# Phishing in Smooth Waters: The State of Banking Certificates in the US

Zheng Dong[1], Kevin Kane[2], and L. Jean Camp[1]

[1] Indiana University, Bloomington, IN, 47405, USA
[2] Microsoft Research, Redmond, WA, 98052, USA

**Abstract.** A critical component of the solution to online masquerade attacks, in which criminals create false web pages to obtain financial information, is the hierarchy of public key certificates. Masquerade attacks include phishing, pharming, and man-in-the-middle attacks. Public key certificates ideally authenticate the website to the person, before the person authenticates to the website. Public key certificates are typically issued by certificate authorities (CAs).

Banks are the most common target of phishing attacks, so we implemented an empirical study of certificates for depository institutions insured by the Federal Depository Insurance Corporation (FDIC) and compared them to general purpose, non-banking certificates. Our study of websites of FDIC-insured banks found that the current configuration fails to support website authentication. The most common failure is an absence of certificates, meaning that a false certificate would be the only valid-named certificate for that institution. Certificates with incorrect names, incorrectly structured certificates, and shared certificates all plague online banking. The vast majority of banks, especially smaller banks, apparently lack the expertise, support, or incentive to implement certificates correctly.

We document the current state of bank certificates. We compare these with general-purpose certificates (e.g., the top one million websites). We survey the various proposals for the certificate market writ large, including pinning and notaries. We identify how those fit and fail to fit the unique problem of banking certificates. We close with policy and technical recommendations to alter the use of certificates so that these can be a valid basis for consumer trust.

**Keywords:** security, policy, consumer protection.

## 1 Introduction

Banks have become targets of increasing phishing attacks, causing severe security problems. Bank websites, like every other entity, should be identified by the certificate. The validation of a legitimate connection between a public/private key set and a domain name is a core purpose of a certificate. The other core purpose of a certificate is to enable encryption during a session to prevent eavesdropping.

## 2 Current PKI Challenges

What is a certificate? A certificate is a set of facts cryptographically signed by third-party organization in order to verify those facts to individuals who are connecting to remote networked entities. Thus, there are three ways that certificates could fail: the cryptography, including the digital signature or hash value, could be flawed; the set of facts embedded in the signature is somehow incorrect; or the individual could perceive that the certificate means something quite different than the intended issuance and implications.

The set of facts verified by the signature may have never been true, may change and no longer be true, or may be technically true but profoundly misinformative. The importance of the fields and the reason why the verified facts are unreliable vary. The certificate has a standard structure following the X.509 format. The following are the primary fields of an X.509 certificate.

1. Certificate Version. This field indicates the format of other certificate fields.
2. Serial Number. This is a unique certificate identifier assigned by the CA.
3. Signature Algorithm. This is the algorithm that is used to generate and verify the digital signature.
4. Message Authentication Algorithm. This algorithm is used to generate the message digest. The message digest is the compressed form of the entire certificate information, and is what is technically verified in the certificate signature.
5. Issuer. This field contains the following information about the CA which issues the certificate: common name, organization, physical address (city, state, country) and email are typically included.
6. Validity. The start and end dates of the certificate's validity period.
7. Subject. Similar to Issuer, the field contains the following information about the entity that the certificate is issued to: common name, organization, physical address (city, state, country) and email are typically included.
8. Certificate Extensions. Depending on the certificate version, a number of optional but important certificate fields may exist. For example, "Basic Constraints" can indicate whether a certificate can be used as an intermediate certificate. "Extended Key Usage" restricts the use of the public key to a list of purposes.

The primary failure mode of certificates is treated in section 2.1, human factors. Organization failures and economically-driven failures are the focus of section 2.2. Failures resulting from flawed cryptography are addressed in section 2.3.

### 2.1 Human Factors and Certificates

People do not look for certificates and do not understand the risks associated with various failure modes. In no small part this is because the categorization created by normal certificate practice is not meaningful in terms of informed risk

and trust decisions. Knowing that there is an entity which agrees that a domain name is owned by another entity does not particularly assist in decision-making of end users, especially in financial decision-making.

How would an end user (often with little time and technical training) make use of the certificate fields? Values of specific certificate fields are often invisible to end users by default. Unless an individual sees a warning they are unlikely to look at a certificate. The warnings are often obtuse or are written with an assumption of a highly educated reader. Examining warnings, we found that a standard certificate warning had an Automated Readability Index of 11.2. The Average Grade Level in text for certificate warnings is 12.4. Thus, reading a certificate warning appears to require a high school degree and in fact may require some college. Consider that the median American adult reads at a 7th or 8th grade level. Most people literally are unable to parse a certificate. Further, they cannot understand the warnings [1].[3] Technical language, confusing terms, and the sheer length of the messages have proven problematic [2]. The cost of reading privacy policies has been found to be excessive [3]. Consider how much easier it is to read a privacy policy than to understand a certificate.

That these certificates are not readable has been documented in previous research. Early human-centered research on phishing indicated that roughly a quarter of the participants ignored security indicators. Participants reported that either they did not notice the indicators (specifically the lock icon) or that the meaning of the warning was unclear [4]. Not only understanding, but even finding relevant information in certificates has proven to be difficult for those without technical expertise [2].

Optimistically, fifty participants at Carnegie Mellon were found to heed security warnings during a phishing experiment (79%), and over half claimed to have read the entire warning [5]. However, even over the time period of the laboratory study, the vast majority (all but one) were effectively phished despite at least passive warnings. When this study was reproduced three years later, the findings could not be validated with a more general population [6], where changes in color of warning, wording of warning, or browser used (i.e., Firefox or IE) were not significant. When the [5] study was replicated, self-reported behavior did not predict actual behavior. When being interviewed, one-third of the participants indicated that they ignored the warning because they were in a secure study environment. Others predicted that they would take action based on the warning but instead rapidly closed the warning box to continue to the (fake, experimental) phishing site [6].

Advocating for identification of specific categories of sites is not new. The W3C Standard Web Security Context: User Interface Guidelines recommends "prior designation of high-value sites" [7]. A subset of the standards' authors proposed additional research in web users' understanding of security, human

[3] The explanation of certificates in Google Chrome had a Flesch-Kincaid Grade Level of 11.6; a Gunning-Fog Score of 15.1; a Coleman-Liau Index 13.1and a SMOG Index of 11.

perception of security cues, how these cues effect users over time, and the testing of UI prototypes [8].

Nor is identification of the trust challenges in PKI radical. Quite early problems with the underlying trust architecture of certificates were identified. The basic facts are that certificates do not align with human trust behaviors as identified in any domain [9], and the models of human and organization trust in PKI design were considered to be subsumed in the (arguably more solvable) technical challenge of enabling cryptographic attestations of certain identity claims [10]. However, the adoption and widespread use would counter the concerns that these certificate in fact are "signifying nothing" [11].

### 2.2 Organizational Challenges in PKI

Due to the large number of certificate authorities, each of which has its own operational processes, and the burden of legacy requirements, issuance practices vary widely and change slowly. There are technical and operational practices in the current certificate authority architecture.

As an important defense technique, HTTPS has been enabled by websites to provide mutual authentication. Ideally, a certificate identifies the entity to which a person is connecting as well as enabling key exchange (in order to prevent eavesdropping). While a secure channel can be established between an end user and the website, there is often no guarantee on the website's real identity. Beyond human factors, there are several technical reasons why public key certificates have not become a reliable indicator of the identity of websites.

Any CA can issue a certificate to any website. Since the actual operations of CAs can vary significantly, it may be possible for an adversary to obtain a valid certificate from a less-diligent CA. Regardless of the businesses and technical practices of a CA, all CAs receive identical trusts from the web browsers.

Secondly, unless Extended Validation (EV) was conducted in the certificate issuance, the CAs typically do not perform strict verifications on the actual identities of websites. There is not an industry standard practice.

Thirdly, EV certificates are not widely used. At this point, there is no research indicating that the visual cues used to distinguish EV certificates in browsers actually enable the general population to distinguish EV and non-EV certificates. Due to the significantly higher cost and difficulty of obtaining an EV certificate, the majority of websites are still using non-EV certificates.

In this work, we systematically analyze the current practice of public key certificates used on FDIC-insured bank websites. Based on our certificates downloaded from the 1 million most popular websites and the FDIC official member list, we report our direct observations as well as the machine-learning classification results. We then propose possible approaches to certificate issuance to the FDIC-insured banks, to enhance the overall level of security of online banking transactions.

### 2.3 Technical Challenges with the Larger PKI

There are sources of strictly technical failures as well as organizational issues and human factors challenges.

The first problematic processes concern the use of weak cryptography. As attacks against RSA have become increasingly sophisticated, the consensus amongst the cryptography community is now that 1024-bit RSA keys offer insufficient security for the typical validity periods of end-entity X.509 certificates, recommending at least a 2048-bit key length for these certificates. But due to legacy platforms whose software is unable to use keys longer than 1024 bits, and resource-constrained platforms which expend more processing time and battery power to do public key operations on longer keys, some CAs continue to allow issuance of certificates with 1024-bit RSA keys for validity periods of at least one year.

The hash algorithm MD5 was standard for certificate signatures before SHA-1, and continued to be used despite increasingly severe attacks on it. The Flame malware attack in 2012 took advantage of a collision in MD5 to create a fraudulent certificate [12]. Fortunately the advent of SHA-1 and the increasingly severe weaknesses in MD5 have helped largely eliminate its use in new issuance, but older certificates are still in use.

The second problematic processes concern the construction of the certificate fields themselves. The use of weak cryptography is complicated by the use of long validity periods, sometimes three, five, seven years or more for end-entity certificates. As the validity period is, amongst other purposes, used to limit the possible exposure of a cryptography break by rendering a certificate useless by the time an attacker could brute-force the key, when the validity period exceeds this time because of advances in cryptanalysis, these certificates become vulnerable but must continue to be accepted.

Revocation is a third challenge. Some of these certificates can have their validity periods shortened through the use of revocation, but revocation is an even larger source of problems. Two standards for revocation, Certificate Revocation Lists (CRLs) and the Online Certificate Status Protocol (OCSP), are in common use. But practices amongst CAs vary, with some issuing certificates with CRL information, some with OCSP, some with both, and some with neither. Even if the CA implements best practices in its certificate issuance, this problem is further complicated by the irregular behavior of browsers and web application clients in checking revocation status. Mozilla Firefox recently made the decision to exclusively use OCSP, meaning that all certificates with only CRL information in the certificate become effectively irrevocable to Firefox clients. Because the use of CRL requires a substantial data download compared to the smaller traffic required for OCSP, clients on constrained data connections, such as cellular connections, may also only use OCSP, if they do any revocation checking at all. Apps and other non-browser web clients that use SSL frequently do no revocation checking at all, making it completely impossible to revoke certificates of servers to which they connect.

CAs have in the past issued certificates with poorly chosen Extended Key Usages (EKUs). The EKU is what restricts a certificate to only be used for particular purposes, such as authenticating an SSL server, authenticating a client, signing code, and providing a trusted timestamp. The Flame malware attack also took advantage of an intermediate Certificate Authority that had an unused but valid Code Signing EKU, allowing rogue certificates issued from it to be used to sign code.

## 3 The Reality of Certificates in the Wild

As part of our larger research project to identify potential masquerade attacks in the PKI ecosystem, we have been collecting public key certificates daily from the following sources.

1. The top 1 million websites visited in the previous day. Our script obtains the website list from Alexa every morning and tries to connect to each website on the list via HTTPS. A certificate is downloaded from the website if it is different from our previous observation. In addition to running the script on servers in the United States, our certificate collection has also been deployed on Asian and European servers through the *PlanetLab* research platform.
2. FDIC-insured bank official websites. The FDIC maintains an official list of its member institutions. For each member, our script retrieves the name, the physical address and the official web domain of the bank (if any). The script then removes invalid URLs (e.g. email addresses) from the list, and tries to download a certificate from each valid website on the official list.
3. For each FDIC website, we downloaded the homepage of the website and searched for HTTPS links. Additional certificates were downloaded by following these hyperlinks. We filtered out popular and common links (e.g., "Like Us on Facebook") from the observations.

Certificates from the two data sources enable us to conduct a thorough analysis on the current status of the banking certificates in the United States. Until Aug 20th 2014, we had observed 1.1 million distinct certificates from 3.8 million popular general websites. Note that our exploration results in a far broader view of the PKI than the average user would experience. One study of browser histories illustrated that for a specific individual, some 90% of all roots would not be encountered at all [13]. Of course, the challenge in leveraging this phenomenon is that the useful 10% has critical individual variances.

## 4 Investigation of Banking Certificates

We have investigated two approaches for analyzing FDIC-insured bank certificates: direct observation and machine-learning classification. In statistical analysis, we can observe distinct patterns from the categories of certificates. In addition, we have several direct observations of problems with banking certificates.

### 4.1 Direct Observations

Even without statistical analysis, several issues are apparent in the compilation of the certificates of depository institution. First of all is the absence of HTTPS. Among all the 27,000 records in the official FDIC list, only 6,000 had valid domains. We tried to connect to every web domain on the list, but were only able to establish HTTPS connections with 3,000 of them.

For twenty thousand banks there were no certificates. The lack of association of domain name and certificate is problematic for two reasons. First, this means that it would be feasible to obtain a certificate for a banking domain name and have that be the sole certificate. As banks are closed, merge or simply change branding it would be quite feasible for a criminal organization to obtain a domain name that is an expired bank domain, and then obtain a certificate. As no certificate would have ever been issued previously for that domain, **none of the proposed changes to the certificate architecture would address this**. We return to this issue in Section 7.

Our second observation was the mismatch of the web domain and subject entries in the certificate. As indicated in the subjects *common name* field and the *subject alternative name* extension, a certificate can only be used in these specific web domains. Among all downloaded certificates, we discovered 498 domain name mismatches.

The third observation was certificate sharing. In addition to certificate mismatches, it is possible that several web domains share the same mismatched certificate. According to our dataset, many of the shared certificates were provided as part of the default server configuration, which have not been changed by their website administrators. As one extreme example, one certificate of *sinkdns.org* has been used by 51 different HTTPS bank domains. A certificate of *webaccess1.com* was used by 43 different banks. Certificates of the virtualization company *Parallels* were shared among 37 websites.

The issue of cloud computing also makes the lack of consistent, identifiable financial certificates problematic. While botnets provide a platform where there is no constraint on criminal activity, the legitimacy and terms of services of cloud providers does not protect them from criminal use. Clouds are misused by attackers, including attackers who engage in masquerade attacks such as phishing. In our research twenty two sites identified by PhishTank as phishing sites were hosted Google Drive. In addition, there are reports of criminal use of Microsoft's Azure [14].

Preventing masquerade fraud against financial institutions requires making these distinctions. The simple model of good versus untrusted in PKI is failing to provide adequate information.

### 4.2 Machine Learning

As reported in our poster [15] at ACSAC 2013, we are able to use machine-learning techniques to classify the public key certificates into a number of predefined groups.

We have examined the classification performance with three different machine-learning algorithms: J48, NBTree and Random Forest. J48 is a Java implementation of a traditional decision-tree algorithm, C4.5. This algorithm builds the decision tree based on information gains of each member in the feature set. NBTree is a combination of the Naive Bayes regression and a decision tree. Random Forest is an *ensemble* algorithm that builds a number of decision trees and makes the final decision based on a majority vote of all decision trees. For each tree in the forest, only a subset of randomly selected features are utilized.

Our machine-learning models tried to classify the unlabeled certificates into two categories: the FDIC-insured banks and the general websites. The classification performance is listed in Table 1. For each algorithm, we report the overall percentages of certificates that have been correctly and incorrectly identified. For each category, we record the true positive and false positive rates, indicating percentages of the correct and incorrect instances for the particular category. According to the table, all three algorithms have achieved 99.8% for correctly classified instances. The true positive rates for the bank category have achieved 96% for all three algorithms and the false positive rates (i.e. the certificates categorized as banks but are actually in the general category) are as low as 0.1%. For the general category, the true positive rates are as high as 99.9%, while only 3.7% of the bank certificates are classified as general, probably indicating the misconfiguration of the bank certificates that were observed in the dataset.

**Table 1.** Classification Performance Summary

| | | J48 | NBTree | Random Forest |
|---|---|---|---|---|
| Correct % | | 99.84 | 99.81 | 99.83 |
| Incorrect % | | 0.16 | 0.19 | 0.17 |
| Bank | True Positive Rate | 96.30 | 96.80 | 96.60 |
| | False Positive Rate | 0.10 | 0.10 | 0.10 |
| General | True Positive Rate | 99.90 | 99.90 | 99.90 |
| | False Positive Rate | 3.70 | 3.20 | 3.40 |

Figure 1 is a detailed demonstration of the classification. The waterfall chart uses horizontal lines to indicate the classification results of each algorithm. A vertical line indicates the same predicted category between two algorithms and an oblique line shows a disagreement between two algorithms. As shown in the chart, the majority of the certificates were correctly classified. The correctness of classification can improve by combining the results of all three machine-learning algorithms.

## 5 Proposed PKI Solutions

There are four basic solutions that are currently being proposed to the challenges of the of bogus certificates: revocation, whitelists, whitelists using history, and limiting competition.

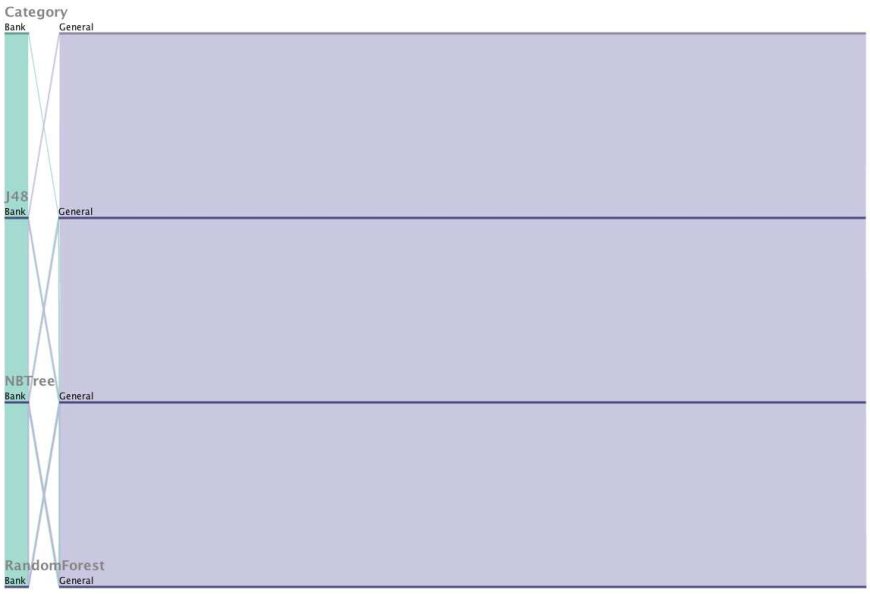


**Fig. 1.** Certificate Classification Results

Several security mechanisms have been proposed to detect fraudulent public key certificates using *revocation*. Certificate Revocation List (CRL) and/or Online Certificate Status Protocol (OCSP) are often provided by the certificate authority to indicate the current validity status of their issued certificates. In addition to building a trusted chain, popular web browsers can usually verify the latest revocation information from these two services. Results from CRL and OCSP are official, but there is usually a lag between a rogue certificate first appears and it becomes blacklisted. In the extreme case when a CA is compromised, the CRL and OCSP may become untrusted altogether.

Another proposal is adding a *whitelist* for valid cryptographic certificates. In this way, the authentication of the remote site is a function of two third-parties: the CA which issued the certificate and the whitelist provider who vouches for the certificate. For example, the Electronic Frontier Foundation (EFF) has constructed a large certificate observatory by actively scanning the IPv4 address space. End users can benefit from the observatory by installing a browser extension and submit their observed certificates through the extension. Warnings will be generated when the submitted certificates are inconsistent with the observatory.

A disadvantage of using whitelists and revocation is that these are inherently centralized. Of the proposals to use *history* in certificate validations two are distributed. These historical approaches are referred to in the literature as *certificate notaries*. *Perspectives* [16] form Carnegie Mellon University, and *Convergence* [17], a Firefox extension, both rely on a comparison between the user-

submitted certificate hashes and observations made by geographically distributed notary servers. Since the end users need to submit their secure browsing history to the services, the observatory or notary approaches unavoidably involves loss of privacy, although all of the service providers claim to collect no personal identifiable information.

Certificate Pinning associates each website with a small whitelist where the whitelist is stored by the local browser. This list is updated upon first visit, as originally proposed in [18], and also from the central Google repository of certificates. This has been implemented in Google Chrome and protected several Google-owned domains from being involved in the use of rogue certificates. One of the weaknesses of this approach is scalability. It can be extremely difficult to coordinate all online websites and browser manufacturers to maintain a huge certificate pinning table. Again another weakness is privacy, as browsing history is submitted to Google.

Limiting competition is the proposal embedded in DANE: DNS-based Authentication of Named Entities [19]. This proposal would require that certificates were associated with domain name issuance at time of use. In this way, every CA would be a domain name service provider and every domain name service provider would be a CA. This would result in combining two markets and reducing competition. It would also exacerbate the difficulties of changing certificate providers. Given that certificate providers often also provide hosting, this proposal would fundamentally change the economics of small-scale hosting with less competition and no empirical evidence that the results would be positive for the end user.

## 6 Best Practices

The X.509 standard itself sets a very low bar for what constitutes a valid certificate. As a result, industry consortiums mandate further requirements, and many of these requirements are obligatory for inclusion in the trusted root certificate list of web browsers. There are a number of issuance best practices that can be added on top of these requirements. Although legacy requirements are chiefly why these best practices are not yet required, they should eventually become so.

The first best practice is the use of strong cryptography. RSA remains the dominant public key algorithm for certificates, and the cryptographic community recommends at least 2048-bit keys for end entity certificates. Where possible, elliptic key algorithms should be considered instead of RSA, as support for these algorithms becomes increasingly deployed. Elliptic curve keys should have at least 256 bits of length for end entity certificates.

Although SHA-1 has not yet been shown weak in the same ways as MD5, there is concern about its long-term security, and so the SHA-2 family of hash algorithms should be employed as part of the signature algorithm as much as possible.

The use of strong cryptography can then be augmented by applying reasonable validity periods to the certificates, such as one to three years for end entity certificates, to limit the exposure of any future attacks.

For cases where there is either compromise of a particular certificate or an attack against an entire class, CAs should include usable revocation information in every certificate. Usable means this revocation information should be usable by every major browser and web app that supports any kind of revocation checking. Although the particulars of revocation checking are beyond the scope of this document, there can be none at all if the CAs do not participate.

The purpose of the certificate is not only to enable a key exchange to occur. Its purpose is also to bind the identity of the server to a particular principal, such as a person or a corporate entity, which has the authority to use that domain. Wildcard certificates arose with the expectation that all the servers under a particular domain name would all belong to the same principal, and therefore it was an acceptable optimization to use a single certificate for a larger of server names, given that each individual certificate incurs a certain cost.

The advent of multi-tenant environments has turned this expectation on its head. Hosting providers that use load-balancing SSL terminators out of necessity deploy certificates with multiple domain names being used by multiple different customers, but because of the structure of an X.509 certificate, only a single subject name is present: that of the hosting company. The registered owner of the domain exists as a point of contact, but the SSL certificate itself doesn't correctly identify the site's owner. But if the hosting provider allocates hostnames from its own domain name but uses a wildcard certificate, not even that information is available. The use of a wildcard certificate in this case, while expedient, breaks a fundamental assumption of the certificate-based identity model. Therefore for each site operated by a different entity, CAs should issue unique certificates as much as possible. In situations where this is not possible, such as the SSL terminator scenario mentioned previously, the CAs should maintain records of attestations from the hosting provider that the domain owners authorize this use.

Recall the Extended Key Usage (EKU) extension which indicates the purpose or valid use of a certification. A best practice is for CAs to issue separate certificates for separate purposes and not combine multiple unrelated EKUs in a single certificate. Further, although not all certificate chain engines check "transitive EKUs" where not only must the end entity certificate possess a certain EKU but all CAs along the path to the root must as well, it is still a best practice for a CA to segregate its intermediate CAs by intended purpose, such as server authentication or code signing, and embed EKUs in the certificates of those CAs as well, so that a compromised CA is still limited to the purposes for which it was originally created.

## 7 A Practical Proposal

The best practices above are a solid starting point for best practices for depository institutions. If certificates are a part of operational risk for an individual institution, then systematic weaknesses in the PKI protecting depository interactions are part of the systematic risk for a banking system. Thus there should be some at least a minimal standard.

Improving certificate practice is not unachievable. Banks, citizens, customers, creators of web browsers, and other legitimate businesses all have a shared interest in recognizable banks online. Creating a mechanism for distinguishing and recognizing banks encourages online banking and online trust. The technical entities understand the requirements for certificates. The regulatory authorities understand the nature of systematic risk. A collaboration that consists of major cloud providers, banking regulators, cryptographic and interaction experts, browser manufactures, and selected banks could feasibly develop best practices suitable for depository institutions.

The situation currently embeds avoidable risks that are not addressed by any of the proposed solutions discussed above. Consider the lack of association between domain names and certificate for tens of thousands of banks. This would not be resolved by any current proposal. In terms of revocation, phishing domains are already in a race with those who defeat them. The domain is detected, labeled as malicious, then associated with a warning, and take-downs usually occur within a day or so [20]. Thus there is no history for comparison so historically based proposals would fail. Similarly with whitelists, warnings are issued when certificates are inconsistent with the observatory. New certificates are not flagged when they first present, and in fact this attack could be exacerbated by the existence of the observatory.

Similarly, DANE would offer no defense to this or any other attack with a legitimate domain name, including cloud misuse. Were domain name providers capable of not issuing domain names to malware providers, botnet controllers, and other malicious parties there would be less of an issue with these threats. DNS sellers are not appropriate gate-keepers. In terms of history, the tens of thousands of banks that do not associate a domain name with a certificate could clearly be subject to masquerade as there would be no previous certificate. DANE's reliance on DNSSEC results in all the problems of DNSSEC being a component of certificates. The problems of DNSSEC are both well-documented (e.g., [21, 22, 23]) and beyond the scope of this work.

Certificate authorities would seem to be the natural choice of critical actors. However, certificate authorities have not resolved this issue. Rogue certificates have been documented as having been issued by six CAs in recent years (Comodo [24], DigiNotar [25], DigiCert [26], TurkTrust [27], French Government CA [28], and India CCA [29]). Nor is this a recent problem. Perhaps most famously, VeriSign issued two certificate's in Microsoft's name in 2001 [11], for which Microsoft could only issue a security bulletin as removing VeriSign as a trusted CA was clearly infeasible (MS01-017).

Consider the best practices in Section 5. In the case of cryptography, RSA remains the standard and a minimum key size of 2048 is recommended. The elliptic curve standard is under discussion in both the IETF and W3C. Thus any requirement for elliptic curve would be premature. The issues with the Dual EC DRBG NIST, specifically the potential back door [30], and Operation Bull run [31] have reasonably resulted in decreased trust in this standards body in the case of encryption standards. While the challenges of operational risk can be handled in part at a national level, cryptographic standards for browsers and interoperability cannot be based on untrusted curves. Thus we recommend the use of RSA with a key size of 2048 bits as market acceptance will not be problematic.

Similarly requiring SHA-2 is a reasonable and arguably necessary step.

Maximum validity periods could be determined empirically. A single year would be ideal; however, two is not beyond the pale. The longest lifetime we have seen in our compilation is forty years (happily, not from a bank). Clearly this is not reasonable.

Wildcards should be discouraged, with a unified certificate issuer being an ideal practice for larger multi-domain entities. Wildcards should be prohibited in multi-tenant environments in the case of hosting services for a depositor entity. A Federally insured bank with a domain name should reasonably be expected to have the corresponding certificate even if that certificate is associated with the domain name as a second-level instead of first-level certificate. Multi-tenant environments can support a unified certificate issuer but may be unable to support domain-specific certificates.

Thus the only significant change we propose is the creation of a certificate verification authority for FDIC-insured entities. This could be used to validate a certificate regardless of where it is hosted. Without the ability to identify a remote entity as a bank, masquerade attacks on the financial system will continue. Having an additional signature, one that is constant across all FDIC-insured entities, can be integrated with the current authentication extensions in Firefox, Chrome, and Internet Explorer to prevent entering banking credentials into non-banking sites. Yet this requires the ability to distinguish general-purpose certificates, malicious certificates, and banking certificates.

Therefore we propose that a Federal entity, either Treasury or the FDIC itself, take two actions.

First, we propose the development of technical requirements for certificate issuance. Minimal requirements to control operational risk are not in any way a banking innovation. Defining maximum lifetimes, minimal cryptographic strength and recommending extensions are a feasible and reasonable way forward.

Second, we propose the cooperative development of a third party certificate notarization authority that applies only to banks. Notice that while this would be a signing authority, it would provide notarization of certificates provided by current CAs. Such a notarization could provide proof that a domain name was operated by a legitimate bank. Rather than having every domain name

reseller attempt to prevent any misleading domain name, this would distinguish legitimate banking sites from other sites.

Of course, this could also provide value to cloud service providers. Providers are currently challenged in that every customer has the right to use the infrastructure of the cloud. By associating financial institutions from other institutions, it has the potential to decrease the need for investment by cloud providers on providing certificates to every hosted site. By making this a second signature, rather than a whitelist or history approach, then citizens can use this list without being required to report their banking or browsing habits to any third party. Such collaborative reporting has proven acceptable in

We propose that this is feasible due to the small number of browser providers and cloud providers at this time. More secure interactions is in the interest of these parties. By augmenting rather than replacing certificate authorities, there is no replacement or decrease of business. In fact, limiting the lifetime of certificates is aligned with CA incentives.

The problem of identification of banks is not solved. However, password managers have the ability to identify and associate passwords with specific sites. A proof of concept Firefox extension would identify banking website via domain, certificate and activity. With this information, any reuse of the credential in a non-banking site is flagged. Were banks, and at some point credit unions, easily identifiable such an approach could become far more simple to implement.

Our proposal does not solve the problem of human factors. In no small part this is because human factors in security is a recent domain, with many problems identified but few solutions proven. Thus, like elliptic curve, standards would be premature. However, it does provide the structure to solve the challenge of human factors by enabling a simple answer to the basic query, "Is this a bank?"

## Acknowledgments

Research was sponsored by the Army Research Laboratory and was accomplished in part under Cooperative Agreement Number W911NF-13-2-0045 (ARL Cyber Security CRA). This material is based upon work supported, in part, by the National Science Foundation under Grant CNS 1250367; DHS BAA 11-02-TTA 03-0107, and Google. The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the Army Research Laboratory, Department of Homeland Security, Google, Microsoft, NSF, Indiana University or the U.S. Government. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes notwithstanding any copyright notation here on.

## References

[1] L. J. Camp, "Beyond usability: Security interactions as risk perceptions," in *Proc. of SOUPS '13*, 2013.

[2] R. Biddle, P. C. van Oorschot, A. S. Patrick, J. Sobey, and T. Whalen, "Browser interfaces and extended validation ssl certificates: an empirical study," in *Proceedings of the 2009 ACM workshop on Cloud computing security*. ACM, 2009, pp. 19–30.

[3] A. M. McDonald and L. F. Cranor, "The cost of reading privacy policies,," *ISJLP*, vol. 4, p. 543, 2008.

[4] R. Dhamija, J. D. Tygar, and M. Hearst, "Why phishing works," in *Proceedings of the SIGCHI conference on Human Factors in computing systems*. ACM, 2006, pp. 581–590.

[5] J. Sunshine, S. Egelman, H. Almuhimedi, N. Atri, and L. F. Cranor, "Crying wolf: An empirical study of ssl warning effectiveness." in *USENIX Security Symposium*, 2009, pp. 399–416.

[6] A. Sotirakopoulos, K. Hawkey, and K. Beznosov, "On the challenges in usable security lab studies: Lessons learned from replicating a study on ssl warnings," in *Proceedings of the Seventh Symposium on Usable Privacy and Security*. ACM, 2011, p. 3.

[7] A. Saldhana and T. Roessler, "Web security context: User interface guidelines," *World Wide Web Consortium LastCall WD-wsc-ui-20100309*, 2010.

[8] M. L. Johnson and M. E. Zurko, "Security user studies and standards: Creating best practices," in *Proceedings of Workshop on Security User Studies, San Jose, California*, 2007.

[9] L. Camp, H. Nissenbaum, and C. McGrath, "Trust: A collision of paradigms," in *Financial Cryptography*. Springer, 2002, pp. 91–105.

[10] C. Ellison and B. Schneier, "Ten risks of pki: What you're not being told about public key infrastructure," *Comput Secur J*, vol. 16, no. 1, pp. 1–7, 2000.

[11] R. Forno and W. Feinbloom, "Inside risks: Pki: a question of trust and value," *Communications of the ACM*, vol. 44, no. 6, p. 120, 2001.

[12] Microsoft, "Microsoft security advisory 2718704: Unauthorized digital certificates could allow spoofing," http://technet.microsoft.com/en-us/security/advisory/2718704, 2012.

[13] J. Braun and G. Rynkowski, "The potential of an individualized set of trusted cas: Defending against ca failures in the web pki," in *Social Computing (SocialCom), 2013 International Conference on*. IEEE, 2013, pp. 600–605.

[14] J. SEGURA, "Cyber-criminals interested in microsoft azure too," https://blog.malwarebytes.org/fraud-scam/2014/04/cyber-criminals-interested-in-microsoft-azure-too/, Apr. 2014.

[15] Z. Dong, A. Kapadia, and L. J. Camp, "Pinning & binning: Real time classification of certificates," 2013, poster presented at ACSAC 2013, Dec. 9–13, New Orleans, LA.

[16] D. Wendlandt, D. G. Andersen, and A. Perrig, "Perspectives: Improving ssh-style host authentication with multi-path probing," in *ATC*, vol. 8, 2008, pp. 321–334.

[17] M. Marlinspike, "The convergence project," convergence.io.

[18] A. Tsow, C. Viecco, and L. J. Camp, "Privacy-aware architecture for sharing web histories," *IBM Systems Journal*, vol. 3, pp. 5–13, 2007.

[19] R. L. Barnes, "Dane: Taking tls authentication to the next level using dnssec," *IETF Journal*, October 2011.

[20] T. Moore and R. Clayton, "Examining the impact of website take-down on phishing," in *Proc. of eCrime '07*, 2007, pp. 1–13.

[21] E. Osterweil, M. Ryan, D. Massey, and L. Zhang, "Quantifying the operational status of the dnssec deployment," in *Proceedings of the 8th ACM SIGCOMM conference on Internet measurement*. ACM, 2008, pp. 231–242.

[22] H. Yang, E. Osterweil, D. Massey, S. Lu, and L. Zhang, "Deploying cryptography in internet-scale systems: A case study on dnssec," *Dependable and Secure Computing, IEEE Transactions on*, vol. 8, no. 5, pp. 656–669, 2011.
[23] W. Lian, E. Rescorla, H. Shacham, and S. Savage, "Measuring the practical impact of dnssec deployment." in *USENIX Security*, 2013, pp. 573–588.
[24] P. Hallam-Baker, "Comodo ssl affiliate the recent ra compromise," https://blogs.comodo.com/uncategorized/the-recent-ra-compromise/, 2011.
[25] D. Fisher, "Diginotar says its ca infrastructure was compromised," *Threat Post. The Kaspersky Lab Security News Service*, 2011.
[26] DigiCert, "2nd clarification statement by digicert sdn berhad," https://www.digicert.com.my/news/news_20111107.htm, 2011.
[27] Microsoft, "Microsoft security advisory 2798897: Fraudulent digital certificates could allow spoofing," https://technet.microsoft.com/library/security/2798897, 2013.
[28] ANSSI, "Revocation of an igc/a branch," http://www.ssi.gouv.fr/en/the-anssi/events/revocation-of-an-igc-a-branch-808.html, 2013.
[29] Microsoft, "Microsoft security advisory 2982792: Improperly issued digital certificates could allow spoofing," https://technet.microsoft.com/en-us/library/security/2982792.aspx, 2014.
[30] D. Shumow and N. Ferguson, "The possibility of a back door in the nist sp800-90 dual ec prng," http://rump2007.cr.yp.to/15-shumow.pdf, 2007, presented at Crypto 2007 Rump Session, Aug. 19–23, Santa Barbara, CA.
[31] N. Y. Times, "Secret documents reveal n.s.a. campaign against encryption," http://www.nytimes.com/interactive/2013/09/05/us/documents-reveal-nsa-campaign-against-encryption.html, May 2013.