Harvesting SSL Certificate Data to Identify Web-Fraud

Mishari Al Mishari, Emiliano De Cristofaro, Karim El Defrawy and Gene Tsudik
Information and Computer Science, University of California Irvine
{malmishari,edecrist,keldefra,gts}@ics.uci.edu

***Abstract*—Web-fraud is one of the most unpleasant features of today's Internet. Two eminent examples of web-fraudulent activities are phishing and typosquatting. Their effects range from relatively benign (such as unwanted or unexpected 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 achieve this, we conduct the first comprehensive study of SSL certificates for legitimate and popular domains, as opposed to those used for web-fraud. Drawing from extensive measurements, we build a classifier that detects malicious domains with high accuracy. We validate our methodology with different data sets collected from the Internet. Our prototype is orthogonal to existing mitigation techniques and can be integrated with other available solutions. Our work shows that, besides its intended benefits of confidentiality and authenticity, the use of HTTPS can help mitigate web-fraud.**

I. Introduction

The Internet and its main application – the Web – have been growing continuously in recent years. Just in the last two years the Web content has doubled in size from 10 billion to over 20 billion pages, according to results from Google and Yahoo [13]. Unfortunately, such growth has been accompanied by a parallel increase of nefarious activities, as reported by [40]. The web represents a very appealing platform for various types of electronic fraud, of which phishing and typosquatting are two examples. *Phishing* is the well-known activity aimed at eliciting sensitive information – e.g., usernames, passwords and credit card details – from unsuspecting users. It typically starts with a user being directed to a fake web site with the look-and-feel of a legitimate, familiar and/or well-known web site. Consequences of phishing can range from denial-of-service to full-blown identity theft, followed by real financial losses. Only in 2007, more than 3 Billion U.S. dollars were lost to such attacks [3]. *Typosquatting* is the practice of registering domain names that are typographical errors (or minor spelling variations) of well-known web site addresses (target domains) [47]. 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 web site that the user intends to visit [47]. In addition to displaying unexpected pages, typo-domains often display malicious, offensive and unwanted content, install malware ([4], [46]) and certain typo-domains of children-oriented web sites redirect users to adult content [10]. Worse yet, typo-domains of financial web sites can serve as natural platforms for *passive* phishing attacks[1]. Recent studies have assessed the popularity of both types of malicious activities [44], [15]. For example, in Fall 2008, McAfee Alert Labs found more than $80,000$ domains typosquatting on just the top $2,000$ web sites [20]. Also, according to the Anti-Phishing Working Group, the number of reported phishing attacks between April 2006 and April 2007 exceeded $300,000$ [2].

Nevertheless, the problem remains far from solved and continues to be a race between the web-fraudsters and the security community.

Our goal is to counter web-fraud by detecting domains hosting such activities. Our approach is inspired by several recent discussions and results within the security community. Security researchers and practitioners increasingly advocate transition to HTTPS for all web transactions, similar to that from Telnet to SSH. Eminent examples of such discussions can be found in [5] and [48]. Moreover, a recent study [38] has pointed out the emergence of Phishing attacks abusing SSL certificates. The main goal is to avoid raising the suspision of users by masquerading as legitimate "secure" site.

This leads us to the following questions:

1) How prevalent is HTTPS on the Internet?
2) How *different* are SSL certificates used by web-fraudsters from those of legitimate domains?
3) Can we use the information in the SSL certificates to identify web-fraud activities such as phishing

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

and typosquatting, without compromising user privacy?

Contributions. This paper makes several contributions. First, we measure the overall prevalence of HTTPS in popular and randomly sampled Internet domains. We then consider the popularity of HTTPS in the context of web-fraud by studying its use in phishing and typosquatting activities. We then analyze, for the first time, all fields of SSL certificates and we identify useful features and patterns that represent possible symptoms of web-fraud.

Based on the measurement findings, we propose a novel technique to identify web-fraud domains that use HTTPS, based on a classifier that analyzes certificates of web domains. We validate the classifier by training and testing it over data collected from the Internet. The classifier achieves a detection accuracy over 80% and, in some cases, as high as 95%. Our classifier is orthogonal to prior mitigation techniques and can be integrated with other methods that do not rely on HTTPS to face a complete range of malicious domains. Also the integration with pre-existing solutions could improve their efficiency in the context of potentially critical applications enforcing HTTPS. Note that the classifier only relies on data in the SSL certificate and not any other private user information. Finally, our findings serve as an additional motivation for increasing the use of HTTPS, not only to guarantee confidentiality and authenticity, but also to help combat web-fraud.

Paper Organization. The rest of the paper is organized as follows. Section II introduces a brief overview of X.509 certificates. Section III presents the rationale and details of our measurements and analysis. In Section IV, we describe the details of a classifier that detects malicious domains, based on information obtained from SSL certificates. Section V discusses the implications of our findings and the limitations of our study. Section VI contains an overview of related work. Finally, we conclude our paper in Section VII.

II. 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[6]. 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, Mozilla/Firefox, Chrome and Safari – come with pre-installed trusted root certificates. Browser software makers determine which CAs are trusted third parties for the browsers' users. The root certificates can be manually removed or disabled, however, users rarely do so. The general structure of an X.509 v3 [6] certificate is shown in Figure 1. As discussed later in Section III, we analyze *all fields* for certificates of both legitimate and malicious domains.

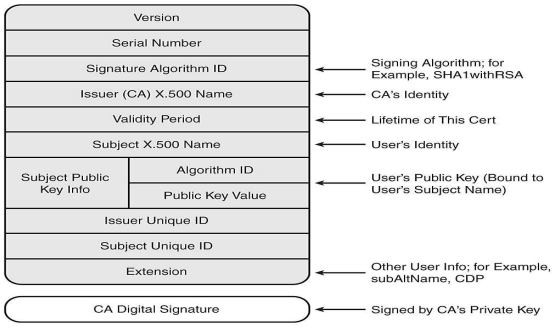

Fig. 1. **X.509 Certificate Format**

Data set	Size	HTTP Only	HTTPS Only	HTTP & HTTPS
Alexa	100 K	66%	8%	26%
.com	100 K	79%	9%	12%
.net	100 K	84%	13%	3%
phishing	2811	70%	5%	25%
typosquatting	9830	95.1%	0%	4.9%

TABLE I
Description of Data sets and Utilization of HTTP/HTTPS

III. Measurements and Analysis of SSL Certificates

We first describe our data sets and how we collect them. We then present our analysis and show how these results guide the design of a classifier that detects web-fraud domains.

A. HTTPS Usage and Certificate Harvest

Our measurement data was collected mainly during September 2009. It includes three types of domain sets: *Legitimate, Phishing and Typosquatting*. Each domain is probed twice. We probe each domain for web existence (by sending an HTTP request) and for HTTPS existence (by sending an HTTPS request). If a domain responds to HTTPS, we harvest its SSL certificate. Below, we describe the data sets and the state of HTTP/HTTPS utilization in them, also summarized in Table I.

Legitimate and Popular Domain Data Sets. The following data sets are considered for measurements and analysis of popular and legitimate domains:

- **Alexa**: $100,000$ most popular domains according to Alexa [1]
- **.com**: $100,000$ random samples of .com domain zone file, collected from VeriSign[29].
- **.net**: $100,000$ random samples of .net domain zone file, collected from VeriSign[29].

Alexa [1] is a subsidiary of Amazon.com known for its web-browser toolbar and for its web site that reports traffic ranking of Internet web sites. The company does not provide information on the number of users using its toolbar, but it claims several millions [14]. We use Alexa ranking as our measure of popularity of legitimate domains. We also randomly sample .com and .net to ensure an unbiased data-set. In the rest of the paper, we use Alexa, .com, and .net to denote three respective data sets containing the domains or certificates harvested as described above.

We find that 34% of Alexa domains use HTTPS; 26% in .com and 16% in .net. The fact that 34% of Alexa domains respond to HTTPS indicates a healthy degree of HTTPS usage among popular Internet domains. Our explanation for lower percentage in .net is that, perhaps because .net domains are historically non-commercial, most of them do not require user interaction and do not involve sensitive data, in contrast with commercial-minded .com domains.

Phishing Data Set. We collected $2,811$ domains considered to be hosting phishing scams. They were obtained from the PhishTank web site [9] as soon as they are listed. We note that reported URLs in PhishTank are verified by several users and, as a result, are malicious with high probability. We consider this data set as a baseline for phishing domains. We find that a significant percentage – 30% – of these phishing web sites employ HTTPS. This represents a 5% increase over earlier measurements conducted in October 2008. a similar trend has also been independently observed in a recent study by Symantec [38]. Furthermore, we discovered a number of phishing domains using HTTPS (~ 10%) for which we cannot obtain corresponding certificates, due to various SSL errors (e.g., illegal certificate size).

Typosquatting Data Set. To collect SSL certificates of typosquatting domains we first identified the typo domains in our .com and .net data sets by using Google's typo correction service [11] to identify possible typos in domain names. This results in $38,617$ typo domains. However, such domains might be benign domains that *accidentally* resemble typos of well-known domains. We identified the typosquatting domains in this set by detecting the parked domains among these typo domains using the machine-learning-based classifier proposed in [30] [2] . We discovered that $9,830$ out of $38,617$ are parked domains. We consider these $9,830$ names as the data set of typosquatting domains.

As shown in Table I, the result is that 486 typosquatting domains use HTTPS. They represent our typosquatting SSL certificate corpus. For convenience, when we refer to typosquatting/phishing set, we mean the set of typosquatting/phishing domains that have SSL certificates.

Overlapping of Sets. There is negligible overlapping among data sets. The size of the domain intersection between .com and Alexa is 81 and the size of domain intersection between .net and Alexa is 17. There is no overlapping between phishing and both .com and .net. The typosquatting set is a subset of the union of .com and .net sets, but it only represents a very small ratio ($\sim 1\%$ of the union of .com and .net sets). Surprisingly, there are four phishing domains that also belong to Alexa set, but the ratio of phishing domains in Alexa set is extremely small ($\sim 0.005\%$ of Alexa set). Thus, popular domains could (in very rare cases) host scam/phishing web sites.

Take Away. Based on the analysis of HTTPS usage in our data sets, we attempt to answer the first question posed in Section I and assess how prevalent is the usage of HTTPS today, We find that a significant percentage of popular and legitimate domains already use HTTPS, and so do a relevant percentage of phishing domains. As a result, we proceed by analyzing the differences between the certificates of legitimate and fraudulent web sites.

B. Certificate Analysis

The goal of this analysis is to guide the design of our detection method, the classifier. One side-benefit of the analysis is that it reveals the differences between certificates used by fraudulent and legitimate/popular domains. In total, we identified 15 distinct certificate features listed in Table II.[3] Most features map to actual

[2]Typosquatters profit from traffic that accidentally comes to their domains. One common way of doing that is to host typosquatting domains from some parking domain service. Parked domains [8] are ad-portal domains that show ads provided by a third-party service called parking service, in the form of ads-listing [47] 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. These features are omitted due to space limitations.

Feature	Name	Type	Used in Classifier	Notes
F1	md5	boolean	Yes	The Signature Algorithm of the certificate is "md5WithRSAEncryption"
F2	bogus subject	boolean	Yes	The subject section of the certificate has bogus values (e.g., –, somestate, somecity)
F3	self-signed	boolean	Yes	The certificate is self-signed
F4	expired	boolean	Yes	The certificate is expired
F5	verification failed	boolean	No	The certificate passes the verification of OpenSSL 0.9.8k 25 Mar 2009 (for Debian Linux)
F6	common certificate	boolean	Yes	The certificate of the given domain is the same as a certificate of another domain.
F7	common serial	boolean number	Yes	The serial number of the certificate is the same as the serial of another one.
F8	validity period > 3 years	boolean	Yes	The validity period is more than 3 years
F9	issuer common name	string	Yes	The common name of the issuer
F10	issuer organization	string	Yes	The organization name of the issuer
F11	issuer country	string	Yes	The country name of the issuer
F12	subject country	string	Yes	The country name of the subject
F13	exact validity duration	integer	No	The number of days between the starting date and the expiration date
F14	serial number length	integer	Yes	The number of characters in the serial number
F15	host-common name distance	real	Yes	The Jaccard distance value [24] between host name and common name in the subject section

TABLE II
FEATURES EXTRACTED FROM SSL CERTIFICATES

certificate fields, e.g., F1 and F2. Others are computed from fields in the certificate but are not directly reflected in the fields, e.g., certificate validation failure (F5) or Jaccard distance between host name and common name in the subject field (F15). Note that some features match to boolean values, whereas, others are integers, reals or strings. Finally, we believe that most interesting results correspond to: F1 (md5), F3 (self-signed), F4 (expired), F5 (verification failed), F6 (common certificate) and F15 (host-common name distance).

Analysis of Certificate Boolean Features. Features F1–F8 have boolean values, e.g., F1 (md5) is true if the signature algorithm used in the certificate is "md5WithRSAEncryption". The analysis and distributions of these features (as shown in Table III) reveal interesting and unexpected issues and differences between legitimate and malicious domains.

F1 (md5): 19% of Alexa certificates (24% and 22 % of .com and .net, respectively) use "md5WithRSAEncryption". This feature has a much higher ratio (35%) in phishing but only 26% in typosquatting. The higher ratio for phishing suggests a possible correlation between phishing domain certificates and "md5WithRSAEncryption" signature algorithm. We were surprised that such a high percentage of legitimate domains is using "md5WithRSAEncryption" as the signature algorithm despite the well-known fact that rogue certificates can be constructed using MD5 ([42], [43]) .

F2 (bogus subject): indicates whether the subject fields have some bogus values (e.g., "ST=somestate", "O=someorganization", "CN=localhost", ...). 11% of Alexa certificates (7% and 8% of .com and .net, respectively) satisfy this feature. This percentage is much higher (20%) in phishing (29% in typosquatting). This might indicate that web-fraudsters fill subject values with bogus data or leave default values when generating certificates.

F3 (self-signed): 28% of Alexa certificates are self-signed (24% and 27% of .com and .net, respectively). We did not expect such a high percentage among legitimate domains. An interesting open question is to investigate the reason(s) for all these certificates being self-signed; however, this is out of the scope of our present work. The percentages of self-signed certificates in phishing and typosquatting are 36% and 53%, respectively. This is expected, since miscreants running such domains avoid leaving a trail by obtaining a certificate from a CA, which requires some documentation and

Feature Name	Alexa (33905)	.com (21178)	.net (16106)	phish (839)	typos (486)
F1	19%	24%	22%	35%	26%
F2	11%	7%	8%	20%	29%
F3	28%	24%	27%	36%	53%
F4	21%	25%	22%	26%	43%
F5	29%	40%	37%	36%	70%
F6	33%	60%	64%	72%	95%
F7	38%	64%	68%	75%	96%
F8	17%	14%	17%	13%	19%

TABLE III
ANALYSIS OF BOOLEAN CERTIFICATE FEATURES (PERCENTAGES SATISFYING THE FEATURES).

payment. Despite high ratios in legitimate sets, there is a significant variance between the ratios in the legitimate sets and phishing (8-12%). This variance is even larger between legitimate sets and typosquatting (25-29%).

F4 (expired): another alarming result is the fraction of expired certificates, 21% of Alexa (25% and 22% of .com and .net) as well as 26% and 43% of phishing and typosquatting, respectively. Such high percentages of expired certificates are less surprising for phishing and typosquatting than for legitimate domains. There is a non-negligible difference in the ratios between phishing and Alexa (5%).

F5 (verification failed): we use OpenSSL 0.9.8k [12] to validate certificates and find that 29% of Alexa certificates (40% and 37% of .com and .net) fail verification. Also, 36% and 70% of phishing and typosquatting certificates fail verification. The percentages in .com and .net sets are higher than in phishing maybe because these are random domains and mostly unpopular thus, having a valid certificate is not essential to these domains.

F6 (common certificate) and F7 (common serial number): F6 indicates whether the certificate is also used for multiple domains in our data sets. 33% of certificates in Alexa are duplicated (60% and 64% of .com and .net, respectively). The percentage goes up to 72% for phishing and 95% for typosquatting. F7 indicates whether a certificate serial number is used in another certificate in our data sets. 38% of Alexa certificates have common serial numbers (33% were entirely duplicated). Also, 64% and 68% of .com and .net as well as 75% and 96% in phishing and typosquatting satisfy F7.

Certificates in phishing and typosquatting have higher duplication ratios than legitimate data sets (especially Alexa) suggesting that F7, perhaps combined with other features, could be helpful in identifying phishing and typosquatting domains.

F8 (validity period > 3 years): it seems intuitive that malicious domains would not acquire certificates valid for long periods of time. Thus, we analyze the percentages of certificates with validity periods exceeding 3 years. Alexa has 17% (14% and 17% of .com and .net, respectively), while phishing has 13% and typosquatting – 19%. We present further details on the exact duration of validity periods in the analysis of non-boolean features in Section III-B. As expected, the percentage of phishing certificates satisfying F8 is smaller than that for Alexa. Whereas, percentages for typosquatting and Alexa are close.

Analysis of Certificate Non-Boolean Features. F9–F11 are related to the certificate issuer: common name, organization name and country. F12 is the certificate subject's country.

F9 (issuer common name) and F10 (issuer organization): [4] some of the common-names are popular in phishing but not in Alexa. For example, "Equifax" is the most popular Common-Name for phishing (16%), whereas it is only 6% in Alexa. (The same holds for the common-name "UTN"). Also, the percentage of certificates without common names ("JustNone") is larger in Alexa (16%) than in phishing (4%). We make similar observations for organization name. For example, "Verisign" represents 10% of Alexa and 2.5% in phishing. Also, "SomeOrganization" accounts for 11% in phishing and 7% in Alexa. Such observations alone are not enough to discriminate phishing from popular/legitimate domains, but they can be combined with others for more effective discrimination. It is also interesting to see that a non-trivial percentage of phishing domains still obtain certificates from legitimate CAs.

F11 and F12 (issuer and subject countries): the USA (US) and South Africa (ZA) are responsible for around 50% and 20% (respectively) of the certificates for both legitimate/popular and malicious domains. All other countries account only for 30%. This is expected since several major CAs are located US and ZA. The granularity of countries is too coarse to derive useful information and the field can be easily filled with bogus values.

Features F13 to F15 have integer/real values corresponding to: length of certificate validity period (in days), length of serial number, and Jaccard Distance [24] between the hostname and the common name of the subject. Results of the analysis are shown using cumulative

[4] We present only the result of Alexa and phishing due to space limitations but similar conclusions were obtained for other data sets

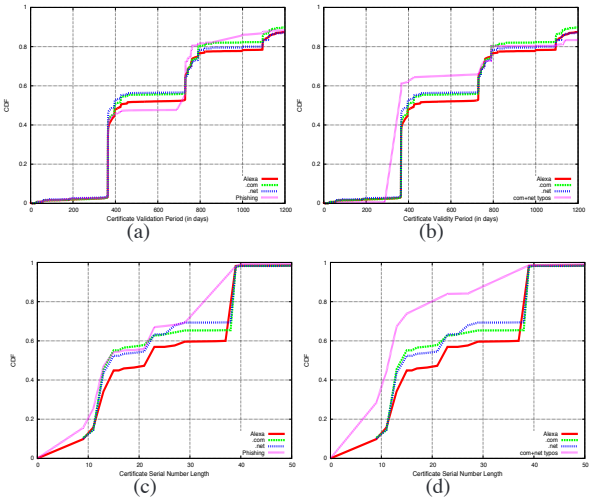

Fig. 2. **CDF of Validity Period of Alexa, .com, .net and: (a) phishing(b) typosquatting and Serial Number Length (c) phishing (d) typosquatting**

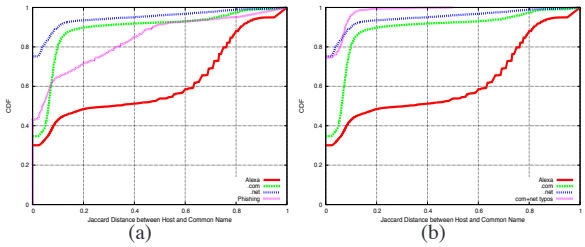

Fig. 3. **CDF of Jaccard Distance between Host and Common Name in Subject Field between Alexa, .com, .net and : (a) phishing(b) typosquatting**

distribution function (CDF) plots in Figures 2, and 3.

F13 (exact validity duration): Figure 2(a) compares the validity period CDF for phishing and legitimate sets. It shows similarities among the distributions: in Alexa, 32% are valid for 365 days and 10% for 730, as opposed to 32% for 365 days and 17% for 731 in phishing. phishing has a gradual increase in validity periods over 731, whereas Alexa has a small jump around $1,095$ (3%). The longest validity period is $4,096$ days in phishing and $2,918,805$ in Alexa. Figure 2(b) compares typosquatting with the legitimate sets. The figure clearly shows that typosquatting certificates are valid for smaller number of days than those in legitimate sets. This difference in distributions is larger than that of phishing.

F14 (serial number length): a genuine SSL certificate issued by a reputable CA is more likely to contain a long serial number, which assures uniqueness. The CDF for serial number length for all data sets in Figure 2(c) shows that phishing certificates tend to have shorter serial number length than legitimate certificates, especially, those in Alexa. Around 40% in Alexa have a serial number of length 39 (18.5% of 13, 11% of 15, 11%, 10% of 9, 10% of 23 and 6% of 11). The distribution is different for phishing, where 29% have length of 39 (22% of 13, 16% of 9, 11% of 23, 10% of 11 and 7% of 15). Figure 2(d) shows that typosquatting certificates tend to have shorter serial number than certificates in legitimate sets. Also, the difference between typosquatting and legitimate sets distributions is larger than that between phishing and legitimate sets.

F15 (host-common name distance): we expect the common name in the subject field to be very similar to the hostname in legitimate certificates. We intuitively expect that the difference between common name and hostname in malicious domain certificates is greater than that in legitimate certificates, since malicious domains may not use complying SSL certificates. This feature captures the Jaccard Distance [24] between hostname and common-name. The Jaccard Distance measures the closeness between two strings (A and B) and is defined as J(A,B) = (A and B)/(A or B). J(A,B) is basically the size of the intersection between set A and B divided by the size of their union. Thus, the larger the Jaccard distance between two strings, the more similar they are.

Figure 3(a) compares the CDF of Jaccard distance of phishing and other legitimate sets. As expected, phishing certificates tend to have smaller Jaccard distance than those in Alexa. 30% in Alexa have a Jaccard Distance of 0 (5% of 1, 4% of 0.7 and several 3-4% jumps from 0.6 until 0.9). For phishing 43% have a Jaccard Distance of 0 (4% a distance of 1 and the main jumps exist between 0 and 0.5 as opposed to 0.6 to 0.9 in Alexa). The distribution of .com (.net) shows a trend different from Alexa. We observe that .com (.net) certificates have smaller Jaccard Distance than phishing. For example, 90% (93%) of certificates in .com (.net) have Jaccard distance $\leq$ 0.2. Whereas, 71% in phishing have a Jaccard distance of $\leq$ 0.2. This might be due to the fact that domains in .com and .net sets are random domains and mostly unpopular (the intersections with Alexa's top 100k domains are negligible). Thus, having a complying SSL certificate is not essential to these domains. Figure 3(b) compares the Jaccard distance CDF of typosquatting with legitimate sets. It clearly shows that typosquatting certificates tend to have smaller Jaccard distance.

Summary of Certificate Feature Analysis. The most important observations from our analysis are:

1) Distributions of malicious sets are significantly

different from those of legitimate sets for several features.

2) Around 20% of legitimate popular domains are still using the signature algorithm "md5WithRSAEncryption" despite its clear insecurity.
3) A significant percentage ($> 30\%$) of legitimate domain certificates are expired and/or self-signed. Several user studies (e.g. [45] and [21]) show that an overwhelming percentage (up to 70-80% in some cases) of users ignore browser warning messages. This may explain the unexpected high percentages of expired and self-signed certificates that we find in the results of all our data sets. Users simply ignore those warnings and this gives no incentive for organizations to update and pay attention to their certificates.
4) Duplicate certificate percentages are very high in phishing domains, which points to the possibility that phishers use certificates only to display the familiar lock icon in the browser, hoping that users will not pay attention.
5) For some features, the difference in distributions between malicious and legitimate sets is small. However, these features turn out to be useful in building a classifier (see Section IV).
6) For most features, the difference in distributions between Alexa and malicious sets is larger than that between .com/.net and malicious sets.

IV. Certificate-based Classifier

The analysis in Section III-B showed that several features have distributions in legitimate data sets that are significantly different from those in malicious sets. Even though some show significant differences, relying on a single feature to identify malicious domains will yield a high rate of false positives. One approach, is to simply use a set of heuristics. For example, if F1-F8 are all true, the given domain is malicious with overwhelming probability. However, this might necessitate performing an exhaustive search over many combinations of heuristics, in order to find the optimal one. Another possibility is to take advantage of machine-learning algorithms and let them find the best combination of features. Machine learning algorithms are well suited for classification problems. We use several machine-learning-based classification algorithms and select the best performer. Specifically, we consider the following algorithms: Random Forest [18], Support Vector Machines [19], [26], Decision Tree [36], Decision Table [28], Nearest Neighbor [32] and Neural Networks [32]. In addition, we explore two optimization techniques for Decision Trees: Bagging [17], [37] and Boosting [23], [37]. Algorithm descriptions are out of the scope; we refer to [18], [19], [26], [36], [28], [32], [32], [17], [37], [23], [37] for relevant background information.

We use precision-recall performance metrics to summarize the 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 malicious domains (phishing or typosquatting) and "Negative" set to refer to legitimate domains. 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 [32]. In it, 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 testing it on the remaining set. We take the average of all results as the final performance result of the classifier. We use most of the features discussed in Section III-B to build two binary classifiers: (1) a Phishing Classifier and (2) a Typosquatting Classifier. The phishing/typosquatting classifier distinguishes non-phishing/non-typosquatting domains from phishing/typosquatting domains using *only* information from SSL certificates.

A. Phishing Classifier

We tried different combinations of features in the classifier and the following yielded the best results: (F1) md5, (F2) bogus subject, (F3) self-signed, (F4) expired, (F6) common certificate, (F7) common serial number, (F8) validity period (F9) issuer common name, (F10) issuer organization name, (F11) issuer country, (F12) subject country, (F14) serial number length and (F15) host-common name distance.

We create two training data sets for two different purposes. The first consists of 840 SSL certificates half of which are for phishing (positive set) and the other half – for legitimate (negative set) domains. Legitimate certificates are those of the 210 top Alexa domains, 105 random .com domains and 105 random .net domains. That is, half of legitimate certificates are for popular domains and the other half are for legitimate random domains. The purpose of training on this data set is to create a classifier that differentiates between phishing and non-phishing domains. Table IV shows performance metric values for different classifiers. The Bagging classifier shows the highest phishing detection accuracy. Despite the limited number of fields in the certificates and the

Classifier	Positive Recall	Positive Precision	Negative Recall	Negative Precision
Random Forest	0.74	0.77	0.78	0.75
SVM	0.68	0.75	0.78	0.71
Decision Tree	0.70	0.79	0.81	0.73
Bagging - Decision Tree	0.73	0.80	0.81	0.75
Boosting - Decision Tree	0.74	0.69	0.67	0.72
Decision Table	0.72	0.78	0.8	0.74
Nearest Neighbor	0.74	0.73	0.73	0.74
Neural Networks	0.7	0.77	0.8	0.73

TABLE IV
PERFORMANCE OF CLASSIFIERS - DATA SET CONSISTS OF: (A)420 PHISHING CERTIFICATES AND (B)420 NON-PHISHING CERTIFICATES (ALEXA, .COM AND .NET).

Classifier	Positive Recall	Positive Precision	Negative Recall	Negative Precision
Random Forest	0.90	0.89	0.89	0.9
SVM	0.91	0.87	0.86	0.90
Decision Tree	0.86	0.89	0.90	0.87
Bagging - Decision Tree	0.90	0.88	0.87	0.9
Boosting - Decision Tree	0.89	0.81	0.80	0.88
Decision Table	0.90	0.84	0.82	0.90
Nearest Neighbor	0.87	0.86	0.86	0.87
Neural Networks	0.87	0.89	0.90	0.87

TABLE V
PERFORMANCE OF CLASSIFIERS - DATA SET CONSISTS OF: (A)420 PHISHING DOMAIN CERTIFICATES AND (B)420 POPULAR DOMAIN CERTIFICATES (ALEXA)

dependencies among them, we can build a classifier with 80% phishing detection accuracy.

The second training data set consists of 840 SSL certificates. Half of these are for phishing domains (positive set) and the rest – legitimate certificates for 420 top Alexa's domains (negative set). The purpose is to train the classifier to differentiate phishing from popular domains. Performance results of different classifiers are shown in Table V. The results improve significantly from those in the first training data set. Random Forest (along with few others) exhibits the best phishing detection accuracy with a rate of 89%. The reason is that (as shown in Section III-B) the difference in distributions of many features between Alexa and phishing is much larger than that between .com (.net) and phishing. Thus, the random .com and .net certificates are polluting the non-phishing set and making it harder to differentiate. Although the number of certificate fields is small, we can reliably differentiate phishing from popular domains with high accuracy.

Similar to machine-learning-based spam filtering solutions, the larger the training data set, the better the performance is. We believe that our solution may incur false positives when actually deployed. However, the number of false positives can be reduced by training on larger data sets and newer samples.

We note that the use of the classifier alone cannot successfully detect *all* phishing web sites, since it only works with HTTPS. However, our approach can be combined with other phishing mitigation techniques (e.g., [50], [27], [22]) to enhance the overall accuracy. Our results show how clearly limited information pulled from SSL certificates (which has been overlooked before) can help in distinguishing phishing and legitimate domains in the context of *claimed-to-be* secure navigation.

B. Typosquatting Classifier

We use the same set of features as in the phishing classifier and also create two training data sets: the first with certificates of 486 domains from the typosquatting (positive) data set (see Section III-A) and 486 non-typosquatting (negative) domains. Half of the negatives are picked randomly from Alexa and the rest are random .com and .net domains (distributed equally between the two). This training set trains the classifier to differentiate between typosquatting and non-typosquatting domains. Performance of various classifiers is shown in Table VI. The metrics are fairly similar across classifiers, however, Decision Tree offers the highest accuracy with a rate of 93%.

The second data set consists of the same typosquatting certificates in the previous data set and certificates of 486 Alexa top domains. This set is needed to train the classifier to differentiate, typosquatting and popular domains. Performance results are shown in Table VII. The performance of many classifiers improves because the difference in distributions of features between Alexa and typosquatting data sets is larger than the difference between .com (.net) and typosquatting data sets. The Neural Network classifier shows the highest typosquatting detection accuracy, with a rate of 95%.

V. DISCUSSION

Based on the measurements presented in the previous sections, we find that a significant percentage of well-known domains *already* use HTTPS (alongside HTTP), thus it is possible to harvest their certificates for classifying them, without requiring any modifications from the domains' side. Furthermore, the non-trivial portion of phishing web sites using HTTPS highlights the need to analyze and correlate the information provided in their server certificates.

Classifier	Positive Recall	Positive Precision	Negative Recall	Negative Precision
Random Forest	0.86	0.88	0.88	0.87
SVM	0.86	0.88	0.89	0.86
Decision Tree	0.84	0.93	0.93	0.85
Bagging - Decision Tree	0.87	0.90	0.90	0.87
Boosting - Decision Tree	0.89	0.85	0.84	0.88
Decision Table	0.86	0.87	0.87	0.86
Nearest Neighbor	0.86	0.84	0.84	0.86
Neural Networks	0.84	0.89	0.90	0.85

TABLE VI
PERFORMANCE OF CLASSIFIERS - DATA SET CONSISTS OF: (A)486 TYPOSQUATTING DOMAIN CERTIFICATES AND (B)486 NON-TYPOSQUATTING DOMAIN CERTIFICATES (TOP ALEXA'S DOMAIN CERTIFICATES, .COM AND .NET RANDOM CERTIFICATES)

Classifier	Positive Recall	Positive Precision	Negative Recall	Negative Precision
Random Forest	0.95	0.93	0.93	0.95
SVM	0.94	0.92	0.92	0.94
Decision Tree	0.95	0.94	0.93	0.95
Bagging - Decision Tree	0.96	0.94	0.93	0.96
Boosting - Decision Tree	0.98	0.90	0.90	0.97
Decision Table	0.96	0.92	0.92	0.96
Nearest Neighbor	0.93	0.94	0.94	0.92
Neural Networks	0.95	0.95	0.95	0.95

TABLE VII
PERFORMANCE OF CLASSIFIERS - DATA SET CONSISTS OF: (A)486 TYPOSQUATTING DOMAIN CERTIFICATES AND (B)486 POPULAR DOMAIN CERTIFICATES (TOP ALEXA'S DOMAIN CERTIFICATES)

Limitations. We acknowledge that our data sets should be extended through longer observation periods. Furthermore, additional data sets of legitimate domains need to be taken into consideration, e.g. popular web sites 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(typosquatting) classifier may incur false positives when actually deployed. However, this is a common problem to many machine-learning-based mitigation solutions (spam filtering and intrusion detections based on machine-learning algorithms) and the number of false positives can be minimized by training the classifier on larger and more comprehensive data sets (and continuing to do so for new samples). Our classifier does not provide a complete standalone solution to the phishing(typosquatting) threat since many domains do not have HTTPS. Instead, integrated with pre-existing solutions (e.g., [50], [27], [22]), it improves their effectiveness in the context of potentially critical applications enforcing HTTPS and it contributes to face a complete range of malicious domains.

Finally, we note that our classifier does not identify all kinds of typosquatting domains. For instance, typo-domains displaying unexpected pages and ads might have no incentive to use HTTPS, since they do not attempt to elicit private information from users. However, in the case of a further increase of HTTPS usage, our technique will be as a consequence increasing its effectiveness. Generic typosquatting detection remains at the moment an interesting open issue that has not found effective solutions so far.

Using information in certificates. Our results show significant differences between certificates of legitimate 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 web site's degree of trustworthiness, thus improving prior metrics, such as [50], [27], [22]. 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 [22], [35]. 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 is not privacy compromising as keeping fine-grained navigation history.

How malicious domains will adapt. Web fraud detection is an arms race and 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 and start analyzing SSL certificate fields in more detail, we expect there will be two scenarios for web-fraudsters to adapt: (1) to get legitimate certificates from CAs and leave a paper trail pointing to "some person or organization" which is connected to such an activity, (2) to craft certificates that have similar values like those that are the most common in those 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) but for some 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.

VI. Related Work

The authors in [25] conducted a measurement 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, the authors showed encouraging measurement results such as the use of AES as default option for most of the servers that did not support AES. Also, a trend toward using stronger cryptographic function has been observed over two years of probing, despite a slow improvement. In [39], 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 [33] 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 [34] 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 [38] 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 [3] that it causes. One notable study is [15] 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 web sites using image shingling. In [22], a machine learning based methodology was proposed for detecting phishing email. 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 [27]. It kept track of sensitive information and warned the user whenever the user tried to enter sensitive information into untrusted web sites. 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 [35]. In [31], the authors tackled the problem of detecting malicious web sites 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, ed al. [47] showed that many typosquatting domains were active and parked with a few parking services, which served ads on them. Similarly, [16] 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 web sites were considered phony. [30] 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 [41]. 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 [7] some capabilities for identifying typosquatters.

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., [45] and [21])— might explain the unexpected high percentage of expired and self signed certificates that we found in the results of all our data sets. In [49], the authors proposed a solution to simplify

the process (for the users) of authenticating the servers' public keys (or SSL certificates) by deploying a set of semi-trusted collection of network servers that continuously probed the servers and collected their public keys (or SSL certificates). When the user was exposed to a new public key (SSL certificate), he referred to these semi-trusted servers to verify the authenticity of public keys (SSL certificates).

We conclude that, 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.

VII. Conclusion

In this paper, we study the prevalence of HTTPS usage 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 using HTTPS based solely on their SSL certificates, thus also preserving user privacy. Our work can be integrated with pre-exisiting detection techniques to improve their effectiveness in the context of potentially critical applications enforcing HTTPS. Finally, we believe that our results may serve as a motivating factor to increase the use of HTTPS on the Web, showing that besides its intended benefits of confidentiality and authenticity, the use of HTTPS can help identifying web-fraud domains.

References

[1] Alexa, The Web Information Company. http://www.alexa.com.

[2] Anti Phishing Working Group. Phishing Activity Trends. Report for the Month of April. 2007.

[3] Gartner Survey. http://www.gartner.com/it/page.jsp?id=565125.

[4] Googkle.com installed malware by exploiting browser vulnerabilities. http://www.f-secure.com/v-descs/googkle.shtml.

[5] HTTP is Hazardous to Your Health. http://nweaver.blogspot.com/2008/05/http-is-hazardous-to-your-health.html.

[6] Internet X.509 Public Key Infrastructure Certificate and CRL Profile (IETF RFC5280). http://www.ietf.org/rfc/rfc5280.txt.

[7] McAfee SiteAdvisor. http://www.siteadvisor.com/.

[8] Parked domain site. http://adwords.google.com/support/aw/bin/answer.py?hl=en&answer=50002.

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

[10] Screenshots of questionable advertisements. http://research.microsoft.com/Typo-Patrol/screenshots.htm.

[11] The Google Spell Checker. http://www.google.co.uk/help/features.html.

[12] The OpenSSL Project. http://www.openssl.org/.

[13] World Wide Web Size. http://www.worldwidewebsize.com/.

[14] Alexa. Alexa Ranking Methodology. http://www.alexa.com/help/traffic_learn_more.

[15] D. S. Anderson, C. Fleizach, S. Savage, and G. M. Voelker. Spamscatter: characterizing internet scam hosting infrastructure. In *16th USENIX Security Symposium*, pages 1–14, 2007.

[16] A. Banerjee, D. Barman, M. Faloutsos, and L. N. Bhuyan. Cyber-Fraud is One Typo Away. In *Infocom 2008 mini-conference*, 2008.

[17] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123–140, 1996.

[18] L. Breiman. Random Forests. *Machine Learning*, 45(1):5–32, 2001.

[19] H. Drucker, V. Vapnik, and D. Wu. Support vector machines for spam categorization. *IEEE Transactions on Neural Networks*, 10(5):1048–1054, 1999.

[20] B. Edelman. Unintended Adventures In Browsing. www.mcafee.com/us/local_content/misc/threat_center/msj_unintended_adventures_brow Fall 2008.

[21] S. Egelman, L. F. Cranor, and J. Hong. You've been warned: an empirical study of the effectiveness of web browser phishing warnings. In *CHI '08: Proceeding of the twenty-sixth annual SIGCHI conference on Human factors in computing systems*, pages 1065–1074, New York, NY, USA, 2008. ACM.

[22] I. Fette, N. Sadeh, and A. Tomasic. Learning to Detect Phishing Emails. In *WWW 2007*.

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

[24] J. Gower and P. Legendre. Metric and euclidean properties of dissimilarity coefficients. *Journal of Classification*, 1986.

[25] H. Lee, T. Malkin and E. Nahum. Cryptographic Strength of SSL/TLS Servers: Current and Recent Practices. In *IMC*, 2007.

[26] T. Joachims. A statistical learning model of text classification with support vector machines. In *Proceedings of the 24th ACM International Conference on Research and Development in Information Retrieval*, 2001.

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

[28] R. Kohavi. The power of decision tables. In *Proceedings of the European Conference on Machine Learning*, 1995.

[29] T. Lord. Resolving Everything: VeriSign Adds Wildcards. http://slashdot.org/articles/03/09/16/0034210.shtml?tid=95.

[30] M. Almishari and X. Yang. Text-based ads-portal domains: Identification and measurements. *Under Revision*.

[31] J. Ma, L. K. Saul, S. Savage, and G. M. Voelker. Beyond blacklists: learning to detect malicious web sites from suspicious urls. In *KDD '09: Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 1245–1254, New York, NY, USA, 2009. ACM.

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

[33] E. Murray. SSL server security survey. http://web.archive.org/web/20031005013455/http://www.lne.com/ericm/papers/ssl_serve

[34] Netcraft. Netcraft SSL survey. http://news.netcraft.com/SSL-Survey, 2008.

[35] B. Parno, C. Kuo, and A. Perrig. Phoolproof Phishing Prevention. In *Financial Cryptography and Data Security 2006*.

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

[37] J. R. Quinlan. Bagging, boosting, and c4.5. In *13th National Conference on Artificial Intelligence and 8th Innovative Applications of Artificial Intelligence Conference*, 1996.

[38] Z. Rezaei. Phishing Toolkit Attacks are Abusing SSL Certificates. http://www.symantec.com/connect/blogs/phishing-toolkit-attacks-are-abusing-ssl-certifi 2009.

[39] S. Yilek, E. Rescorla, H. Shacham, B. Enrigh and S. Savage. When private keys are public: Results from the 2008 debian openssl vulnerability. In *IMC*, 2009.

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

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

[42] M. Stevens, A. Lenstra, and B. Weger. Chosen-prefix collisions for md5 and colliding x.509 certificates for different identities. In *EUROCRYPT '07: Proceedings of the 26th annual international conference on Advances in Cryptology*, pages 1–22, Berlin, Heidelberg, 2007. Springer-Verlag.

[43] M. Stevens, A. Sotirov, J. Appelbaum, A. Lenstra, D. Molnar, D. A. Osvik, and B. de Weger. Short chosen-prefix collisions for md5 and the creation of a rogue ca certificate, 2009. http://eprint.iacr.org/.

[44] W. Sturgeon. Serial typo-squatters target security firms. http://news.zdnet.com/2100-1009_22-5873001.html, September 2005.

[45] J. Sunshine, S. Egelman, H. Almuhimedi, N. Atri, and L. F. Cranor. Crying wolf: An empirical study of SSL warning effectiveness. In *Proceedings of the 18th Usenix Security Symposium*, August 2009.

[46] Y. Wang, D. Beck, X. Jiang, R. Roussev, C. Verbowski, S. Chen, and S. King. Automated Web Patrol with Strider HoneyMonkeys. In *NDSS'06*, pages 35–49, 2006.

[47] Y.-M. Wang, D. Beck, J. Wang, C. Verbowski, and B. Daniels. Strider typo-patrol: Discovery and analysis of systematic typo-squatting. In *SRUTI'06*, pages 31–36, July 2006.

[48] L. Weinstein. http: Must Die! http://lauren.vortex.com/archive/000338.html.

[49] D. Wendlandt, D. G. Andersen, and A. Perrig. Perspectives: Improving ssh-style host authentication with multi-path probing. In *Proceedings of the USENIX Annual Technical Conference (Usenix ATC)*, June 2008.

[50] M. Wu, R. Miller, and G. Little. Web wallet: preventing phishing attacks by revealing user intentions. In *SOUPS'06*, pages 102–113, 2006.