Identification of malicious domains based on temporal features of X.509 certificates and registrar records.

MSc Research Project
Cybersecurity

Ronan Shaw
Student ID: 22129618

School of Computing
National College of Ireland

Supervisor: Noel Cosgrave

National College of Ireland

MSc Project Submission Sheet

School of Computing

Student Name: Ronan Shaw

Student ID: 22129618

Programme: MSCCYB1 **Year:** 2022

Module: Research Project

Supervisor: Noel Cosgrave

Submission Due Date: 18th September 2023

Project Title: Identification of malicious domains based on temporal features of X.509 certificates and registrar records.

Word Count: ~7075 excl. refs & cover sheets **Page Count** 21 excl. cover sheets

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Ronan Shaw

Date: 14th September 2023

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	□
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	□
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	□

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Identification of malicious domains based on temporal features of X.509 certificates and registrar records.

Ronan Shaw
22129618

Abstract

Malicious network traffic relies on DNS and TLS to evade detection by appearing legitimate with techniques which include using algorithmically generated domains which are paired with legitimately issued X.509 certificates.

The underlying concept of this work is that the time between a domain's registration and the issuance of TLS certificates can be used to identify if a domain is malicious or not, regardless of the specificity of the threat. This paper takes several temporal attributes, from domain registrar WHOIS records and Certificate Transparency Logs, together with a novel certificate wildcard attribute, and engineers features used to train and test multiple models. Groups of feature sets are compared against each other on an intra and infra model basis.

This research demonstrates the accuracy (92%) of the engineered features considered, with very low FPR (0.2%) and f1-scores of 0.92 for prediction of malicious domains and 0.93 for benign domains. Furthermore, it identifies two temporal features which are of high significance and importance. In addition, it establishes the potential contribution of the novel wildcard certificate feature for identifying malicious domains.

1 Introduction

1.1 Background

The ability to successfully detect and prevent malicious network traffic is a key component in cybersecurity threat response. Successful malware and phishing campaigns depend upon internet protocols, such as DNS and HTTPS, to obfuscate and deceive people and systems into believing that they are in fact benign. This permits an initial foothold, in a network or through social engineering, to be parlayed into the ability to carry out the objective of the malicious software. As internet access has become ubiquitous over the last decade the volume of malware has increased from 300M to over 1.2B incidents[1].

The correlation, as discussed by Heron (2009), Hao, Feamster and Pandrangi (2011), Spring, Metcalf and Stoner (2011) , between spam domains and age of those domains lead to the adoption of real-time blocklists. In response the use of automation for registering and algorithmically generating domains was developed by spammers (Holz *et al.*, 2008).

Many of the techniques observed in spam domains were adapted for use by more recent malicious software. Seemingly legitimate domains using homographs (Quinkert *et al.*, 2019)

[1] https://portal.av-atlas.org/malware

are used by phishing sites to socially engineer victims, and obfuscated domains provide command and control for remote access, malware download and data exfiltration (Gallagher, 2021).

Similarly, the use of HTTPS forms a key part of the deception techniques used by malicious software. Fast-flux DNS coupled with automated HTTPS certificate processing, provided by the ACME protocol (Aas *et al.*, 2019), allows malicious software to rapidly obtain signed Domain Validated (DV) X.509 certificates from legitimate Certificate Authorities. These can be in the form of homomorphic domains for deception of phishing victims or as a malware domain with a legitimate DV certificate. Automation increases the rate of change presenting a challenge for DNS blocklists to maintain up to date lists. Similarly, Certificate Authorities face the challenges of timely revocation of certificates, distributing Certificate Revocation Lists (Wilson, 2022) and updating OSCP (Santesson *et al.*, 2013).

A related challenge arises when the data volumes surpass the link capacity of clients or servers, leading to delays in identifying fast-fluxing domains and certificates or rendering the information quickly outdated and ineffective. Additionally, low energy capabilities of IoT edge devices may hinder continuous processing of these feeds.

To address these issues when identifying malicious domains, classification models using domain registration times or data mining domain names, have been proposed by He *et al.* (2010), Spring *et al.* (2011) and Maroofi *et al.* (2020). Similarly, for the identification of phishing websites using HTTPS, classification based on the certificates in Certificate Transparency logs was proposed, amongst others, by Fasllija, Ferit Enişer and Prünster (2019), Sakurai *et al.* (2020) and Drichel *et al.* (2021). However, they generally address these aspects in isolation.

These papers along with others, such as Dong, Kane and Camp (2016), either consider only phishing or malware, certificates or domains, but not the combination of generally malicious domains along with certificates. More recent research by AlSabah *et al.* (2022) and Drury, Lux and Meyer (2022), which considers domain registration dates, and certificate validity elements is limited to considering only phishing domains. Similarly, the paper by Chiba *et al.* (2018) evaluates domains for general maliciousness, but only using DNS and WHOIS data.

1.2 Objective

This paper will address the gaps identified in existing literature through the research question: "How accurate are the temporal features of domain registration and TLS X.509 certificates for identifying malicious domains used for command and control, social engineering through hosting or phishing, and distribution of malware?"

In addition, it will include a novel use of parent domain wildcard certificates where there is no certificate matching the FQDN. Combinations of domain registration and certificate temporal features will be compared using Logistic Regression and Random Forest classifiers to determine which approach gives the best accuracy, while ensuring a low false positive rate.

1.3 Report structure

In Section 1, the introduction sets the stage with a brief history of the problem, the context of prior related work, and the reports objective. Moving on to Section 2, a comprehensive literature review discusses key papers that underpin the latest research in the field, focusing on their relevance to the report's central topics and research niche. Subsequently, in Section 3, the research methodology is presented, with context related to prior works. Sections 4 and 5 serve as supplements to Section 3, providing the design specifications, and implementation details. The outcomes of examined feature sets and models are critically evaluated in Section 6. Finally, the report concludes in Section 7, offering insights and suggestions for potential future work.

2 Related Work

2.1 Background

The formative analysis by Whittaker, Ryner and Nazif (2010) established that malicious websites hosting phishing and general malware could be identified by classification of features extracted from the site's URLs, domain and page content. Their research demonstrated high detection rates with low false positives, although it focuses specifically on phishing and relies on access to features of the proprietary Google PageRank (Page *et al.*, 1999) algorithm. The results of He *et al.* (2010) paper support the validity of domain name features for classifying malicious sites, again focusing on phishing rather than malware. Through examining X.509 public key certificates alongside domain name features Dong *et al.* (2015) and Anderson, Paul and McGrew (2018) expanded upon this prior research and adapted the prior approaches for the identification of malware.

More recent research, including Drichel *et al.* (2021), Le Pochat *et al.* (2020) and AlSabah *et al.* (2022) neglects to consider all malicious domain types such as those used for command and control, phishing and malware. An exception to this is the thorough work by Maroofi *et al.* (2020) which considers all three types of malicious domains and combines the analysis of lexicographical domain name elements and the issuance details (CA, price & validity) derived from the Certificate Transparency (CT) Log.

This paper will consider all types of malicious traffic, not only malware or phishing, which uses DNS and TLS certificates and focus on examining the temporal features which can be used for classification of domains.

2.2 Domains and DNS

As an essential internet service, DNS (Mockapetris, 1987b; Mockapetris, 1987a), provides the ability for all clients, legitimate and malicious, to resolve fully qualified domain names (FQDN) to IP addresses. Initial approaches to detecting malicious domains focused on pre-compiled "Realtime" block lists but the advent of fast-flux networks (Salusky and Danford, 2007), where DNS domains and records are rapidly changed or discarded, overwhelmed attempts to update these lists in real time.

The DNS protocol, unlike CT Logs which will be used in this research, does not provide a native mechanism for tracking changes in DNS records. Weimer (2005) proposed passive DNS (pDNS) record collection, through network interception, as a mechanism to record these changes, with sensors placed on the DNS hierarchy to gather and aggregate the changes.

Antonakakis *et al.* (2010) went on to quantify how the reputation of a domain data in pDNS could be classified dynamically using known lexicographical features, IP address and BGP Autonomous System features. This requires a large amount of data from multiple ISP recursive resolvers and the practical use of such an approach was acknowledged as limited in their follow up paper (Antonakakis *et al.*, 2011). In this work they introduced a model to detect malware domains without access to large data feeds from other networks. However, both papers make no attempt to classify domains beyond the second level e.g., *.example.com*.

More recently research by AlSabah *et al.* (2022) and Darwish, Farhan and Elzoghabi (2023) has leveraged pDNS datasets[2], to detect phishing websites, but access to these datasets can be challenging as they are commercially restricted, private or require an invite. This limits the reproducibility of studies which leverage these datasets.

Alternative domain related features which can be used are provided by the domain registrar WHOIS information. These were identified by previously by Weimer (2005) in their research but discounted as registrant information was often unreliable, although they failed to recognise that WHOIS records "Creation Date" attributes are idempotent across registrar changes and domain renewals. In recent years the use of WHOIS data has been impacted by GDPR (European Parliament and Council of the European Union, 2016) privacy requirements resulting in personally identifiable information being redacted, as recognised by Le Pochat *et al.* (2020) and Lu *et al.* (2021), resulting in timestamps being some of the only distinctive attributes in WHOIS records for identifying malicious sites. Although acknowledged, and used as a mechanism to verify if a domain had been resurrected from dormancy, AlSabah *et al.* (2022) did not use the WHOIS timestamp data in their model for phishing domain detection.

This paper will utilise the WHOIS domain creation timestamp as a component part of the features used to train a model to detect any malicious domain, not only phishing or malware domains.

2.3 X.509 Certificates

While there is no equivalent built-in standard within DNS protocol and infrastructure for tracking changes in records, CT Logs now provide a full historical record of all CA issued certificates by design.

The pioneering study by Mishari *et al.* (2012) utilised nine X.509 certificate attributes for the detection of phishing and typosquatting concluded that while the model was useful the limited level of HTTPS adoption at the time prevented wider use. Dong *et al.* (2015) expanded on the use of certificate attributes by adding engineered features. Both these studies were limited to examination of the current TLS certificates deployed on web servers as they predate

[2] https://www.circl.lu/services/passive-dns/

the widespread adoption of Certificate Transparency Logs in 2018, and the models can behave poorly, as demonstrated by Dong *et al.* (2015), when faced with fast-flux domains.

The pilot paper by Scheitle *et al.* (2018) examined the potential use of CT Logs for phishing detection, but limited the attributes used to the CN and SAN fields. This work was expanded upon with Phish-Hook (Fasllija *et al.*, 2019) which incorporated a live feed of newly issued certificates to train a classifier on lexicographical features but, notably excluded temporal elements such as certificate validity attributes and evaluated performance against an unidentified UCI Machine Learning repository dataset.

While CT Logs are open and directly accessible, prior research papers, such as AlSabah *et al.* (2022) and Drury, Lux and Meyer (2022), used the crt.sh[3] certificate monitor website to retrieve certificate information. One exception is Scheitle *et al.* (2018), which used the bulk CT log data, although they failed to specify which CT Log source was used for the dataset in the paper.

In this paper the bulk CT Log data is a key dataset from which the certificate temporal attributes will be extracted to create engineered features for the models.

2.4 Combining Domain and Certificate

Use of certificate validity period (in days), retrieved directly, to identify phishing websites was introduced by Drury and Meyer (2019) and expanded upon by Drichel *et al.* (2021), including domain and keyword features. Drichel *et al.* (2021) explicitly excluded the use of WHOIS in their analysis and certificate validity dates or other similar temporal features are not considered.

The use of both DNS, domain registration data along with X.509 certificate features was systematically examined in Maroofi *et al.*, (2020). Despite gathering and evaluating the use of temporal features their research only used CT log as a substitute for domain registration dates rather than a standalone feature in the analysis. Their COMAR model included a novel use of the Internet Archive Wayback Machine to identify the earliest recorded live instance of a domain as part of the extensive evaluation against malware, command and control and dynamically generated domains.

Most recently, Drury, Lux and Meyer (2022), included earliest certificate validity dates from the CT Log and domain creation dates for the identification of phishing domains. Both AlSabah *et al.* (2022) and Drury, Lux and Meyer, (2022) acknowledge the limitations of using crt.sh, referring to the underlying causes as “due to server or network errors” AlSabah *et al.* (2022) and “connection errors” Drury, Lux and Meyer, (2022) leading to results being either discarded or alternative data sources used.

These approaches demonstrate that crt.sh is unsuitable as the primary data source for CT Log information and this research will, instead, use a bulk download of the CT Log as a primary source and only supplement it with queries to the monitor if entries are missing.

The presence of a wildcard certificate for a domain, amongst prior research utilising X.509 certificate features, has seldom been considered by prior studies. Although it was identified by Drichel *et al.* (2021) it was not selected as a feature within their model.

[3] https://crt.sh/

This analysis will focus on using the temporal features of X.509 certificates, from the CT Log, along with a novel wildcard parent domain certificate feature, combined with domain registration, from WHOIS, to determine how well the features perform in identifying both malware and phishing domains.

3 Research Methodology

3.1 Dataset selection:

To obtain the required temporal data several sources were considered for the malicious and benign FQDNs as well as potential sources for domain and certificate temporal features. These sources were collated from prior research papers in addition to independent research. As many prior papers (AlSabah *et al.*, 2022; Darwish *et al.*, 2023) relied on closed data sources this study will, to ensure reproducibility and the potential for practical implementation in the future, use the openly accessible public data sources.

For obtaining the domain registration dates the canonical approach taken in prior research was the use of WHOIS data (Lu *et al.*, 2021; Drury *et al.*, 2022). Despite the semi-structured textual format of the data there are several python libraries[4] supporting queries against the different formats provided by registrars. The Registration Data Access Protocol[5] was considered and investigated but adoption of it is not universal, and the APIs were found to be rate limited. WHOIS lookup followed by traversal of the FQDN by subdomain is carried out to top level registered domain.

Historical validity data for X.509 certificates is openly available since the widespread adoption of CT Logs. Prior research has primarily accessed this using an API provided by a certificate log monitor service, such as used by AlSabah *et al.* (2022) and Drury, Lux and Meyer (2022). These monitor services, as noted by these papers, implement query limits and thus the alternative direct download of CT Log data was chosen, as carried out in Scheitle *et al.* (2018), although they fail to specify the exact source CT Log in their paper.

CT Logs record all certificates issued by Certificate Authorities (CAs) to provide an audit trail for legitimacy and prevent rogue issuance. The logs consist of an append-only signed Merkle tree. Modern browsers require that certificates be included in the CT Log which is indicated by including the signed certificate timestamps from the CA in the certificate, Firefox being the exception. Thus, most certificates are included in at least one log. However, analysing certificate attributes becomes challenging due to variations across logs and their use of temporal sharding for efficient storage. For this analysis the Cloudflare Nimbus 2023 CT Log[6] was chosen as the primary data source for certificate lookups giving a view of issued certificates from January to June 2023.

[4] https://pypi.org/search/?q=WHOIS

[5] https://www.icann.org/rdap

[6] https://ct.cloudflare.com/logs/nimbus2023

A supplementary dataset[7] identifying TLD suffixes from the Mozilla project was used to exclude these from parent domain wildcard certificate checks and recursive WHOIS lookups.

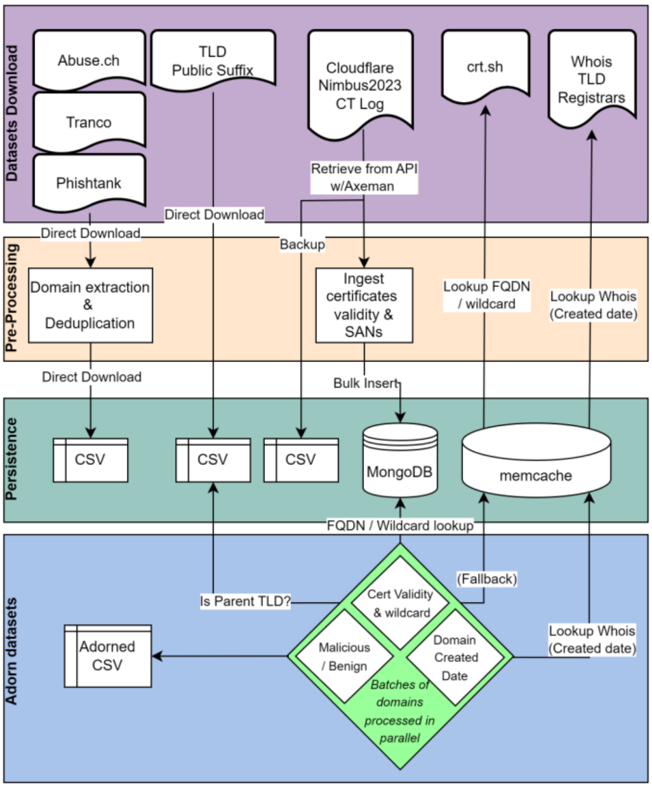


Figure 1 - Abstract of the Python dataset processing

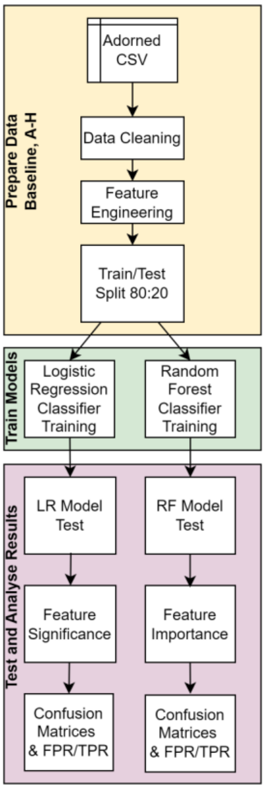


Figure 2 - Abstract of the Training, Test and Analysis process

To train and test a model, using the temporal features of domain registration and TLS certificates, three primary datasets are chosen to provide the FQDNs and a label: Two malicious domain datasets, PhishTank (*PhishTank,* nd), URLhaus by Abuse.ch (*URLhaus*, nd), and one benign domain dataset, Tranco (Pochat *et al.*, 2019).

The two malicious datasets were chosen to provide domains which had been observed as participating in Phishing as well as domains used by Malware and Botnets. This gives a broad malicious dataset on which to train the model. The Tranco dataset aggregates a research-oriented list of most popular domains from multiple sources, providing the benign dataset. For this paper the Tranco 1m Umbrella dataset was chosen.

3.2 Dataset pre-processing

While the domain datasets are all in CSV format, the fields provided vary, thus the datasets were parsed, using pandas, to extract the domain from the URL in the case of the PhishTank and Abuse.ch dataset, and the domain directly from the Tranco dataset. During the initial analysis and verification of the data the “dateadded” field in the Abuse.ch dataset was identified

[7] https://wiki.mozilla.org/Public_Suffix_List

as matching the WHOIS created timestamp, thus for this dataset an additional online lookup can be avoided where this field is available.

The CT Log dataset is extremely large as each certificate is included along with the entire signature chain, with new certificates added at a rate of ~160K per hour[6]. Retrieving and parsing this data was a two-step process. The first step used a fork of the Axeman utility[8], which has been updated to parse Version 2 of the log format, to retrieve the logs from Cloudflare. The resulting CSV files are then parsed, with the CN, SAN and Validity fields from each certificate inserted into a MongoDB database to make it searchable by domain.

During data processing the use of a wildcard certificate on the parent domain for a fully qualified domain in the dataset was captured. Initial analysis of this data showed that there was potentially a difference between the use of wildcard certificates on benign and malicious domains, Figure 3. This novel feature was included as no prior research using it had been identified.

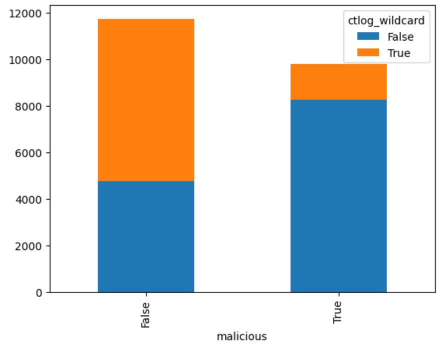


Figure 3 - Certificate Wildcard vs maliciousness by Domain

3.3 Feature preparation

To generate the training and test datasets a sample is randomly selected, from both the phishing and malware (malicious) domain dataset, along with an equally sized random sample from the benign domain dataset. A Boolean field, malicious (Table 1), is added for each domain as the dependent variable, with the value based on which dataset it originated from.

Table 1 Extracted Features

Name	Description	Source
$F_{Malicious}$	The label/target variable: 1 if malicious, 0 if benign.	Malicious: Abuse.ch, PhishTank Benign: Tranco
$F_{DomainCreationDate}$	The creation date of the domain.	WHOIS
$F_{EarliestCertValidityStart}$	The earliest certificate validity start timestamp for this domain, or wildcard parent.	Cloudflare CTLog, crt.sh
$F_{LatestCertValidityEnd}$	The latest certificate validity end timestamp for this domain, or wildcard parent.	Cloudflare CTLog, crt.sh

[8] https://github.com/cyrill-k/Axeman

The dataset is split into batches of domains which are processed in parallel as WHOIS lookups are considerably slower than local queries and subject to retries and timeouts when compared with local CT Log database lookups.

Within a batch each domain's creation date is queried with the WHOIS protocol and cached to improve performance of subsequent lookups for the same domain. A lookup using the FQDN against the certificate CNs and SANs in the CT Log database is carried out to retrieve validity dates, or if a parent domain wildcard certificate was found. If unsuccessful, as a fallback, a lookup is made against crt.sh and the results are cached.

Once parallel processing of a batch completes the resulting records containing the features listed in Table 1 are merged with previously completed batches and saved to a CSV file.

3.4 Feature engineering

Once the additional fields are added to the dataset the engineered features, Table 2, for training and test are calculated and merged with the dataset.

Table 2 Engineered Features

Name	Description
$F_{DomainToEarliestCertDelta}$	Absolute delta in days between $F_{DomainCreationDate}$ and $F_{EarliestCertValidityStart}$
$F_{DomainToLatestCertDelta}$	Absolute delta in days between $F_{DomainCreationDate}$ and $F_{LatestCertValidityEnd}$
$F_{CTLogWildcard}$	Set if only a parent domain wildcard certificate was found in the CT log for the domain. 1 if True, 0 if False
$F_{EarliestCertDayOfWeekSin}$ $F_{EarliestCertDayOfWeekCos}$	Circular encoding of the day of the week the earliest certificate was valid from. Calculated from $F_{EarliestCertValidityStart}$
$F_{LatestCertDayOfWeekSin}$ $F_{LatestCertDayOfWeekCos}$	Circular encoding of the day of the week the latest certificate was valid from. Calculated from $F_{LatestCertValidityEnd}$
$F_{WHOISCreatedDayOfWeekSin}$ $F_{WHOISCreatedDayOfWeekCos}$	Circular encoding of the day of the week of the creation date of the domain from WHOIS. Calculated from $F_{DomainCreationDate}$

To examine the relationship between purely temporal aspects of malicious domain registration and certificate issuance the engineered features are chosen based on their contribution to when such an event occurred. Interpretation of timestamp events such as the domains registration date and the certificate validity dates require that they be transformed to an interval number of days. The domain registration date is considered the start of the interval and the subsequent certificate validity date the end of the interval. From this we derive the two "date delta" features: $F_{DomainToEarliestCertDelta}$ and $F_{DomainToLatestCertDelta}$.

Cyclical encoding, using the sine and cosine of the day of the week of the date for the domain registration or certificate validity time allows the classifier to interpret if a particular day of the week for each is more likely to be associated with a malicious domain.

3.5 Model Training, Testing and Result Analysis

Following feature selection, the combined dataset is split, 80:20, into training and testing sets. To determine the impact upon accuracy, when training with different groups of features, nine sets of analyses are carried out using both Logistic Regression (LR) and Random Forest (RF) based classifiers.

A baseline using only $F_{DomainToEarliestCertDelta}$ was chosen to establish the accuracy of the models based on this single temporal datapoint. The feature sets for comparison of the models were chosen based on grouping the three types of temporal data in the original features extracted from the datasets:

- Features derived from the Domain Registration date.
- Features derived from the Earliest issued certificate validity from date.
- Features derived from the Latest issued certificate validity to date.

The non-temporal datapoint derived from a wildcard certificate was included with some groups. The final groupings were chosen based on having comparative groups with a spread of the feature represented.

Table 3 lists the features chosen for each set of analysis. The key feature, $F_{DomainToEarliestCertDelta}$, is included in all sets.

Table 3 Features in each set.

Feature	Baseline	A	B	C	D	E	F	G	H
$F_{DomainToEarliestCertDelta}$	X	X	X	X	X	X	X	X	X
$F_{EarliestCertDayOfWeekSin}$		X		X		X		X	X
$F_{EarliestCertDayOfWeekCos}$		X		X		X		X	X
$F_{CTLogWildcard}$			X	X		X		X	X
$F_{WHOISCreatedDayOfWeekSin}$							X	X	X
$F_{WHOISCreatedDayOfWeekCos}$							X	X	X
$F_{DomainToLatestCertDelta}$					X	X			X
$F_{LatestCertDayOfWeekSin}$						X			X
$F_{LatestCertDayOfWeekCos}$						X			X

A harness allows all notebooks to be executed in series and the results exported as a HTML page and saved for future reference. For both models the resulting output contains the summary statistics for the data, Classification Report and Confusion Matrices. In addition, the LR reports contain the Logit Regression report, with significance values for each feature and the RF model reports contain the hyperparameters, selected by grid search, and the importance of each feature. Collation, comparison, and analysis of the results from each report is performed separately using MS Excel.

4 Design Specification

4.1 Feature preparation

A key part of the preparation of data is the use of caching along with recursive lookup strategy to retrieve the domain creation dates. This reduces the load on network services and seeks to avoid rate limiting if FQDNs in the domain list share common registrars.

A FQDN in the domain dataset may have multiple levels of domain hierarchy applied. To find the registered domain creation date the FQDN is recursively traversed checking the cache for stored results. This is followed by a direct lookup for each level in the hierarchy against WHOIS until a result is found, Figure 4. A TLD Suffix List[7] is used to avoid queries against potential second level and top level domains which are not valid, such as *.co.uk*.

If the CT Log entry for the domain is not available in the database following preprocessing of the CT Log a lookup against the Certificate Monitor service, crt.sh, is used. Similar to the WHOIS lookup, this process also relies on caching the results, Figure 5, but differs by only evaluating if the FQDN or the direct parent domain wildcard certificate exists in the CT Log.

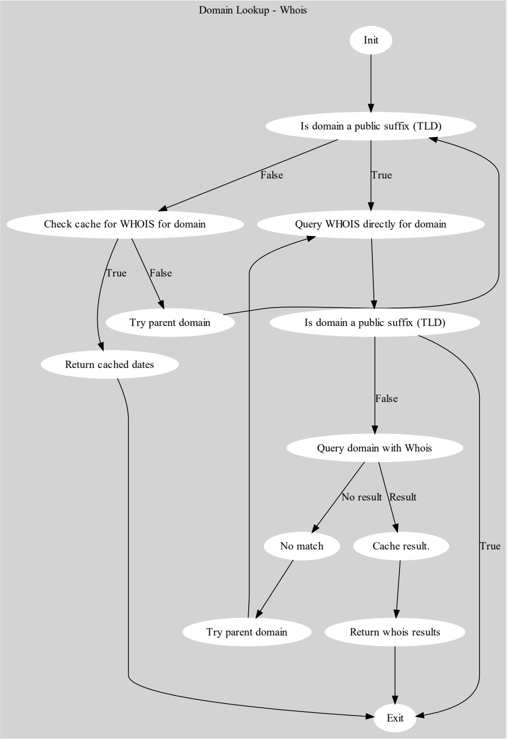


Figure 4 - WHOIS Domain Lookup Function

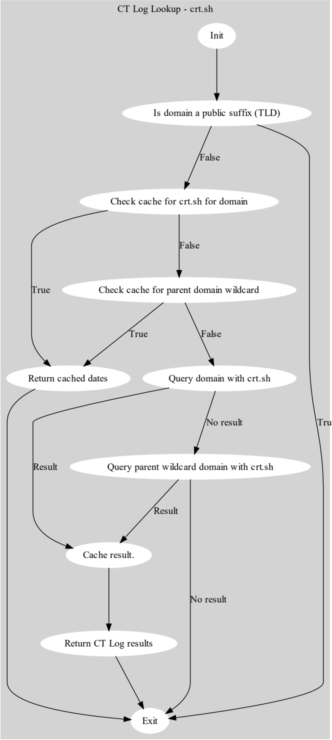


Figure 5 - CT Log crt.sh Lookup Function

4.2 Classification

Logistic Regression & Random Forest classifiers were chosen based on the requirement to identify qualitatively whether a domain belongs in either the malicious or benign category - a binary dependent variable, based on the groups of features - the independent variables. These classifiers are used in several prior related papers including He *et al.*, (2010), Maroofi *et al.* (2020), Drichel *et al.* (2021) and Abuadbba *et al.* (2022).

The choice of two classifiers allows a comparison and verification of the results, ensuring more certainty that temporal features are, or are not, useful for identifying malicious domains.

Multiple Logistic Regression is used for Feature Sets A-H, with the Baseline Feature set being the simple Logistic Regression case where there is one independent variable. The resulting significance scores for each feature can be used for evaluating which features are statistically significant within the feature set.

Random Forests (Breiman, 2001), is an "ensemble learning" technique combining decision tree predictors, bagging of features and randomization of features. It is particularly robust, with the Gini coefficient applied, in resisting noise in the data and makes a good candidate for comparison to the LR classifier. The predictive power of RF relies on the hyperparameters chosen for the model to adjust the accuracy or speed.

For this research a grid search is carried out to evaluate all potential combinations of features, with the hyperparameters in Table 4. For evaluation the feature importance, derived from the mean decrease of node impurity scores, provides a mechanism for comparing each feature within the model.

Table 4 - RF Hyperparameter Candidate Values

Hyperparameter	Candidate Values
Number of estimators	50,100,150,200
Max features	Square root, $\log^2$
Max depth	2,3,4,5
Criterion	Gini, Entropy

4.3 Analysis of Results

Comparative analysis between the Logistic Regression and Random Forest classifiers is carried out by collating all classifier results for the feature set groups and examining the Accuracy, Recall, Precision & F1-score metrics along with a comparison of True Positive Rate vs False Positive rate, along with Confusion Matrices for each feature group and classifier.

Furthermore, the influence of each of the features is evaluated using the significance scores for features from Logistic Regression classifiers and feature importance from Random Forest classification. This allows intra and inter model identification of significant features or any outliers.

5 Implementation

Python3 is used exclusively for automating dataset processing and lookups of the WHOIS and CT Log features. Jupyter notebooks are then used to carry out the training, testing, and reporting of the results. Both Python and Jupyter are executed from a pipenv virtual environment shell to ensure an idempotent environment with a reproducible set of dependencies.

5.1 Dataset retrieval & pre-processing

Dataset retrieval is a manual process for the malicious and benign domain data with dataset saved locally before preprocessing, using pandas, to extract relevant fields and remove duplicate domains.

The CT Log download from Cloudflare's Nimbus 2023 log is carried out using the Axeman tool[9]. This totals 1.4TBytes of CSV files, when compressed. A custom parser extracts the files in batches in parallel and retrieves the validity start time (validFrom) of the earliest and validity

[9] *The large size of the CT Logs and the rate at which they are appended to, presents challenges which were overcome by the tools developer using cloud computing (Retrieving, Storing and Querying 250M+ Certificates Like a Boss) to speed up the process. When the same approach was attempted for this research, it was discovered that the CT Log APIs for all providers now enforce extreme rate limiting which caused the tooling to fail to download the logs. Thus, a local download approach was used.*

end time (validTo) of the latest certificate in addition to the CN & SANs attributes and any wildcard domain entries. The extracted data is stored in MongoDB with an index applied to the list of CN & SANs to enable efficient searches, this reduces the dataset to less than 250GB.

Using pandas, the datasets are loaded, and the malware, phishing and benign domain datasets are sampled based on the size of the dataset that will be supplied to the models – early iteration of the process used 100-1000, the final iteration used 10,000. Double the sample is taken from the benign dataset to maintain balance with the malware and phishing domains.

5.2 Feature retrieval

A Python script is used to process the combined dataset, using parallel threads. Loading the pre-processed data from CSV into a pandas data frame it is converted to a Python list of dictionaries for parallel processing. The data is split into batches of 100 domains each. Each batch joins a queue for processing, with the number of parallel batches equal to the system core count.

In series, a lookup is performed for each domain in the batch to retrieve the WHOIS created date, Figure 4, and CT Log entry dates, Figure 5:

- The domain WHOIS lookup will traverse from the FQDN to the TLD checking if the entry exists in memcache and failing that uses the Python WHOIS module to lookup directly against the appropriate registrar. If the domain has no WHOIS created date it is given the Unix epoch date, 1970-01-01. By always returning date values complex comparisons of different types are avoided. Positive responses from WHOIS are cached in memcache, to speed up subsequent executions.
- A lookup of the domain against the CN and SAN entries extracted to the Mongo DB is carried out. If the FQDN is not found a search for the parent domain wildcard certificate occurs. If no CT Log entry is found in the database a fallback lookup is applied with a query to the crt.sh certificate transparency monitor. As noted, this can be unreliable, or subject to rate limiting. If an entry is found in crt.sh it is parsed and the earliest, latest and wildcard values are cached in memcache. If no details are found the Unix epoch is returned.

As each parallel batch completes it is appended to a new data frame and when all complete the interim results are saved to disk as CSV, before further processing is applied to calculate the engineered features, and then saved to disk as CSV.

5.3 Feature Engineering

In the final implementation the interim data CSV is loaded by the Jupyter notebooks, using pandas, and any records which contain the epoch, or null values for the features are removed. The correct data types and absolute transformations are applied to the data frame fields. Feature engineering is then applied using lambda with custom helper functions. Unused features are now dropped. Following feature lookup, extraction, and data cleaning there remained 9,810 malicious domains and 11,739 benign domains.

5.4 Train, Test and Results

On each of the feature sets the training & testing for the LR model is carried out with statsmodel, using Logit(). This produces a results report for feature significance evaluation. For the RF model the sklearn GridSearchCV() was used to determine the parameters to use with RandomForestClassifier(), and feature importance rankings generated for the model.

Evaluation of both models and feature sets was carried out using sklearn. The confusion_matrix() function results are collated, the True Positive Rate (TPR) and False Positive Rate (FPR) calculated, and presented in Tables 5 & 7, and classification_report() function results are collated and presented in Figure 6 to 9.

This comparison of LR feature significance with the RF feature importance provides an important cross check of correctness. During the evaluation a mismatch between results identified a code issue with the presentation of the RF feature importance results.

6 Evaluation

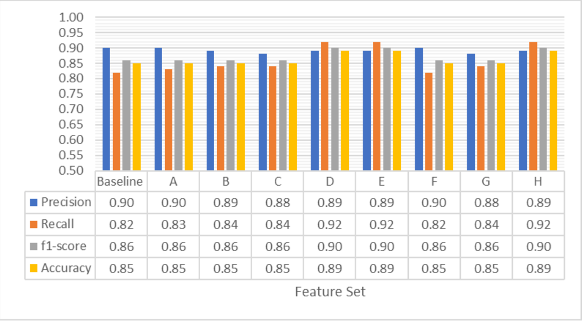
An assessment of the results of the two classifiers for the chosen feature sets was carried out to evaluate the research question, as stated in section 1.2.

To evaluate the results the models are examined with the following criteria:

- Examination & comparison of the f1-scores for benign and malicious predictions by feature set, both intra-model and inter-model.
- Examination & comparison of the TPR and FPR, calculated from confusion matrices, by feature set, both intra-model and inter-model.
- Significance of per feature p-value for Logistic Regression Model
- Feature importance score for the Random Forest Model.

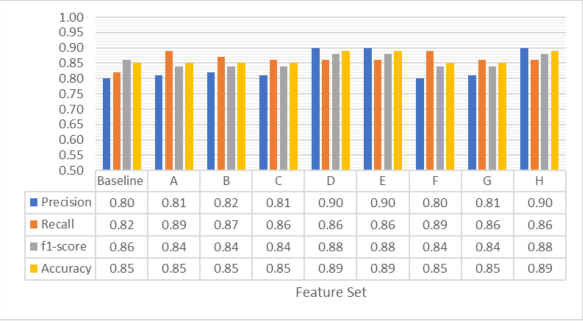
6.1 Logistic Regression

Comparison of the LR models for the D, E & H feature sets shows they outperformed the others, achieving f1-scores of 0.9 and 0.88 respectively for both benign and malicious predictions, Figure 6, 7



	Baseline	A	B	C	D	E	F	G	H
🟦 Precision	0.90	0.90	0.89	0.88	0.89	0.89	0.90	0.88	0.89
🟧 Recall	0.82	0.83	0.84	0.84	0.92	0.92	0.82	0.84	0.92
⬜ f1-score	0.86	0.86	0.86	0.86	0.90	0.90	0.86	0.86	0.90
🟨 Accuracy	0.85	0.85	0.85	0.85	0.89	0.89	0.85	0.85	0.89

Figure 6 Logistic Regression Benign Scores



	Baseline	A	B	C	D	E	F	G	H
🟦 Precision	0.80	0.81	0.82	0.81	0.90	0.90	0.80	0.81	0.90
🟧 Recall	0.82	0.89	0.87	0.86	0.86	0.86	0.89	0.86	0.86
⬜ f1-score	0.86	0.84	0.84	0.84	0.88	0.88	0.84	0.84	0.88
🟨 Accuracy	0.85	0.85	0.85	0.85	0.89	0.89	0.85	0.85	0.89

Figure 7 Logistic Regression Malicious Scores

. This is reflected in the FPR of 0.08 and TPR of 0.86 for these feature sets presented in Table 5. The accuracy score achieved, for these feature sets, of 0.89 is above the random probability (0.5) of correctly classifying the domain as malicious or benign.

The remaining feature sets trailed slightly with f1-scores between 0.84 and 0.86 in both malicious and benign predictions. Accuracy for these sets of features was consistently 0.85.

The common subset of features in sets D, E & H are $F_{DomainToEarliestCertDelta}$ and $F_{DomainToLatestCertDelta}$ and sets E & H also have the common features of $F_{EarliestCertDayOfWeekSin}$, $F_{EarliestCertDayOfWeekCos}$, $F_{CTLogWildcard}$ and $F_{LatestCertDayOfWeekSin}$, $F_{LatestCertDayOfWeekCos}$. The strong correlation of these features and accuracy scores of the model suggests that temporal aspects of the domain registration and certificate issuance validity may be an indicator of malicious domains, more than random chance alone suggests.

Table 5 Logistic Regression FPR/TPR results by Feature Set

Feature Set	False Positive Rate	True Positive Rate
Baseline	0.18	0.89
A	0.17	0.89
B	0.16	0.87
C	0.16	0.86
D	0.08	0.86
E	0.08	0.86
F	0.18	0.89
G	0.16	0.86
H	0.08	0.86

To further analyse this following hypothesis was considered:

- Null hypothesis H_0: A given temporal feature of domain registration and TLS X.509 certificates is an accurate mechanism for identifying malicious domains.
- Alternative hypothesis H_1: A given temporal feature of domain registration and TLS X.509 certificates is no more accurate at detecting malicious domains, than random chance alone.

A significance level was chosen of α = 0.05.

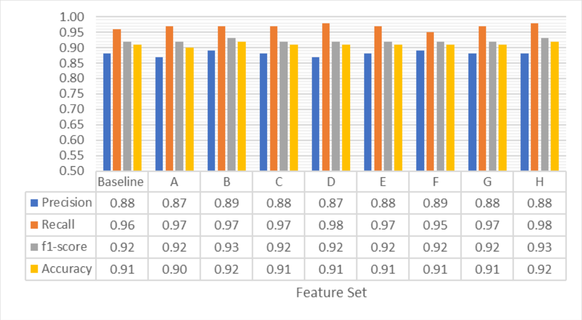
Table 6 Logistic Regression Feature Significance

	P>|z|								
Feature	**Baseline**	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**
$F_{DomainToEarliestCertDelta}$	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
$F_{EarliestCertDayOfWeekSin}$		0.000		0.000		0.004		0.000	0.004
$F_{EarliestCertDayOfWeekCos}$		0.000		0.000		0.000		0.000	0.000
$F_{CTLogWildcard}$			0.000	0.000		0.004		0.000	0.005
$F_{WHOISCreatedDayOfWeekSin}$							0.000	0.001	0.213
$F_{WHOISCreatedDayOfWeekCos}$							0.046	0.173	0.158
$F_{DomainToLatestCertDelta}$					0.000	0.000			0.000
$F_{LatestCertDayOfWeekSin}$						0.000			0.000
$F_{LatestCertDayOfWeekCos}$						0.000			0.000

The statistical significance of each feature in each set of features was compiled in Table 6 from the *Logistic Regression Report*. Based on the chosen α = 0.05 the p-value is <0.05 to determine if a feature's significance is as extreme of more extreme than if observed by random chance alone. The results show extreme, significant, test statistics for the features in nearly all cases but, notably, the two "*WhosisCreatedDayOfWeek*" features are not as extreme suggesting that the day of the week when a domain is registered has potentially no bearing whether a domain is malicious or not, other than by random chance.

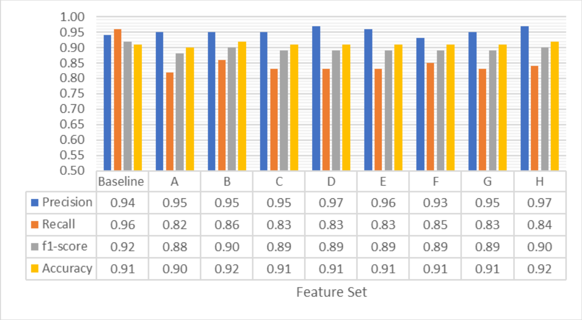
6.2 Random Forest

The RF models provide an improved FPR when compared with the LR model, albeit at the expense of some sensitivity. The classifiers using feature sets D & H performed well at predicting benign domains with f1-scores of 0.92 and 0.93 respectively, with their f1-scores for malicious domains lagging slightly behind – 0.89 and 0.9 in Figure 8, 9. This represents an improvement in predicting malicious domains on the LR model, as evidenced by the lower f1-scores. This improvement can also be observed in the FPR of 0.02, but with a slight decrease in the TPR, as presented in Table 7.



	Baseline	A	B	C	D	E	F	G	H
🟦 Precision	0.88	0.87	0.89	0.88	0.87	0.88	0.89	0.88	0.88
🟧 Recall	0.96	0.97	0.97	0.97	0.98	0.97	0.95	0.97	0.98
⬜ f1-score	0.92	0.92	0.93	0.92	0.92	0.92	0.92	0.92	0.93
🟨 Accuracy	0.91	0.90	0.92	0.91	0.91	0.91	0.91	0.91	0.92

Figure 8 Random Forest Benign Scores



	Baseline	A	B	C	D	E	F	G	H
🟦 Precision	0.94	0.95	0.95	0.95	0.97	0.96	0.93	0.95	0.97
🟧 Recall	0.96	0.82	0.86	0.83	0.83	0.83	0.85	0.83	0.84
⬜ f1-score	0.92	0.88	0.90	0.89	0.89	0.89	0.89	0.89	0.90
🟨 Accuracy	0.91	0.90	0.92	0.91	0.91	0.91	0.91	0.91	0.92

Figure 9 Random Forest Malicious Scores

Feature sets D & H, with $F_{\text{DomainToEarliestCertDelta and}}$ $F_{\text{DomainToLatestCertDelta}}$ in common, again score well with low FPR, although their TPR does not perform as well compared to other features sets, deviating from what was observed with the LR model.

Table 7 Random Forest FPR/TPR results by Feature Set

Feature Set	False Positive Rate	True Positive Rate
Baseline	0.04	0.84
A	0.03	0.82
B	0.03	0.86
C	0.03	0.83
D	0.02	0.83
E	0.03	0.83
F	0.05	0.85
G	0.03	0.83
H	0.02	0.84

To further examine the contribution of features in each set to the overall TPR/FPR results the importance of each feature in each set was calculated and compiled in Table 8. From this the contribution of $F_{\text{DomainToEarliestCertDelta}}$ is visible in all sets, and the contribution of $F_{\text{DomainToLatestCertDelta}}$ when included in sets D, E & H is clear.

Table 8 Random Forest Feature Importance

	Feature Set								
Feature	**Baseline**	**A**	**B**	**C**	**D**	**E**	**F**	**G**	**H**
$F_{\text{DomainToEarliestCertDelta}}$	1.000	0.965	0.862	0.875	0.526	0.459	0.963	0.808	0.453
$F_{\text{EarliestCertDayOfWeekSin}}$		0.023		0.006		0.006		0.007	0.004
$F_{\text{EarliestCertDayOfWeekCos}}$		0.012		0.018		0.011		0.021	0.015
$F_{\text{CTLogWildcard}}$			0.138	0.101		0.076		0.136	0.072
$F_{\text{WhoisCreatedDayOfWeekSin}}$							0.023	0.009	0.007
$F_{\text{WhoisCreatedDayOfWeekCos}}$							0.015	0.018	0.005
$F_{\text{DomainToLatestCertDelta}}$					0.474	0.438			0.437
$F_{\text{LatestCertDayOfWeekSin}}$						0.005			0.005
$F_{\text{LatestCertDayOfWeekCos}}$						0.005			0.005

6.3 Discussion

The results of both the Logistic Regression and Random Forest models on the dataset emphasise that some features contribute more to determining if a domain and X.509 certificate's temporal characteristics indicate if a domain is malicious or benign.

When comparing the results from the two models there is a clear indication that the number of days between domain registration and certificate validity dates (both earliest and latest) correlate with the classification of a domain as shown by the feature sets D & H scoring well on FPR with both models. The f1-scores exhibited for these feature sets were amongst the highest for both malicious and benign predictions in Figure 6-9.

Notably the novel feature proposed in this paper, $F_{\text{CTLogWildcard}}$, contributed to good TPR for feature set B in both models, as indicated by the p-value significance in LR and feature importance scores in RF.

It is noted that these results are based on a dataset size which is comparable in order of magnitude with prior related research (Maroofi *et al.*, 2020; AlSabah *et al.*, 2022) but is not as large as can be obtained given further time and resources (Drichel *et al.*, 2021). The time required to download and process ever increasing volumes of CT Log data requires a compromise where fallback to Certificate Log Monitors is required for missing data and, furthermore, the execution of the lookups during this data processing phase requires a significant amount of time. It would be beneficial, for research and practical use, if CT Log information was available, for example through shared public cloud object storage, rather than requiring large volumes of API calls to multiple providers.

The large volumes of data also hamper reproduction of prior research to make direct comparison of their methods, and this papers approach. Notwithstanding this, the results obtained compare favourably with the FPR rates for CT Log of 9.8% using RF in AlSabah *et al.* (2022) although it is acknowledged that the results do not reach the same levels of f1-score observed by others (Fasllija *et al.*, 2019; Maroofi *et al.*, 2020). These models perform better in this regard, despite not including the temporal features examined in this paper, likely due to their models including a much larger number of potential indicators of maliciousness.

Furthermore, this paper supports the conclusion of Le Pochat *et al.* (2020), by adding further evidence that time based attributes are difficult for attackers to evade.

7 Conclusion and Future Work

The work in this paper demonstrates that temporal features of domain registration and TLS X.509 certificates can be used to accurately identify malicious network activity. Through using well-established machine learning algorithms combined with publicly available WHOIS and CT Log data a model can be trained to identify malware, phishing, and benign domains. The assumption behind this work, that a small set of temporal features can provide insight, holds true in the low FPR scores and reasonable accuracy and f1-scores achieved using RF across all feature set groups. By combining these features with existing models, the state of the art can potentially be advanced to further improve predictions. The novel addition of a feature for the existence of FQDN parent wildcard certificate shows promise, based on significance and importance rankings where it is included in feature sets, as another potential indicator of domain maliciousness.

CT Log volumes, and API limitations, presented a challenge which was only partly overcome and as certificate validity duration decreases future research, and any implementation based on this approach, will require support from the CT group[10] to distribute the logs in a more efficient manner. Obtaining domain registration data is relatively reliable with open-source client libraries but there are edge cases with which additional development would have allowed a more complete dataset to be considered. The advent of the RDAP API has the potential to reduce parsing issues but at the expense of reduced accessibility.

Both these improvements would allow a much larger set of domains to be considered with future research and leveraging cloud network and compute capacity would allow for faster compilation of datasets and feature lookups. There is an important caveat that this additional compute capacity can be offset by more frequent rate limits requiring adaptation of the processes used in this paper.

Consequently, potential future work includes the addition of more detailed integration with the CT Log infrastructure, coordinating cloning of CT log monitors databases, along with examining the use of the RDAP API to reduce the number of domains discarded due to failed feature lookups. Practical implementation of the trained model by integration with a firewall or web proxy is a logical addition to prove the techniques in real world networks.

Finally, from the perspective of FQDNs, the contribution of temporal aspects of record creation may also be a useful indicator, while this paper notes that pDNS restricts the practical accessibility of that information, it may be worthwhile to examine the use of pDNS to obtain temporal features for evaluation in future research.

[10] https://certificate.transparency.dev/

References

Aas, J. *et al.* (2019) ‘Let’s Encrypt: An Automated Certificate Authority to Encrypt the Entire Web’. In *Proceedings of the 2019 ACM SIGSAC Conference on Computer and Communications Security*. CCS ’19. New York, NY, USA: Association for Computing Machinery, pp. 2473–2487. DOI: 10.1145/3319535.3363192.

Abuadbba, A. *et al.* (2022) ‘Towards Web Phishing Detection Limitations and Mitigation’. *ArXiv*, abs/2204.00985. DOI: 10.48550/arXiv.2204.00985.

AlSabah, M. *et al.* (2022) ‘Content-Agnostic Detection of Phishing Domains Using Certificate Transparency and Passive DNS’. In *Proceedings of the 25th International Symposium on Research in Attacks, Intrusions and Defenses*. RAID ’22. New York, NY, USA: Association for Computing Machinery, pp. 446–459. DOI: 10.1145/3545948.3545958.

Anderson, B., Paul, S. and McGrew, D. (2018) ‘Deciphering Malware’s Use of TLS (without Decryption)’. *Journal of Computer Virology and Hacking Techniques*, 14(3), pp. 195–211. DOI: 10.1007/s11416-017-0306-6.

Antonakakis, M. *et al.* (2010) ‘Building a Dynamic Reputation System for DNS’. In *Proceedings of the 19th USENIX Conference on Security*. USENIX Security’10. USENIX Security Symposium. USA: USENIX Association, p. 18.

Antonakakis, M. *et al.* (2011) ‘Detecting Malware Domains at the Upper DNS Hierarchy’. In *Proceedings of the 20th USENIX Conference on Security*. SEC’11. USENIX Security Symposium. USA: USENIX Association, p. 27.

Breiman, L. (2001) ‘Random Forests’. *Machine Learning*, 45(1), pp. 5–32. DOI: 10.1023/A:1010933404324.

Chiba, D. *et al.* (2018) ‘DomainChroma: Building Actionable Threat Intelligence from Malicious Domain Names’. *Computers & Security*, 77, pp. 138–161. DOI: 10.1016/j.cose.2018.03.013.

Darwish, S.M., Farhan, D.A. and Elzoghabi, A.A. (2023) (2) ‘Building an Effective Classifier for Phishing Web Pages Detection: A Quantum-Inspired Biomimetic Paradigm Suitable for Big Data Analytics of Cyber Attacks’. *Biomimetics*, 8(2), p. 197. DOI: 10.3390/biomimetics8020197.

Dong, Z. *et al.* (2015) ‘Beyond the Lock Icon: Real-Time Detection of Phishing Websites Using Public Key Certificates’. In *2015 APWG Symposium on Electronic Crime Research (ECrime)*. 2015 APWG Symposium on Electronic Crime Research (eCrime). pp. 1–12. DOI: 10.1109/ECRIME.2015.7120795.

Dong, Z., Kane, K. and Camp, L.J. (2016) ‘Detection of Rogue Certificates from Trusted Certificate Authorities Using Deep Neural Networks’. *ACM Transactions on Privacy and Security*, 19(2), p. 5:1-5:31. DOI: 10.1145/2975591.

Drichel, A. *et al.* (2021) ‘Finding Phish in a Haystack: A Pipeline for Phishing Classification on Certificate Transparency Logs’. *The 16th International Conference on Availability, Reliability and Security*. DOI: 10.1145/3465481.3470111.

Drury, V., Lux, L. and Meyer, U. (2022) ‘Dating Phish: An Analysis of the Life Cycles of Phishing Attacks and Campaigns’. In *Proceedings of the 17th International Conference on Availability, Reliability and Security*. ARES ’22. New York, NY, USA: Association for Computing Machinery, pp. 1–11. DOI: 10.1145/3538969.3538997.

Drury, V. and Meyer, U. (2019) ‘Certified Phishing: Taking a Look at Public Key Certificates of Phishing Websites’. In *15th Symposium on Usable Privacy and Security (SOUPS’19). USENIX Association, Berkeley, CA, USA*. SOUPS’19. Berkley, CA: USENIX Association, pp. 211–223. Available at: https://www.usenix.org/sites/default/files/conference/protected-files/soups19_slides_drury.pdf (Accessed: 19 June 2023).

European Parliament and Council of the European Union (2016) *Regulation (EU) 2016/679 of the European Parliament and of the Council*. Available at: https://data.europa.eu/eli/reg/2016/679/oj (Accessed: 13 April 2023).

Fasllija, E., Ferit Enişer, H. and Prünster, B. (2019) *Phish-Hook: Detecting Phishing Certificates Using Certificate Transparency Logs*. Available at: https://pure.tugraz.at/ws/portalfiles/portal/25394076/156259641564590.pdf (Accessed: 1 January 2023).

Gallagher, S. (2021) *Nearly Half of Malware Now Use TLS to Conceal Communications*. *Sophos News*. Available at: https://news.sophos.com/en-us/2021/04/21/nearly-half-of-malware-now-use-tls-to-conceal-communications/ (Accessed: 13 February 2023).

Hao, S., Feamster, N. and Pandrangi, R. (2011) ‘Monitoring the Initial DNS Behavior of Malicious Domains’. In *Proceedings of the 2011 ACM SIGCOMM Conference on Internet Measurement Conference*. IMC ’11. New York, NY, USA: Association for Computing Machinery, pp. 269–278. DOI: 10.1145/2068816.2068842.

He, Y. *et al.* (2010) ‘Mining DNS for Malicious Domain Registrations’. In *6th International Conference on Collaborative Computing: Networking, Applications and Worksharing (CollaborateCom 2010)*. CollaborateCom 2010. pp. 1–6.

Heron, S. (2009) ‘Technologies for Spam Detection’. *Network Security*, 2009(1), pp. 11–15. DOI: 10.1016/S1353-4858(09)70007-8.

Holz, T. *et al.* (2008) ‘Measuring and Detecting Fast-Flux Service Networks’.

Le Pochat, V. *et al.* (2020) ‘A Practical Approach for Taking Down Avalanche Botnets Under Real-World Constraints’. In *Proceedings of the 27th Annual Network and Distributed System Security Symposium*. 27th Annual Network and Distributed System Security Symposium, Date: 2020/02/23 - 2020/02/26, Location: San Diego, CA, US. Internet Society. DOI: 10.14722/ndss.2020.24161.

Lu, C. *et al.* (2021) ‘From WHOIS to WHOWAS: A Large-Scale Measurement Study of Domain Registration Privacy under the GDPR’. In *Proceedings 2021 Network and Distributed System Security Symposium*. Network and Distributed System Security Symposium. Virtual: Internet Society. DOI: 10.14722/ndss.2021.23134.

Maroofi, S. *et al.* (2020) ‘COMAR: Classification of Compromised versus Maliciously Registered Domains’. In *2020 IEEE European Symposium on Security and Privacy (EuroS&P)*. 2020 IEEE European Symposium on Security and Privacy (EuroS&P). pp. 607–623. DOI: 10.1109/EuroSP48549.2020.00045.

Mishari, M.A. *et al.* (2012) (arXiv:0909.3688) Available at: http://arxiv.org/abs/0909.3688 (Accessed: 1 April 2023).

Mockapetris, P.V. (1987a) *RFC1034: Domain Names - Concepts and Facilities*. USA: RFC Editor.

Mockapetris, P.V. (1987b) *RFC1035: Domain Names - Implementation and Specification*. USA: RFC Editor.

Page, L. *et al.* (1999) *The PageRank Citation Ranking: Bringing Order to the Web.* Available at: http://infolab.stanford.edu/~backrub/pageranksub.ps (Accessed: 7 April 2023).

PhishTank | Join the Fight against Phishing. Available at: https://phishtank.org/ (Accessed: 1 April 2023a).

Pochat, V.L. *et al.* (2019) 'Tranco: A Research-Oriented Top Sites Ranking Hardened Against Manipulation'. In *Proceedings 2019 Network and Distributed System Security Symposium*. DOI: 10.14722/ndss.2019.23386.

Quinkert, F. *et al.* (2019) 'It's Not What It Looks Like: Measuring Attacks and Defensive Registrations of Homograph Domains'. In *2019 IEEE Conference on Communications and Network Security (CNS)*. 2019 IEEE Conference on Communications and Network Security (CNS). pp. 259–267. DOI: 10.1109/CNS.2019.8802671.

Sakurai, Y. *et al.* (2020) 'Discovering HTTPSified Phishing Websites Using the TLS Certificates Footprints'. *2020 IEEE European Symposium on Security and Privacy Workshops (EuroS&PW)*. DOI: 10.1109/eurospw51379.2020.00077.

Salusky, W. and Danford, R. (2007) 'Know Your Enemy: Fast-Flux Service Networks'. *The Honeynet Project*, pp. 1–24.

Santesson, S. *et al.* (2013) (RFC 6960) *X.509 Internet Public Key Infrastructure Online Certificate Status Protocol - OCSP*. Internet Engineering Task Force DOI: 10.17487/RFC6960.

Scheitle, Q. *et al.* (2018) 'The Rise of Certificate Transparency and Its Implications on the Internet Ecosystem'. In *Proceedings of the Internet Measurement Conference 2018*. IMC '18. New York, NY, USA: Association for Computing Machinery, pp. 343–349. DOI: 10.1145/3278532.3278562.

Spring, J.M., Metcalf, L.B. and Stoner, E. (2011) *Correlating Domain Registrations and DNS First Activity in General and for Malware. In: Securing and Trusting Internet Names: SATIN 2011. National Physical Laboratory (2011)*. Available at: https://discovery.ucl.ac.uk/id/eprint/10037792/ (Accessed: 1 January 2023).

URLhaus | Malware URL Exchange. Available at: https://urlhaus.abuse.ch/ (Accessed: 5 April 2023b).

Weimer, F. (2005) 'Passive DNS Replication'. In FIRST conference on computer security incident. pp. 1–14.

Whittaker, C., Ryner, B. and Nazif, M. (2010) 'Large-Scale Automatic Classification of Phishing Pages'. In Network and Distributed System Security (NDSS) Symposium. San Diego, United States: ISOC.

Wilson, K. (2022) *Revocation Reason Codes for TLS Server Certificates. Mozilla Security Blog.* Available at: https://blog.mozilla.org/security/2022/05/16/revocation-reason-codes-for-tls-server-certificates (Accessed: 3 April 2023).