Harvesting SSL Certificate Data to Identify Web-Fraud

Mishari Almishari, Emiliano De Cristofaro, Karim El Defrawy, and Gene Tsudik
(Corresponding author: Mishari Almishari)

Information and Computer Science, University of California Irvine
ICS Building, Irvine, CA 92697, USA
(Email: malmisha@ics.uci.edu)

(Received Aug. 30, 2011; revised and accepted Oct. 12, 2011)

Abstract

Web-fraud is one of the most unpleasant features of today's Internet. Two well-known examples of fraudulent activities on the web are phishing and typosquatting. Their effects range from relatively benign (such as unwanted ads) to downright sinister (especially, when typosquatting is combined with phishing). This paper presents a novel technique to detect web-fraud domains that utilize HTTPS. To this end, we conduct the first comprehensive study of SSL certificates. We analyze certificates of legitimate and popular domains and those used by fraudulent ones. Drawing from extensive measurements, we build a classifier that detects such malicious domains with high accuracy.

Keywords: SSL, Certificates, Phishing, Typosquatting

1 Introduction

The Internet and its main application – the web – have been growing continuously in recent years. In the last three years, Web contents have doubled in size, from 10 billion to over 20 billion pages [10]. Unfortunately, this growth has been accompanied by a parallel increase in nefarious activities, as reported by [38]. Indeed, the web is an appealing platform for various types of electronic fraud, such as phishing and typosquatting.

Phishing aims to elicit sensitive information – e.g., user names, passwords or credit card details – from unsuspecting users. It typically starts with a user being directed to a fake website with the look-and-feel of a legitimate, familiar and/or well-known one. Consequences of phishing range from denial-of-service to full-blown identity theft, followed by real financial losses. In 2007, more than 3 billion U.S. dollars were lost because of phishing [26].

Typosquatting is the practice of registering domain names that are typographical errors (or minor spelling variations) of addresses of well-known websites (target domains) [51]. It is often related to domain parking services and advertisement syndication, i.e., instructing browsers to fetch advertisements from a server and blending them with content of the website that the user intends to visit [51]. In addition to displaying unexpected pages, typo-domains often display malicious, offensive and unwanted content, or install malware [14, 45]. Certain typo-domains of children-oriented websites were found to even redirect users to adult content [27]. Worse yet, typo-domains of financial websites can serve as natural platforms for *passive* phishing attacks.[1]

Recent studies have assessed popularity of both types of malicious activities, e.g., [3, 41]. In the fall of 2008, McAfee Alert Labs found more than 80,000 domains typosquatting on just the top 2,000 websites [11]. According to the Anti-phishing Working Group (APWG), the number of reported phishing attacks between April 2006 and April 2007 exceeded 300,000 [5]. In August 2009, 56,632 unique phishing websites were reported by APWG – the highest number in APWG's history [4]. Nevertheless, these problems remain far from solved.

Our goal is to counter web-fraud by detecting domains hosting such malicious activities. Our approach is inspired by recent discussions in the web-security community. Security researchers and practitioners have been increasingly advocating a transition to HTTPS for all web transactions, similar to that from Telnet to SSH. Examples of such discussions can be found in [13, 46, 47]. Our work is also a response to an alarming trend observed by a recent study [36]: the emergence of sophisticated phishing attacks abusing SSL certificates. These attacks rely on SSL to avoid raising users' suspicion, by masquerading as legitimate "secure" websites.

This brings us to the following questions:

[1] Passive phishing attacks do not rely on email/spam campaigns to lead people to access the fake website. Instead, users who mistype a common domain name end up being directed to a phishing website.

1) To what extent is HTTPS adopted on the Internet?
2) How *different* are SSL certificates used by web-fraudsters from those of legitimate domains?
3) Can we use information in SSL certificates to identify web-fraud activities, such as phishing and typosquatting, without compromising user privacy?

1.1 Roadmap

First, we measure the overall prevalence of HTTPS in popular and randomly sampled Internet domains. Next, we consider popularity of HTTPS in the context of web-fraud by studying its use in phishing and typosquatting activities. Finally, we analyze, for the first time, all fields in SSL certificates and identify useful features and patterns that can help identify web-fraud.

Leveraging our measurements, we propose a novel technique to identify web-fraud domains that use HTTPS. We construct a classifier that analyzes certificates of such domains. We validate our classifier by training and testing it over data collected from the Internet. The classifier achieves a detection accuracy ranging from 99%, in the worst-case, to 96%. It only relies on data stored in SSL certificates and does not require any user information. Our classifier is orthogonal to prior mitigation techniques and can be integrated with other methods (that do not rely on HTTPS), thus improving overall effectiveness and facilitating detection of a wider range of malicious domains. This might encourage legitimate websites that require sensitive user information (thus, potential phishing targets) to enforce HTTPS.

Finally, we highligh some indirect benefits of HTTPS: it does not only guarantee confidentiality and authenticity, but can also help combat web-fraud.

Paper Organization. The rest of the paper is organized as follows. In Section 2, we briefly overview X.509 certificates. Section 3 presents the rationale and details of our measurements, as well as their analysis. In Section 4, we describe details of a novel classifier that detects malicious domains, based on information obtained from SSL certificates. Section 5 discusses implications of our findings and limitations of our solution. Section 6 overviews related work. Finally, we conclude in Section 7.

2 X.509 Certificates

The term *X.509 certificate* usually refers to an IETF's PKIX Certificate and CRL Profile of the X.509 v3 certificate standard, as specified in RFC 5280 [9]. In this paper we are concerned with the public key certificate portion of X.509. In the rest of the paper, we refer to the public key certificate format of X.509 as a *certificate*, or an *SSL/HTTPS certificate*.

According to X.509, a Certification Authority (CA) issues a certificate binding a public key to an X.500 Distinguished Name (or to an Alternative Name, e.g., an e-mail address or a DNS-entry). Web-browsers – such as Internet Explorer, Firefox, Chrome and Safari – come with pre-installed trusted root certificates. Browser software makers determine which CAs are trusted third parties. Root certificates can be manually removed or disabled, however, it is unlikely that typical users do so. The general structure of an X.509 v3 [9] certificate is presented in Figure 1. As discussed later in Section 3, we analyze *all fields* in certificates collected from both legitimate and malicious domains.

3 Measurements and Analysis of SSL Certificates

In this section, we describe our data sets and our collection methodology. We then present our analysis leading to the design of a classifier that detects web-fraud domains.

3.1 HTTPS Usage and Certificate Harvest

Our measurement data was collected during the following periods:

1) from September to October 2009, and
2) from March to October 2010.

We include three types of domain sets: *Popular, Random and Malicious (Phishing and Typosquatting)*. Each domain was probed twice. We probed each domain for web existence (by sending an HTTP request to port 80) and for HTTPS existence (by sending an HTTPS request to port 443). We harvested the SSL certificate when a domain responded to HTTPS. Table 2 lists our data sets and the number of corresponding SSL certificates.

Popular Domains Data Set. The Alexa [2] top $10,000$ domains were used as our data set for popular domains. Alexa is a subsidiary of Amazon.com known for its web-browser toolbar and for reporting traffic ranking of Internet websites. The company does not provide exact information on the number of users using its toolbar, but it claims several millions [1]. We use Alexa ranking as a popularity measure for legitimate domains. From the top $10,000$ Alexa domains, we collected certificates from $2,984$ different domains (Table 2). Only $2,679$ of these certificates are unique. 13.5% of domains have identical certificates to some other domain in the same data set. Some popular domains belong to the same owner and represent the same company, e.g., `google.co.in` and `google.de`. In this paper, we refer to this data set as Alexa set.

Random Domain Data Set. We randomly sampled .com and .net domains to create a data set that represents a random portion of the web, i.e., allegedly benign but "unpopular" domains. We created this data set to

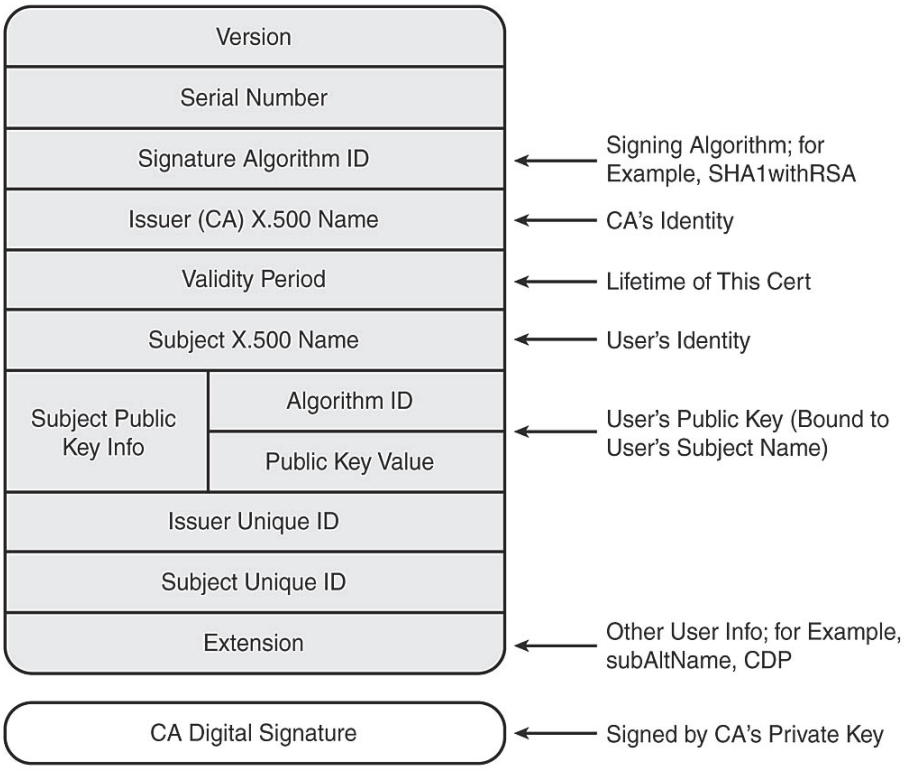

Figure 1: X.509 certificate format.

Table 1: Description of data sets and corresponding number of certificates.

Data set	**Type**	**Number of Certificates**	**Number of Unique Certificates**
Alexa	Popular	2,984	2,679
.com/.net	Random	22,063	16,342
Phishing	Malicious	5,175	2,310
Typosquatting	Malicious	486	100

Table 2: Description of data sets and corresponding number of certificates.

Data set	**Type**	**No. of Certificates**
Alexa	Popular	2,984
.com/.net	Random	22,063
Phishing	Malicious	5,175
Typosquatting	Malicious	486

test whether (or not) certificates of random domains contain similar features to those of either popular or malicious domains. We randomly sampled 100,000 domains from the .com Internet Zone File (and 100,000 from the .net counterpart). The .com/.net Internet Zone Files have been collected from VeriSign [44]. After sampling the Zone Files, we downloaded existing SSL certificates and, as shown in Table 2, we obtained 22,063 different domains with their SSL certificates. Some of these certificates were duplicates(16,342 unique certificates). The percentage of domains having the same certificate as another domain in the same data set in this case is 36.4%. In this paper, we refer to this data set as *.com/*.net set.

3.1.1 Malicious Data Set

Phishing. We collected SSL certificates of 5,175 different domains of phishing urls. The number of unique certificates is 2,310 (Table 2). Phishing domains were obtained from the PhishTank website [33] as soon as they were listed. Reported URLs in PhishTank are verified by several users and, as a result, are malicious with very high probability. We consider this data set as a baseline for phishing domains. The percentage of domains that have an identical certificate as another domain within the data set is 63%. This is a significant increase compared to the popular and random sets. One possible explanation is related to the fact that a large portion of phishing certificates are self signed. These certificates are generated locally and then re-used for several domains under the control of the same malicious entity. In this paper, we refer to this data set as Phishing set.

Typosquatting. In order to collect SSL certificates of typosquatting domains, we first identified the typo domains in our *.com/*.net random domains by using Google's typo correction service [18]. This resulted in 38,617 possible typo domains. However, these might be benign domains that *accidentally* resembled typos of well-known domains. We identified the typosquatting domains in this set by detecting the parked domains among them, using the machine-learning-based classifier proposed in [22].[2] We discovered that 9,830 out of 38,617 were parked domains. We consider these 9,830 names as the data set of typosquatting domains. We then probed these domains to get their SSL certificates.

As reported in Table 2, our Typosquatting data set is composed of 486 domains, i.e., the parked domains having HTTPS and responding with SSL certificates. However, note that only 100 out of 486 certificates are unique. The percentage of domains that have a duplicate certificate with another domain in this data set is 87%. In this paper, we refer to this data set as Typosquatting set.

We acknowledge that the size of the Typosquatting data set is relatively limited. Therefore, we do not claim conclusive results from its analysis, but we include it for completeness. We believe that the limited size of the Typosquatting set is due to the lack of incentives from using HTTPS in this context. Using HTTPS in typosquatting domains does not help in luring users (unlike Phishing). Nonetheless, we believe that, being the first of its kind, such an analysis of HTTPS-enabled typosquatting domains is interesting and shows an initial insight into this fraudulent activity.

3.2 Certificate Analysis

The goal of this analysis is to guide the design of our detection method, i.e., the classifier presented in Section 4. One side-benefit is to reveal differences between certificates used by fraudulent and legitimate/popular domains. In total, we identified 9 relevant certificate features, listed in Table 3.[3] Most features map to actual certificate fields, e.g., F1 and F2. Others are computed from certificate fields but are not directly reflected in the certificate, e.g., host-common-name-sim (F4). Some features are boolean, whereas, others are integers, reals or strings. The computation of features is performed on all the certificates (including the duplicates) as our goal is to identify malicious domains (regardless of its certificates being duplicates or not).

3.2.1 Analysis of Certificate Boolean Features

Features F1-F4 have boolean values, e.g., F1 (md5) is true if the signature algorithm used in the certificate is "md5WithRSAEncryption." The results of analyzing these features are summarized in Table 4. This analysis reveals interesting and unexpected issues and differences between legitimate and malicious domains.

F1 (md5)
9.9% of Alexa certificates use "md5WithRSAEnc ryption", much less than those in *.com/*.net (27.4%), Phishing (17.4%) and Typosquatting (26.1%). Note that rogue certificates can be constructed using MD5 ([39, 40]). The difference in the percentages may indicate that many of *.com/*.net, Phishing and Typosquatting certificates are issued without a significant effort to check their security.

F2 (bogus subject)
A bogus subject indicates whether the subject fields have some bogus values (e.g., "ST=somestate", "O=someorganization", "CN=localhost", ...). We identified a list of such meaningless values and considered subjects to be *bogus* if they contain one of these

[2]Note that typosquatters profit from traffic that accidentally comes to their domains. One common way of achieving this is to host typosquatting domains from a domain parking service. Parked domains [17] are ads-portal domains that show ads provided by a third-party service called a parking service, in the form of ads-listing [51] so typosquatters may profit from incoming traffic (e.g., if a visitor clicks on a sponsored link).

[3]Other features and fields (e.g., RSA exponent, Public Key Size, ...) had no substantial differences between legitimate and malicious domains and we omit them here.

Table 3: Features extracted from SSL certificates.

Feature	Name	Type	Notes
F1	md5	boolean	The signature algorithm of the certificate is md5WithRSAEncryption
F2	bogus subject	boolean	The subject section of the certificate is clearly bogus
F3	self-signed	boolean	The certificate is self-signed
F4	host-common-name-sim	boolean	Whether the common name of the subject and domain name of the certificate has the same basic part of the domain
F5	issuer common	string	The common name of the issuer
F6	issuer organization	string	The organization name of the issuer
F7	issuer country	string	The country name of the issuer
F8	subject country	string	The country name of the subject
F9	validity duration	integer	The validity period in days

Table 4: Analysis of boolean certificate features (percentages satisfying each feature).

Feature	Alexa	*.com/*.net	Phishing	Typosquatting
F1	9.9%	27.4%	17.4%	26.1%
F2	7.7%	11.8%	18.3%	29.8%
F3	15.8%	35.4%	30.6%	53.5%
F4	74.3%	10.4%	9.8%	0%

Table 5: APCR values.

Feature	Sets	APCR
issuer common name	Phishing Alexa	52.7%
issuer common name	Phishing *.com/*.net	51.2%
issuer organization name	Phishing Alexa	57.7%
issuer organization name	Phishing *.com/*.net	46%
issuer country	Phishing Alexa	28.4%
issuer country	Phishing *.com/*.net	18.6%
subject country	Phishing Alexa	26.5%
subject country	Phishing *.com/*.net	24%

values. 7.7% of Alexa certificates satisfy this feature (similar percentage in *.com/*.net). This percentage is much higher (18.3%) in Phishing and Typosquatting (29.8%). This indicates that web-fraudsters fill subject values with bogus data or leave default values when generating certificates.

F3 (self-signed)
15.8% of Alexa certificates are self-signed, somehow unexpectedly. One possible explanation is that some of the popular domains in the Alexa set represent companies having their own CA and issuing their own certificates (e.g., google, microsoft, yahoo ...etc.). The percentages of self-signed certificates in *.com/*.net, Phishing, and Typosquatting is higher (resp., 35.4%, 30.6% and 53.5%). This is expected for Phishing, since miscreants would like to avoid leaving any trace by obtaining a certificate from a CA (which requires documentation and a payment). For *.com/*.net, a higher percentage could be explained by the use of locally generated certificates, e.g., using OpenSSL [31]. The difference between Alexa on one side and Phishing, Typosquatting, and *.com/*.net on the other side is quite significant (more than 14%).

F4 (host-common-name-sim)
We expect the common name in the subject field to be very similar to the hostname in popular domain certificates, while we intuitively expect this to be lower in malicious domain certificates, e.g., because malicious domains may not use complying SSL certificates. In order to assess this, we define a feature called "host-common-name sim". This feature measures the similarity between domain name of the SSL certificate and common name of the subject field in it. For instance, if the hostname and common name are `www.google.com.sa` and `google.com` respectively, host-common-name-sim is set to true. Whereas, if the hostname and common name are equal to `www.domain-x.com` and `www.domain-y.com`, respectively, the feature is set to false since domain-x is not equal to domain-y.

74.3% of Alexa certificates satisfies this feature. The percentages in *.com/*.net, Phishing, and Typosquatting are 10.4% and 9.8%, 0% respectively. The difference between Alexa on one side and Phishing, Typosquatting, and *.com/*.net on the other side is quite remarkable. This feature, together with the previous one (excluding bogus subject), suggests that there is some strong similarity among *.com/*.net, Phishing, and Typosquatting certificates.

3.2.2 Analysis of Certificate Non-Boolean Features

We now present the analysis of non-boolean features. F5–F6 are related to the certificate issuer: common name, organization name and country, while F7–F8 are related to the country issuer or country subject and F9 is related the validity duration.

F5 (issuer common name) and F6 (issuer organization)
First, we noticed that some values of issuers' common names are popular in only one domain set. For example, 10.1% of Phishing certificates are issued by UTN-USERFIRST-HARDWARE (only 4.2(5.3)% in Alexa(*.com/*.net)). Similarly, some issuers' organization names are popular only in one domain set. For example, 9.6% of Phishing certificates have COMODO-CA-LIMITED as their issuer's organization name, as opposed to 2.1% in Alexa and 3.2% in *.com/*.net certificates.

To elaborate more on the difference of the issuer common name between any two domain data sets, **A** against **B**, we extract all issuer common names that are more popular(in terms of ratio) in set **A** than in **B**. We name these popular common names **Popular-Issuer-Common-Name-A-B**. Then, we compute what percentage of the certificates in **A** and **B** that have their issuer common name in **Popular-Common-Name-A-B**. In a similar way, we extract **Popular-Issuer-Organization-A-B**, **Popular-Issuer-Country-A-B**, and **Popular-Subject-Country-A-B** sets corresponding to issuer organization, issuer country, and subject country. Table 6 shows the ratios when **A** is Phishing and **B** is Alexa or *.com/*.net. For issuer common name(issuer organization name), we can clearly notice the difference in ratios between Phishing and Alexa(*.com/*.net) which emphasizes on the differences of issuer common name(issuer organization name) feature among the domain sets (The ratios when **A** is Alexa or *.com/*.net suggest similar conclusions).

To quantify the difference in issuer's common/organization name, we measure change in the posterior probabilities of phishing certificates, i.e., the probability that a certificate is phishing given the common/organization name is equal to a specific value. To this end, we merge Phishing and Alexa sets and observe how the posterior probability changes. Similarly, we merge Phishing and *.com/*.net sets. In order to measure the changes in the posterior probability, we borrow the metric in [22], i.e., called Average Posterior Change Ratio $APCR$. In the following we define $APCR$. Note that P_p stands for posterior probability of phishing certificates and P_a stands for prior probability of being a phishing certificate. In the following, I stands for a common name issuer value [4].

[4]The $APCR$ defined for issuer's common name can be similarly defined for issuer's organization name, issuer's country, and subject country.

Table 6: The table shows the ratios of Popular-Issuer-Common-Name-A-B, Popular-Issuer-Organization-A-B, Popular-Issuer-Country-A-B and Popular-Subject-Country-A-B in set A and B. In this table, A is Phishing and B is either Alexa or *.com/*.net.

Feature	Sets	Ratio_A	Ratio_B
issuer common name	Phishing Alexa	72%	23%
issuer common name	Phishing *.com/*.net	79%	31%
issuer organization name	Phishing Alexa	88%	33%
issuer organization name	Phishing *.com/*.net	76%	34%
issuer country	Phishing Alexa	92%	65%
issuer country	Phishing *.com/*.net	33%	15%
subject country	Phishing Alexa	55%	30%
subject country	Phishing *.com/*.net	60%	28%

$$APCR = \sum_{\forall I} P_p \ Change \ Ratio \ in \ I \times \ Fraction \ of \ Certificates \ in \ I \quad (1a)$$

$$where:$$

$$P_p \ Change \ Ratio \ in \ I = \frac{|P_p \ in \ I - P_a|}{Max \ P_P \ Displacement \ in \ I} \quad (1b)$$

$$and:$$

$$Max \ P_p \ Displacement \ in \ I = \begin{cases} 1 - P_a & \text{if } P_p in \ I > P_a \\ P_a & \text{otherwise} \end{cases} \quad (1c)$$

The Posterior Change Ratio measures relative change of posterior probability P_p from the prior probability, P_a. For instance, a value of 0 indicates no relative change and a value of 1 indicates the largest relative change. Note that the average is weighted, such that more "weight" is given to common names corresponding to more certificates (For more information, readers can refer to [22]).

Table 5 shows the $APCR$ of for issuer's common name. When we merge Phishing and Alexa sets, we obtain an $APCR$ value of 52.7% (51.2% for Phishing and *.com/*.net). This indicates that change in posterior probability is quite significant. Table 5 shows similar $APCR$ values for issuer's organization name.

F6 and F7 (issuer and subject countries)
Similarly, some of issuers' country names are popular only in one data set. For example, 10.3% of Phishing certificates have GB as their issuer's country name (only 1.4% in Alexa and 4.3% in *.com/*.net). Additionally, some of the subject country names are popular only in the Phishing set. For example, 5.3% of Phishing certificates have FR as their country in the subject field. This happens only in 2.2% of Alexa certificates (0.9% in *.com/*.net).

Table 6, shows the ratios of domains that have their certificates issuer country(subject country) in **Popular-Issuer-Country-A-B** (**Popular-Subject-Country-A-B**) set where **A** is Phishing and **B** is Alexa or *.com/*.net. The ratios in Phishing and Alexa(*.com/*.net) have significant differences which emphasizes on the differences of issuer country(subject country) feature among the domain sets. Table 5 shows the $APCR$ of the issuer country feature. When we combine the Phishing and Alexa sets, we obtain an $APCR$ value of 28.4% (18.6% for the Phishing and *.com/*.net sets). This shows that there is a change in the posterior probability, however, not as significant as for the issuer common/organization name. Table 5 also shows similar $APCR$ values for subject country feature which suggest similar conclusions.

F9 (validity duration)
Table 7, shows the distribution of domains over different duration periods. As shown in the Table, domains sets have different ratios for different periods. For example, Phishing has the largest ratio when the period is less than a year, *.com/*.net when the period is in [365-729] days, and Alexa when the period is in [730-1094] days. Additionally, we plot the CDF of the certificate duration for different data sets in Figure 2 which elaborates more on the differences.

3.2.3 Summary of Certificate Feature Analysis

We now highlight the most important observations from our analysis:

1) For most of the features, distributions of malicious certificates are significantly different from Alexa certificates. Therefore, popular domains can be easily differentiated from malicious domains based on their SSL certificates.

2) A self signed certificate is more likely to be for popular domain. The percentage of self signed certificates in popular domains is only 15.8%.

3) We observe strong similarities between the *.com/*.net set and malicious sets in many features. One reason is that certificates in both sets may be issued without applying appropriate control, as opposed to certificates obtained from popular

Table 7: The table shows the distribution of domains over different duration periods for different domain sets.

Feature	Alexa	*.com/*.net	Phishing
0 - 364	2.5%	4.2%	6.3%
365 - 729	51.7%	62.3%	49.6%
730 - 1094	26.1%	16.1%	20.1%
1095 - 3649	15.9%	11.1%	10.4%
3649 -	3.8%	6.3%	13.5%

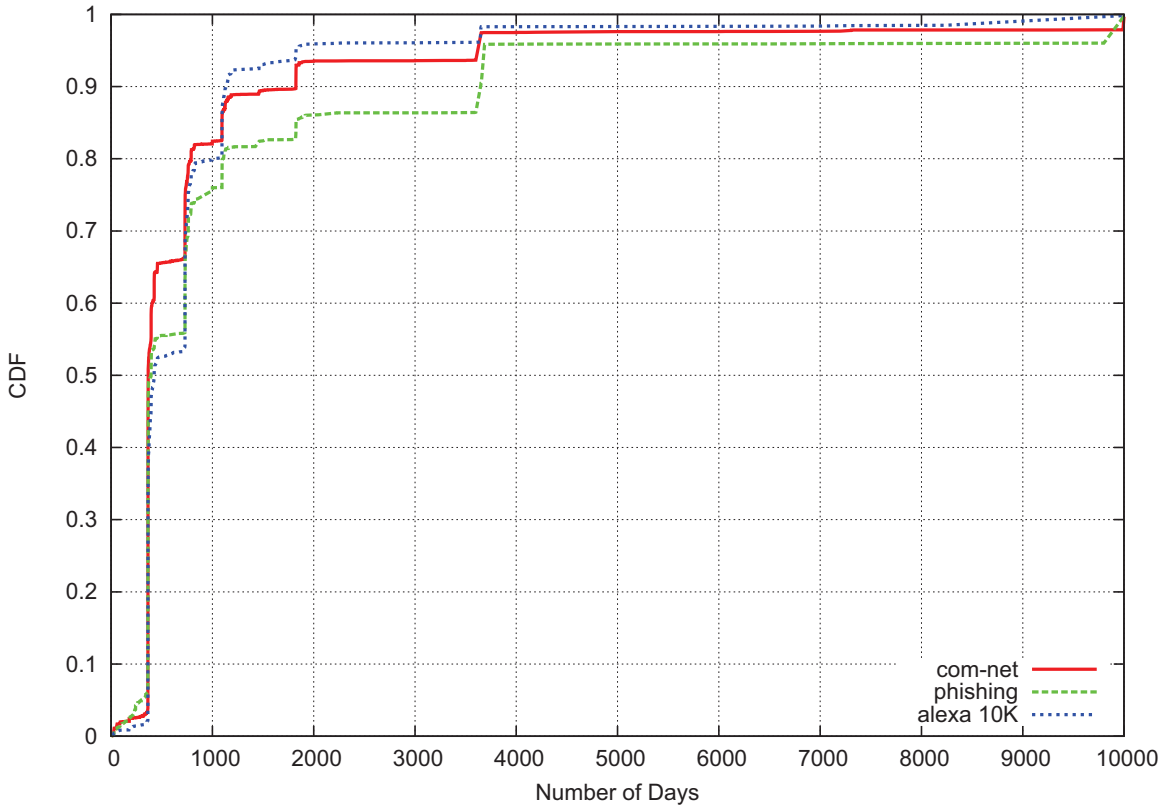

Figure 2: CDF of certificate duration.

Alexa domains. Another possible explanation is that a large portion of certificates from Phishing Typosquatting and *.com/*.net domains are locally generated (e.g., self-signed) and use the same default values from the employed software (e.g., OpenSSL).

4) Host-common-name-sim feature is a very discriminative feature in differentiating between popular and other domains.

4 Certificate-based Classifier

The analysis above shows that several features have distributions that vary among different data sets. Relying on a single feature to identify malicious certificates will yield a high rate of false positives. Therefore, we combine all features and feed them to a set of machine learning classifiers in order to differentiate among certificates belonging to different data sets. We use several machine-learning-based classification algorithms and select the one with the best performance. Specifically, we consider the following algorithms: Random Forest [8], Decision Tree [34], and Nearest Neighbor [28]. In addition, we explore two optimization techniques for Decision Trees: Bagging [7, 35] and Boosting [16, 35].

Nearest Neighbor classifier does not have a training phase and it simply labels the incoming test record to the most similar training record. Decition Tree classifier builds a tree that classifies a record by following a path from the root to the leaf. To select a specific branch, the record has to satisfy its condition. The Random Forest classifier is derived from a multiple of Decision Trees each of which has some selection of the feature set. Since, detailed algorithm descriptions are out of the scope of this paper, we refer to [7, 16, 28, 35] for relevant background information.

We use precision-recall performance metrics to summarize performance of a classifier. These metrics consists of the following ratios: Positive Recall, Positive Precision, Negative Recall, and Negative Precision. We use the term "Positive" set to denote Phishing (and *.com/*.net in one case) certificates and "Negative" set to refer to Alexa (popular) certificates. The positive (negative) recall value is the fraction of positive (negative) domains that are correctly identified by the classifier. Positive (negative) precision is the actual fraction of positive (negative) domains in the set identified by the classifier.

We evaluate the performance of the classifier using the ten-fold cross validation method [28]. We randomly divide the data set into 10 sets of equal size, perform 10 different training/testing steps where each step consists of training a classifier on 9 sets, and then we test it on the remaining set. Finally, we average all results to assess the final performance of the classifier.

4.1 Classifier Features

For feature F5 - issuer common name, we extract 6 boolean sub-features each of which corresponds to whether the domain certificate issuer common name belongs to **Popular-Issuer-Country-A-B** for a specific pair selection (**A**, **B**). We try all the possible pair combinations; specifically, (Phishing, Alexa), (Phishing, *.com/*.net), (Alexa,*.com/*.net), (Alexa,Phishing), (*.com/*.net,Alexa), and (*.com/*.net,Phishing). This gives us the boolean sub-features **issuer-common-name-A-B**. Similarly, we extract 6 sub-features from F6, F7, and F8 giving us **issuer-organization-A-B**, **issuer-country-A-B**, and **subject-country-A-B** sub-features for 6 different pairs of (A,B). It is these sub-features that we use in the classifiers instead of F5-F8.

For feature F9 - validity duration, we convert it to a nominal feature before feeding it to the classifier. That is, for each domain validity duration, we convert it to a nominal value by assigning it one of five values corresponding to different duration periods shown in Table 7.

For the other features, we use them as they are. Note that in the classifier the Negative set is Alexa and the positive set could be either Phishing or Phishing and *.com/*.net. When the positive set is both *.com/*.net and Phishing we use F1-F4, sub-features of F5-F8, and the converted version of F9. When the positive set is only Phishing, we use the same features but we exclude all the sub-features that has *.com/*.net in the pair combination.

4.2 Classifier Results

We first train the classifier on a data set that consists of Alexa and Phishing certificates. The purpose is to train the classifier to differentiate malicious from popular certificates. Performance results of different classifiers are shown in Table 8. Most of the classifiers have comparable performance results and the phishing detection accuracy could be as high as 88%.

Since *.com/*.net certificates have similar distributions as Phishing certificates in a number of features, we also build a classifier that differentiates between very popular certificates (Alexa set) and non popular sets (*.com/*.net and Phishing certificates). The Positive data set consists of all certificates from *.com/*.net and Phishing sets. The Negative data set consists of all certificates in the Alexa set. One can regard a certificate in the Positive set as a certificate issued with due diligence, unlike one in the Negative set. Thus, this classifier differentiates "neat" certificates from "sloppy" ones, indicating that the corresponding domain might be malicious. The results of our training is shown in Table 9. Note that all classifiers have relatively similar results and the malicious detection accuracy could be as high as 96%.

Similar to machine-learning-based spam filtering solutions, a larger training data set results in better per-

Table 8: Performance of classifiers - data set consists of: (a)positive: Phishing certificates and (b) negative: Alexa certificates.

Classifier	Positive Recall	Positive Precision	Negative Recall	Negative Precision
Random Forest	0.94	0.88	0.778	0.881
Decision Tree	0.939	0.881	0.779	0.88
Bagging - Decision Tree	0.935	0.877	0.773	0.874
Boosting - Decision Tree	0.94	0.881	0.78	0.882
Nearest Neighbor	0.94	0.879	0.774	0.882

Table 9: Performance of classifiers - data set consists of: (a)positive: *.com/*.net and Phishing certificates and (b) negative: Alexa certificates.

Classifier	Positive Recall	Positive Precision	Negative Recall	Negative Precision
Random Forest	0.974	0.958	0.61	0.72
Decision Tree	0.975	0.958	0.611	0.727
Bagging - Decision Tree	0.972	0.96	0.631	0.708
Boosting - Decision Tree	0.97.4	0.95.7	0.604	0.719
Nearest Neighbor	0.975	0.957	0.598	0.725

Table 10: Performance of Classifiers - data set consists of: (a)positive: Phishing certificates (500, 1000, 2000) and (b) negative: Alexa certificates.

Classifier	Positive Recall	Positive Precision	Negative Recall	Negative Precision
Decision Tree (500)	0.732	0.819	0.973	0.956
Decision Tree (1000)	0.817	0.827	0.943	0.939
Decision Tree (2000)	0.826	0.875	0.921	0.888

formance. We acknowledge that our solution would incur false positives when actually deployed. However, the number of false positives can be reduced by training on larger data sets and constantly updating training samples (See Section 4.3).

The best performing classifiers in both cases can be used to label domains by classifying their certificates (labels are: phishing and legitimate in the first classifier, and suspicious and non-suspicious in the second classifier.). The classifiers work by first extracting the features from the SSL certificates and then feeding them to the classifier model which gives the labels as a result. We envision our solution as a part of larger phishing mitigation system and the role of our solution is just to label the domains with what it thinks about the domains certificates. Thus, the results of our classifiers are served as inputs to a larger phishing mitigation system that would combine our recommendations with other observations to provide an overall more accurate judgment. This larger phishing mitigation system can be either at the client machines doing the mitigation as the user browses the Internet or at the server machines taking a preemptive investigation of some suspicious domains to either blacklist them or block them in case they turn out to be phishing.

4.3 Classifier Results with Different Set Sizes

To verify how the data set size would affect the classifier performance, we use as the negative set the same Alexa we use in the previous section. But for the positive set, we use 3 Phishing sets of different sizes; namely, 500, 1000, and 2000. The ten-fold cross validation results of Decision Tree is shown in Table 10. We can see as we increase the size of the positive set, positive precision and recall increase. Thus, a large size of Phishing set is essential in having a high Phishing detection accuracy.

4.4 Classifier Results With Minimal Set of Features

To see how the classifier perform when we use a feature set that is less vulnerable to being manipulated, we restrict the feature set to only the sub-features related to the issuer; namely, issuer common name, issuer organization and issuer country. We use Alexa as our negative set and Phishing as our positive set. Table 11 shows the results of the Decision Tree (the others are of comparable performance) when we only use the issuer related sub-features. Even though we use less features, we still get reasonably good positive precising and recall.

5 Discussion

Based on measurements presented in previous sections, we find that a significant percentage of well-known domains *already* use HTTPS. It is possible to harvest their certificates for our classification purpose, without requiring any modifications on the domains' side. Furthermore, the non-trivial portion of phishing websites utilizing HTTPS highlights the need to analyze and correlate information provided in their certificates.

Using information in certificates.
Our results show significant differences between certificates of popular domains and those of malicious domains. Not only is this information alone sufficient to detect fraudulent activities as we have shown, but it is also a useful component in assessing a website's degree of trustworthiness, thus improving prior metrics, such as [15, 21, 50]. Our method should be integrated with other techniques to improve the effectiveness of detecting malicious domains.

Keeping state of encountered certificates.
We deliberately chose to conduct our measurements as general as possible, without relying on user navigation history or on user specific training data. These components are fundamental for most current mitigation techniques [15, 32]. Moreover, we believe that keeping track of navigation history is detrimental to user privacy. However, our work yields effective detection by analyzing certain coarse-grained information extracted from server certificates and not specific to a user's navigation patterns. This does not violate user privacy as keeping fine-grained navigation history would.

Limitations.
We acknowledge that additional data sets of legitimate domains need to be taken into consideration, e.g. popular websites from DNS logs in different organizations and countries. Data sets of typosquatting domains can be strengthened by additional and more effective name variations. Also, we acknowledge that our phishing classifier may incur false positives when actually deployed. However, this is a common problem to many machine-learning-based mitigation solutions (e.g., spam filtering and intrusion detection based on machine-learning techniques) and the number of false positives can be minimized by training the classifier on larger and more comprehensive data sets. Our classifier does not provide a complete standalone solution to the phishing threat since many domains do not have HTTPS. Instead, integrated with pre-existing solutions (e.g., [15, 21, 50]), it improves their effectiveness in the context of potentially critical applications enforcing HTTPS.

How malicious domains will adapt.
Web-fraudsters are diligent and quickly adapt to new security mechanisms and studies that threaten their business. We hope that this work will raise the bar and make it more difficult for web-fraudsters to deceive users. If web-browsers use our classifier to an-

Table 11: Performance of Classifiers - data set consists of: (a)positive: Phishing certificates and (b) negative: Alexa certificates.

Classifier	Positive Recall	Positive Precision	Negative Recall	Negative Precision
Decision Tree	0.895	0.823	0.667	0.785

alyze SSL certificate fields, we expect one of two responses from web-fraudsters: (1) to acquire legitimate certificates from CAs and leave a paper trail pointing to "some person or organization" which is connected to such malicious activities, (2) to craft certificates that have similar values to those that are most common in certificates of legitimate domains. Some fields/features will be easy to forge with legitimate values (e.g., country of issuer, country of subject, subject common and organization name, validity period, signature algorithm, serial number ...etc). For some other fields this will not be possible (issuer name, signature ...etc) because otherwise the verification of the certificate will fail. In either case the effectiveness of web-fraud will be reduced. Additionally, we show in Section 4.4 how the classifier still performs well when only relying on issuer related features.

6 Related Work

The work in [19] conducted a study to measure the cryptographic strength of more than 19,000 public servers running SSL/TLS. The study reported that many SSL sites still supported the weak SSL 2.0 protocol. Also, most of the probed servers supported DES, which is vulnerable to exhaustive search. Some of the sites used RSA-based authentication with only 512-bit key size, which is insufficient. Nevertheless, it showed encouraging measurement results, e.g., the use of AES as default option for most of the servers that did not support AES. Also, a trend toward using a stronger cryptographic function has been observed over two years of probing, despite a slow improvement. In [37], the authors performed a six-month measurement study on the aftermath of the discovered vulnerability in OpenSSL in May 2008, in order to measure how fast the hosts recovered from the bug and changed their weak keys into strong ones. Through the probing of thousands of servers, they showed that the replacement of the weak keys was slow. Also, the speed of recovery was shown to be correlated to different SSL certificate characteristics such as: CA type, expiry time, and key size. The article in [29] presented a profiling study of a set of SSL certificates. Probing around 9,754 SSL servers and collecting 8,081 SSL certificates, it found that around 30% of responding servers had weak security (small key size, supporting only SSL 2.0,...), 10% of them were already expired and 3% were self-signed. Netcraft [30] conducts a monthly survey to measure the certificate validity of Internet servers. Recently, the study showed that 25% of the sites had their certificates self-signed and less than half had their certificates signed by valid CA. Symantec [36] has observed an increase in the number of URLs abusing SSL certificates. Only in the period between May and June 2009, 26% of all the SSL certificate attacks have been performed by fraudulent domains using SSL certificates. Although our measurement study conducts a profiling of SSL certificates, our purpose is different from the ones above. We analyze the certificates to show how malicious certificates are different from benign ones and to leverage this difference in designing a mitigation technique.

The importance and danger of web-fraud (such as phishing and typosquatting) has been recognized in numerous prior publications, studies and industry reports mainly due to the tremendous financial losses [26] that it causes. One notable study is [3] which analyzed the infrastructure used for hosting and supporting Internet scams, including phishing. It used an opportunistic measurement technique that mined email messages in real-time, followed the embedded link structure, and automatically clustered destination websites using image shingling. In [15], a machine learning based methodology was proposed for detecting phishing emails. The methodology was based on a classifier that detected phishing with 96% accuracy and false negative rate of 0.1%. Our work differs since it does not rely on phishing emails which are sometimes hard to identify. An anti-phishing browser extension (AntiPhish) was given in [21]. It kept track of sensitive information and warned the user whenever the user tried to enter sensitive information into untrusted websites. Our classifier can be easily integrated with AntiPhish. However, AntiPhish compromised user privacy by keeping state of sensitive data. Other anti-phishing proposals relied on trusted devices, such as a user's cell phone in [32]. In [23], the authors tackled the problem of detecting malicious websites by only analyzing their URLs using machine-learning statistical techniques on the lexical and host-based features of their URLs. The proposed solution achieved a prediction accuracy around 95%. Other studies measured the extent of typosquatting and suggested mitigation techniques. Wang, et al. [51] showed that many typosquatting domains were active and parked with a few parking services, which served ads on them. Similarly, [6] showed that, for nearly 57% of original URLs considered, over 35% of all possible URL variations existed on the Internet. Surprisingly, over 99% of such similarly-named websites were considered phony. [22] devised a methodology for identifying ads-portals and

parked domains and found out that around 25% of (two-level) .com and .net domains were ads-portals and around 40% of those were typosquatting. McAfee also studied the typosquatting problem in [20]. A set of 1.9 million single-error typos was generated and 127,381 suspected typosquatting domains were discovered. Alarmingly, the study also found that typosquatters targeted children-oriented domains. Finally, McAfee added to its extension site advisor [25] some capabilities for identifying typosquatters. In [24], the authors proposed a technique to counter-against phishing and pharming attacks that is based on mutual authentication which can be easily adopted in the current systems. In [43], the authors proposed an anomaly detection technique that is based on hidden Markov models which is very suitable to Windows environment.

To the best of our knowledge, no prior work has explored the web-fraud problem in the context of HTTPS and proposed analyzing server-side SSL certificates in more detail. Our work yields a detailed analysis of SSL certificates from different domain families and a classifier that detects web-fraud domains based on their certificates.

Finally, some usable security studies have attempted to measure effectiveness of available anti-phishing tools and security warnings. User studies analyzing the effectiveness of browser warning messages indicated that an overwhelming percentage (up to 70-80% in some cases) of users ignored them. This observation—confirmed by recent research results (e.g., [12] and [42])— might explain the unexpected high percentage of expired and self signed certificates that we found in the results of all our data sets. Furthermore, [52] evaluated available anti-phishing tools and points out that only 1 out of 10 popular anti-phishing tool identified more than 90% of phishing URLs correctly. Also, [49] pointed out that users failed to pay attention to the toolbar or explained away tool's warning if the content of web pages looked legitimate. Similarly, [48] highlighted, through eyetracker data, that users commonly looked at lock icons, but rarely used the certificate information. As a result, one may wonder about the incentive for phishers and typosquatters to utilize SSL certificates. We believe that our classifier can be used together with available tools, e.g., AntiPhish [21], which keeps track of sensitive information and warns the user whenever users enter sensitive information into insecure websites. In this case, phishers need to provide SSL certificates to bypass the AntiPhish block.

7 Conclusion

In this paper, we study the prevalence of HTTPS in popular and legitimate domains as well as in the context of web-fraud, i.e., phishing and typosquatting. To the best of our knowledge, this is the first effort to analyze information in SSL certificates to profile domains and assess their degree of trustworthiness. We design and build a machine-learning-based classifier that identifies fraudulent domains that utilize HTTPS. The classifier solely relies on SSL certificates of such domains, thus preserving user privacy. Our work can be integrated with existing detection techniques to improve their effectiveness. Finally, we believe that our results may serve as a motivation to increase the adoption of HTTPS. We believe that aside from its intended benefits of confidentiality and authenticity, HTTPS can help identify web-fraud domains.

References

[1] Alexa.com. "Alexa ranking methodology". http://www.alexa.com/help/traffic_learn_more/

[2] Alexa.com. "The Web Information Company". http://www.alexa.com

[3] D. S. Anderson, C. Fleizach, S. Savage, and G. M. Voelker, "Spamscatter: Characterizing internet scam hosting infrastructure," in *USENIX Security Symposium*, pp. 1–14, 2007.

[4] Anti Phishing Working Group, *Phishing Activity Trends Report – 3rd Quarter 2009.* http://www.antiphishing.org/reports/apwg_report_Q3_2009.pdf

[5] Anti Phishing Working Group, *Phishing Activity Trends Report – April 2007.* http://www.antiphishing.org/reports/apwg_report_april_2007.pdf

[6] A. Banerjee, D. Barman, M. Faloutsos, and L. N. Bhuyan, "Cyber-Fraud is One Typo Away," in *INFOCOM 2008*, pp. 1939-1947, 2008.

[7] L. Breiman, "Bagging predictors," *Machine Learning*, vol. 24, no. 2, pp. 123–140, 1996.

[8] L. Breiman, "Random Forests," *Machine Learning*, vol. 45, no. 1, pp. 5–32, 2001.

[9] D. Cooper and et al, *Internet X.509 Public Key Infrastructure Certificate and CRL Profile*, (IETF RFC5280). http://www.ietf.org/rfc/rfc5280.txt

[10] M. D. Kunder, *World Wide Web Size*, 2010. http://www.worldwidewebsize.com/

[11] B. Edelman, "McAUnintended adventures in browsing," *Mcafee Security Journal*, 2008. http://www.mcafee.com/us/local_content/misc/threat_center/msj_unintended_adventures_browsing.pdf

[12] S. Egelman, L. F. Cranor, and J. Hong, "You've been warned: an empirical study of the effectiveness of web browser phishing warnings," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems* , pp. 1065–1074, 2008.

[13] Electronic Frontier Foundation, *HTTPS Everywhere*, 2010. https://www.eff.org/https-everywhere

[14] F-Secure, *Googkle.com Installed Malware by Exploiting Browser Vulnerabilities*, 2009. http://www.f-secure.com/v-descs/googkle.shtml

[15] I. Fette, N. Sadeh, and A. Tomasic, "Learning to Detect Phishing Emails," in *Proceedings of the 16th international conference on World Wide Web*, pp. 649-656, 2007.

[16] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," in *European Conference on Computational Learning Theory*, pp. 23-37, 1995.

[17] Google.com, *Parked Domain Site.* http://adwords.google.com/support/aw/bin/answer.py?hl=en&answer=50002.

[18] Google.com, *The Google Spell Checker.* http://www.google.co.uk/help/features.html.

[19] H. Lee and T. Malkin and E. Nahum, "Cryptographic strength of SSL/TLS servers: Current and recent practices," in *Internet Measurement Comference*, pp. 83-92, 2007.

[20] S. Keats, *What's In A Name: The State of Typo-Squatting, 2007.* http://www.siteadvisor.com/studies/typo_squatters_nov2007.html

[21] E. Kirda and C. Kruegel, *Protecting Users Against Phishing Attacks*, Oxford University Press, 2005.

[22] M. Almishari and X. Yang, "Text-based ads-portal domains: Identification and measurements," *ACM Transactions on the Web*, vol. 4, no. 2, artice 4, 2010.

[23] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker, "Beyond blacklists: learning to detect malicious web sites from suspicious URLs," in *KDD 2009*, pp. 1245-1254, 2009.

[24] A. S. Martino and X. Perramon, "Phishing secrets: History, effects, and countermeasures," *International Journal of Network Security*, vol. 11, no. 3, pp. 163-171, 2010.

[25] McAfee, *McAfee SiteAdvisor.* http://www.siteadvisor.com/.

[26] T. McCall, "Gartner survey shows phishing attacks escalated in 2007; more than $3 billion lost to these attacks," 2007. http://www.gartner.com/it/page.jsp?id=565125

[27] Microsoft Researc/, *Screenshots of Questionable Advertisements*, 2006. http://research.microsoft.com/Typo-Patrol/screenshots.htm

[28] T. Mitchell, *Machine Learning*, McGraw Hill, 1997.

[29] E. Murray, *SSL Server Security Survey.* http://web.archive.org/web/20031005013455/http://www.lne.com/ericm/papers/ssl_servers.html

[30] Netcraft, *Netcraft SSL Survey*, 2008. http://news.netcraft.com/SSL-Survey

[31] OpenSSL, *The OpenSSL Project*, 2007. http://www.openssl.org/.

[32] B. Parno, C. Kuo, and A. Perrig, "Phoolproof phishing prevention," in *Proceedings of the 10th International Conference on Financial Cryptography and Data Security*, pp. 1-19, 2006.

[33] Phishtank.com. http://www.phishtank.com

[34] J. R. Quinlan, *c4.5: Programs for Machine Learning.* Morgan Kaufmann, 1993.

[35] J. R. Quinlan, "Bagging, boosting, and c4.5," in *13th National Conference on Artificial Intelligence and 8th Innovative Applications of Artificial Intelligence Conference*, pp. 725-730, AAAI Press, 1996.

[36] Z. Raza, *Phishing Toolkit Attacks are Abusing SSL Certificates*, 2009. http://www.symantec.com/connect/blogs/phishing-toolkit-attacks-are-abusing-ssl-certificates

[37] S. Yilek, E. Rescorla, H. Shacham, B. Enrigh, and S. Savage, "When private keys are public: Results from the 2008 debian openssl vulnerability," in *Proceedings of the 9th ACM SIGCOMM Conference on Internet Measurement Conference (IMC 2009)*, pp. 15-29, 2009.

[38] D. M. Sena, *Symantec Internet Security Threat Report Finds Malicious Activity Continues to Grow at a Record Pace*, 2009. http://www.symantec.com/about/news/release/article.jsp?prid=20090413_01

[39] M. Stevens, A. Lenstra, and B. D. Weger, "Chosen-prefix collisions for MD5 and colliding X. 509 certificates for different identities," in *Proceedings of the 26th Annual International Conference on Advances in Cryptology - Eurocrypt' 07*, pp. 1–22, 2007.

[40] M. Stevens, A. Sotirov, J. Appelbaum, A. Lenstra, D. Molnar, D. Osvik, and B. D. Weger, "Short chosen-prefix collisions for MD5 and the creation of a rogue CA certificate," in *Proceedings of the 29th Annual International Cryptology Conference on Advances in Cryptology - Crypto' 09*, pp. 55–69, 2009.

[41] W. Sturgeon, *Serial Typo-squatters Target Security Firms*, Sep. 2005. http://news.zdnet.com/2100-1009_22-5873001.html

[42] J. Sunshine, S. Egelman, H. Almuhimedi, N. Atri, and L. F. Cranor in *Usenix Security Symposium*, 2009.

[43] X. Tang, C. N. Manikopoulos, and S. G. Ziavras, "Generalized anomaly detection model for windows-based malicious program behavior," *International Journal of Network Security*, vol. 7, no. 3, pp. 428-435, Nov. 2008.

[44] VeriSign. http://www.verisign.com/.

[45] Y. M. Wang, D. Beck, X. Jiang, R. Roussev, C. Verbowski, S. Chen, and S. King, "Automated web patrol with strider honeymonkeys," in *Network and Distributed System Security Symposium (NDSS2006)*, pp. 35–49, 2006.

[46] N. Weaver, *HTTP is Hazardous to Your Health*, 2008. http://nweaver.blogspot.com/2008/05/http-is-hazardous-to-your-health.html

[47] L. Weinstein, *Http: Must die!.* http://lauren.vortex.com/archive/000338.html

[48] T. Whalen and K. M. Inkpen, "Gathering evidence: use of visual security cues in web browsers," in *Proceedings of Graphics Interface 2005*, pp. 137–144, 2005.

[49] M. Wu, R. C. Miller, and S. L. Garfinkel, "Do security toolbars actually prevent phishing attacks?," in *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI' 06)*, pp. 601–610, 2006.

[50] M. Wu, R. C. Miller, and G. Little, "Web wallet: Preventing phishing attacks by revealing user intentions," in *Proceedings of the second symposium on Usable privacy and security (SOUPS' 06)*, pp. 102–113, 2006.

[51] Y. Wang, D. Beck, J. Wang, C. Verbowski, and B. Daniels, "Strider typo-patrol: Discovery and analysis of systematic typo-squatting," in *Proceedings of the 2nd conference on Steps to Reducing Unwanted Traffic on the Internet (SRUTI' 06)*, vol. 2, pp. 31–36, July 2006.

[52] Y. Zhang, S. Egelman, L. Cranor, and J. Hong, "Phinding phish: Evaluating anti-phishing tools," in *Network and Distributed System Security Symposium (NDSS' 07)*, 2007.

Mishari Almishari is a PhD candidate at the computer science department in University of California, Irvine. His research interests include Data Mining, Security and Privacy. He got a B.S in Computer Science from King Saud University in 2001 and an M.S. from USC in 2006.

Emiliano De Cristofaro is a researcher at PARC. His research interests include Applied Cryptography, Security and Privacy. He got a B.S. in Computer Science from University of Salerno in 2005 and a PhD from University of California, Irvine in 2011.

Karim El Defrawy is a researcher at Hughes Research Lab. His research interests include Applied Cryptography, Security and Privacy. He got a B.S. in Electrical Engineering from University of Cairo in 2003 and a PhD from University of California, Irvine in 2010.

Gene Tsudik is a professor in the Computer Science department at University of California, Irvine. His research interests include Applied Cryptography, Security and Privacy. He got a PhD in Computer Science from USC in 1991.