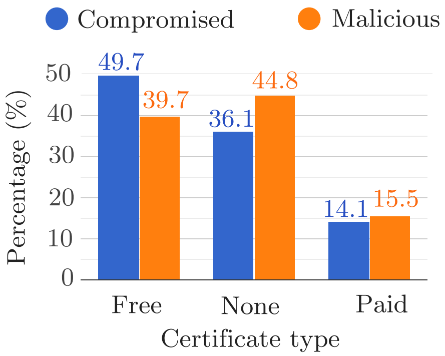


Figure 9 shows the statistics of TLS certificates issued for malicious and benign (and later compromised) domains involved in phishing attacks. The use of

TLS certificates is less widespread among phishers than for benign (but compromised) domain names. 63.9% of phishing attacks using compromised domains take advantage of TLS certificates issued at the request of benign domain owners while 55.2% of maliciously registered domains use TLS certificates deliberately deployed by malicious actors to lure their victims. Surprisingly, 15.5% of maliciously registered domains used most likely paid TLS certificates. We further investigate these domains and find that 48.7% of them had a TLS certificate issued by Sectigo [42]. The majority of these domains were registered with Namecheap [37] which offers a cheap all-in-one hosting package including a domain name registration, hosting, and a one-year valid TLS certificate issued by Sectigo [42].

5 Conclusions and Future Work

Domain reputation systems are of great importance for Internet intermediaries to accurately assess whether domain names are benign, likely to have been maliciously registered, or have been compromised and used to distribute malicious content or ensure the proper functioning of malicious infrastructures. Developing a domain name classification system suitable for deployment in an operational environment requires careful design. To address the shortcomings of some existing systems, we first proposed an approach to automatically generate ground truth data based on mitigation actions undertaken by the relevant intermediaries involved in domain registration and hosting. We carefully selected publicly available features to ensure that our system can be implemented by different actors: from DNS operators and hosting providers to law enforcement agencies. We carefully measured the extent of missing values stemming from measurement and parsing errors, since even the most important features have no real value if they cannot be collected and used in classification. Our results show that for 36.5% of domain names, we can use a complete feature vector to classify domain names. To handle missing values, we proposed a new approach based on multiple models as an alternative to simple statistical techniques. Since the performance of different models varies, we propose a systematic approach to choosing models based on the expected rate of classified domain names and the performance required by DNS operators to support their anti-abuse practices.

We applied the prototype classifier to blacklisted URLs over a 20-month period and explored selected characteristics of abused domain names and their distributions across different types of TLDs. We found that approximately one-quarter of domain names used to launch phishing campaigns are compromised and generally cannot be blocked at the DNS level. The percentage of domains registered with malicious intent to compromised domains in new gTLDs (96%) is much higher than in ccTLDs (73%) and legacy gTLDs (69%). The results also indicate that malicious actors usually put little effort into deploying multiple technologies when designing websites on maliciously registered domain names and typically use them shortly after registration.

The proposed design approaches can be applied to any domain name classification problem, and the designed prototype has demonstrated its utility in an operational environment, as two ccTLD registries have adopted it to support their DNS anti-abuse practices.

We plan to use the proposed classifier to perform a longitudinal analysis on phishing URLs to observe the changes in attackers' behavior over time such as the use of popular keywords in maliciously registered domain names. Finally, since the training cost is low and mainly related to automated truth data generation, we can apply active learning to the proposed system, adapting it to new trends and techniques used by attackers over time.

Acknowledgments

We thank Benoît Ampeau, Marc van der Wal (AFNIC) and the anonymous reviewers for their valuable feedback, Anti-Phishing Working Group, OpenPhish, and PhishTank for providing access to their URL blacklists. This work has been carried out in the framework of the COMAR project funded by SIDN, the .NL Registry and AFNIC, the .FR Registry. It was partially supported by the Grenoble Alpes Cybersecurity Institute (under contract ANR-15-IDEX-02), and the French Ministry of Research (PERSYVAL-Lab project under contract ANR-11-LABX-0025-01, and DiNS project under contract ANR-19-CE25-0009-01).

References

1. Alowaisheq, E., Wang, P., Alrwais, S., Liao, X., Wang, X., Alowaisheq, T., Mi, X., Tang, S., Liu, B.: Cracking the Wall of Confinement: Understanding and Analyzing Malicious Domain Take-Downs. In: Proc. of NDSS (2019)
2. Amazon: Alexa: SEO and Competitive Analysis Software. (2022), https://www.alexa.com/
3. Anti-Phishing Working Group: Global phishing survey: Trends and domain name use in 2016 (2016), https://docs.apwg.org/reports/APWG_Global_Phishing_Report_2015-2016.pdf
4. Antonakakis, M., Perdisci, R., Dagon, D., Lee, W., Feamster, N.: Building a Dynamic Reputation System for DNS. In: Proc. of USENIX Security. p. 18 (2010)
5. Bayer, J., Nosyk, Y., Hureau, O., Fernandez, S., Paulovics, I., Duda, A., Korczyński, M.: Study on Domain Name System (DNS) Abuse: Technical Report. arXiv preprint arXiv:2212.08879 (2022)
6. Bilge, L., Kirda, E., Kruegel, C., Balduzzi, M.: EXPOSURE: Finding Malicious Domains Using Passive DNS Analysis. In: Proc. of 18th NDSS (2011)
7. Bilge, L., Sen, S., Balzarotti, D., Kirda, E., Kruegel, C.: Exposure: A Passive DNS Analysis Service to Detect and Report Malicious Domains. ACM Trans. Inf. Syst. Secur. **16**(4) (apr 2014)
8. Corona, I., Biggio, B., Contini, M., Piras, L., Corda, R., Mereu, M., Mureddu, G., Ariu, D., Roli, F.: DeltaPhish: Detecting Phishing Webpages in Compromised Websites. arXiv:1707.00317 (2017)
9. Daigle, L.: Whois protocol specification. Tech. rep., RFC Editor (Sep 2004)

10. De Silva, R., Nabeel, M., Elvitigala, C., Khalil, I., Yu, T., Keppitiyagama, C.: Compromised or Attacker-Owned: A Large Scale Classification and Study of Hosting Domains of Malicious URLs. In: Proc. of USENIX Security. pp. 3721–3738 (2021)
11. DNS Abuse Framework: https://dnsabuseframework.org/
12. Donders, A.R.T., van der Heijden, G.J., Stijnen, T., Moons, K.G.: Review: A Gentle Introduction to Imputation of Missing Values. Journal of Clinical Epidemiology **59**(10), 1087–1091 (Oct 2006)
13. Emmanuel, T., Maupong, T., Mpoeleng, D., Semong, T., Mphago, B., Tabona, O.: A Survey on Missing Data in Machine Learning. Journal of Big Data volume **8** (May 2021)
14. Farsight Security: Passive DNS Historical Internet Database: Farsight DNSDB (2022), https://www.farsightsecurity.com/solutions/dnsdb/
15. Felegyhazi, M., Kreibich, C., Paxson, V.: On the Potential of Proactive Domain Blacklisting. In: Proc. of 3rd USENIX LEET (2010)
16. Frosch, T., Kührer, M., Holz, T.: Predentifier: Detecting Botnet C&C Domains from Passive DNS Data. In: Zeilinger, M., Schoo, P., Hermann, E. (eds.) Advances in IT Early Warning, pp. 78–90. AISEC (2013)
17. Google: Certificate Transparency, https://certificate.transparency.dev/
18. Google Safe Browsing: https://safebrowsing.google.com/
19. Halvorson, T., Der, M.F., Foster, I., Savage, S., Saul, L.K., Voelker, G.M.: From .Academy to .Zone: An Analysis of the New TLD Land Rush. In: Proc. of IMC. p. 381–394 (2015)
20. Hao, S., Kantchelian, A., Miller, B., Paxson, V., Feamster, N.: PREDATOR: Proactive Recognition and Elimination of Domain Abuse at Time-Of-Registration. In: Proc. of ACM SIGSAC. p. 1568–1579 (2016)
21. Hollenbeck, S.: Extensible Provisioning Protocol (EPP) Domain Name Mapping. RFC 3731, RFC Editor (March 2004)
22. ICANN: EPP Status Codes — What Do They Mean, and Why Should I Know?, https://www.icann.org/resources/pages/epp-status-codes-2014-06-16-en
23. Internet Archive: Wayback Machine: https://archive.org/web/
24. Kheir, N., Tran, F., Caron, P., Deschamps, N.: Mentor: Positive DNS Reputation to Skim-Off Benign Domains in Botnet C&C Blacklists. In: Proc. of 29th IFIP IISC. pp. 1–14. Springer (Jun 2014)
25. Kintis, P., Miramirkhani, N., Lever, C., Chen, Y., Romero-Gómez, R., Pitropakis, N., Nikiforakis, N., Antonakakis, M.: Hiding in Plain Sight. In: Proc. of ACM SIGSAC (Oct 2017)
26. Kohavi, R.: A Study of Cross-Validation and Bootstrap for Accuracy Estimation and Model Selection. In: Proc. of 14th IJCAI - Volume 2. p. 1137–1143 (1995)
27. Korczyński, M., Tajalizadehkhoob, S., Noroozian, A., Wullink, M., Hesselman, C., van Eeten, M.: Reputation Metrics Design to Improve Intermediary Incentives for Security of TLDs. In: Proc. of IEEE Euro SP (April 2017)
28. Korczyński, M., Wullink, M., Tajalizadehkhoob, S., Moura, G.C.M., Noroozian, A., Bagley, D., Hesselman, C.: Cybercrime After the Sunrise: A Statistical Analysis of DNS Abuse in New gTLDs. In: Proc. of ACM ASIACCS (May 2018)
29. Le Page, S., Jourdan, G.V., Bochmann, G., Onut, I.V., Flood, J.: Domain Classifier: Compromised Machines Versus Malicious Registrations, pp. 265–279. Springer International Publishing (04 2019)
30. Le Pochat, V., Van Goethem, T., Tajalizadehkhoob, S., Korczyński, M., Joosen, W.: Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation. In: Proc. of NDSS. Internet Society (2019)

31. Le Pochat, V., Van hamme, T., Maroofi, S., van Goethem, T., Preuveneers, D., Duda, A., Joosen, W., Korczyński, M.: A Practical Approach for Taking Down Avalanche Botnets Under Real-World Constraints. In: Proc. of 27th NDSS (2020)
32. Liu, S., Foster, I., Savage, S., Voelker, G.M., Saul, L.K.: Who is .Com? Learning to Parse WHOIS Records. In: Proc. of IMC. p. 369–380 (2015)
33. Ma, J., Saul, L.K., Savage, S., Voelker, G.M.: Beyond Blacklists: Learning to Detect Malicious Web Sites from Suspicious URLs. In: Proc. of 15th ACM SIGKDD ICKDDM. pp. 1245–1254. KDD (2009)
34. Maroofi, S., Korczyński, M., Hesselman, C., Ampeau, B., Duda, A.: COMAR: Classification of Compromised versus Maliciously Registered Domains. In: Proc. of IEEE EuroS&P. pp. 607–623 (2020)
35. Matthews, B.: Comparison of the Predicted and Observed Secondary Structure of T4 Phage Lysozyme. Biochimica et Biophysica Acta (BBA) - Protein Structure **405**(2), 442–451 (Oct 1975)
36. Moura, G.C.M., Müller, M., Davids, M., Wullink, M., Hesselman, C.: Domain Names Abuse and TLDs: From Monetization Towards Mitigation. In: Proc. of IFIP/IEEE. pp. 1077–1082 (2017)
37. Namecheap: https://www.namecheap.com/
38. Newton, A., Hollenbeck, S.: Registration Data Access Protocol (RDAP) Query Format. Tech. rep., RFC Editor (Mar 2015)
39. OpenPhish: https://openphish.com/
40. PhishLabs: Abuse of HTTPS on Nearly Three-Fourths of all Phishing Sites (Jun 2020), https://www.phishlabs.com/blog/abuse-of-https-on-nearly-three-fourths-of-all-phishing-sites/
41. PhisLabs: https://www.phishlabs.com
42. Sectigo Limited: Sectigo® Official - SSL Certificate Authority & PKI Solutions, https://sectigo.com/
43. SiteAdvisor, M.: https://www.siteadvisor.com
44. Spamhaus: http://www.spamhaus.org/
45. Spooren, J., Vissers, T., Janssen, P., Joosen, W., Desmet, L.: Premadoma: An Operational Solution for DNS Registries to Prevent Malicious Domain Registrations. In: 35th ACSAC. pp. 557–567 (2019)
46. SURBL: https://surbl.org/
47. Tajalizadehkhoob, S., Böhme, R., Gañán, C., Korczyński, M., Eeten, M.V.: Rotten Apples or Bad Harvest? What We Are Measuring When We Are Measuring Abuse. ACM Trans. Internet Technol. **18**(4) (2018)
48. Tajalizadehkhoob, S., Van Goethem, T., Korczyński, M., Noroozian, A., Böhme, R., Moore, T., Joosen, W., van Eeten, M.: Herding Vulnerable Cats: A Statistical Approach to Disentangle Joint Responsibility for Web Security in Shared Hosting. In: Proc. of ACM SIGSAC. p. 553–567 (2017)
49. Ulevitch, D.: PhishTank Join the fight Against Phishing (2006), https://phishtank.org/
50. URIBL: http://www.uribl.com/
51. Wang, Y.M., Beck, D., Wang, J., Verbowski, C., Daniels, B.: Strider Typo-Patrol: Discovery and Analysis of Systematic Typo-Squatting. In: Proc of USENIX Association - Volume 2. p. 5 (2006)
52. Zhang, P., Oest, A., Cho, H., Sun, Z., Johnson, R., Wardman, B., Sarker, S., Kapravelos, A., Bao, T., Wang, R., Shoshitaishvili, Y., Doupé, A., Ahn, G.J.: CrawlPhish: Large-Scale Analysis of Client-Side Cloaking Techniques in Phishing. In: Proc. of IEEE S&P. pp. 1109–1124 (2021)

Appendix

A Machine Learning Metrics

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN}$$

$$FPR = \frac{FP}{FP + TN} \qquad FNR = \frac{FN}{FN + TP}$$

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}, \tag{2}$$

where TN, TP, FN, and FP represent the numbers of true negative, true positive, false negative, and false positive, respectively. We refer to compromised domains as positives and to maliciously registered ones as negatives. Accuracy is the proportion of correctly predicted labels among all samples. We also make use of a Matthews Correlation Coefficient (MCC) as defined in Equation 2 [35]. This metric was developed to evaluate the quality of a binary classification and its values vary between -1 and +1, where +1 means perfect prediction (the best score), 0 is equivalent to random results, and -1 shows that all samples were misclassified (the worst score). In contrast to accuracy, MCC provides a more realistic metric for imbalanced datasets such as ours.

B Scatter Plots of Probability Changes

Fig. 10: Scatter plot of probability changes between the full model and the model without features related to WHOIS data (FS2).

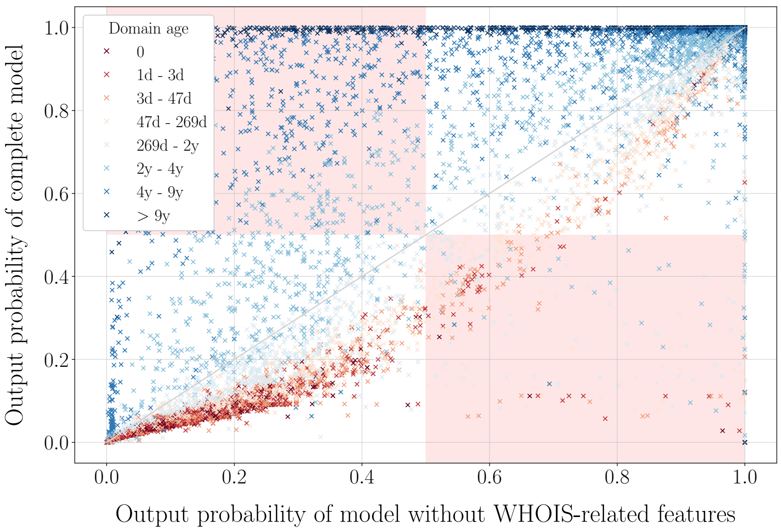
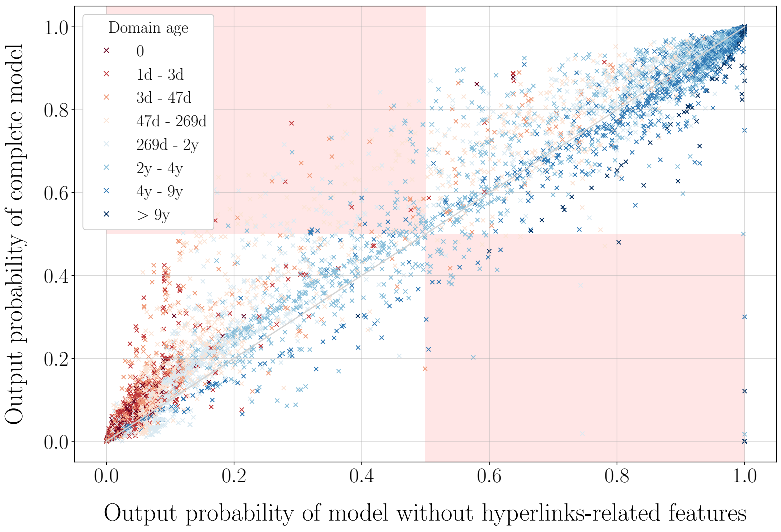


Fig. 11: Scatter plot of probability changes between the full model and the model without features related to hyperlinks (FS9).


Domain age
× 0
× 1d - 3d
× 3d - 47d
× 47d - 269d
× 269d - 2y
× 2y - 4y
× 4y - 9y
× > 9y
Output probability of complete model
Output probability of model without hyperlinks-related features