**Related Works**

There are many works published on securing SSL in infrastructure. The Related works section is divided into the following sections: Phishing and Browser related User Vulnerabilities, PKI Trust Infrastructure, and SSL protocol Vulnerabilities.

**PKI Trust Infrastructure**

**Why CA trust is important, suggest this should lead the related work section**

The security of applications such as SSL is dependent on the security of the Certification Authorities (CA) used to implement the functionality (big paper, can add here if needed). \cite{kent1998evaluating} CAs can’t necessarily be trusted in SSL. Various attacks...

\cite{park2012web}

It is important to analyze various properties of a CA, such as certificate type, security risks of the key holder, certificate and organizaiton of CA. \cite{zhang2010improved}

Trusting of CA roots: It is possible to pose as the MITM and insert a certificate signed by a different root CA due to the browser’s trusting of multiple roots. (the problem with multiple roots cite)

**An attack on the PKI model even when CAs are valid**

Even if CAs are secure, most novice users are susceptible to what is known as a Phishing attack. Therefore it is important to consider this style of attack and the presence of novice security users when designing (our CA plugin name). Phishing is a big problem (etc if we need more)...

Dhamija studied the effectiveness of phishing in. \cite{dhamija2006phishing}

rafftseder analyzed the development of an anti phishing plugin. \cite{raffetseder2007building}

Several other (are they similar?) solutions to phishing attacks in the form of browser plugins have been proposed. \cite{joshiPhishguardPlugin} \cite{mahmood2006Plugin} \cite{upadhyayaPlugin}

Wu said something about the effectiveness of these toolbars and plugins \cite{wu2006securitytoolbars}

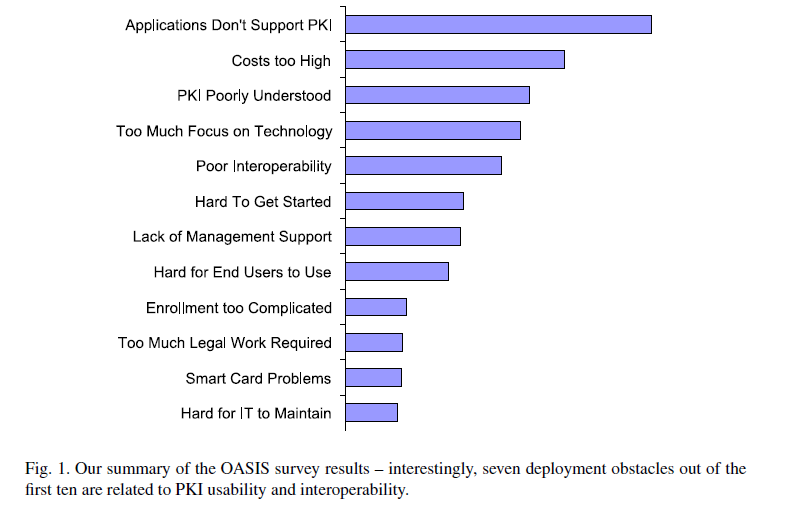
The next generation of phishing attacks will prove to be more sophisticated than current attacks strictly focused on collecting personal information. Due to current SSL technology being a one-way, server to user, system, a MITM attacker can now authenticate credentials in real time and present a seemingly valid phishing site to the user. Hashing of the user’s password with the website’s public key can be used to prevent these types of attacks. \cite{joshiMitigatingMITM} A hashing plugin is originally described in PwdHash, where a browser plugin hashes the user’s password with the website’s domain name, thus making the password hashed password useless for other websites. \cite{ross2005strongerauthen}

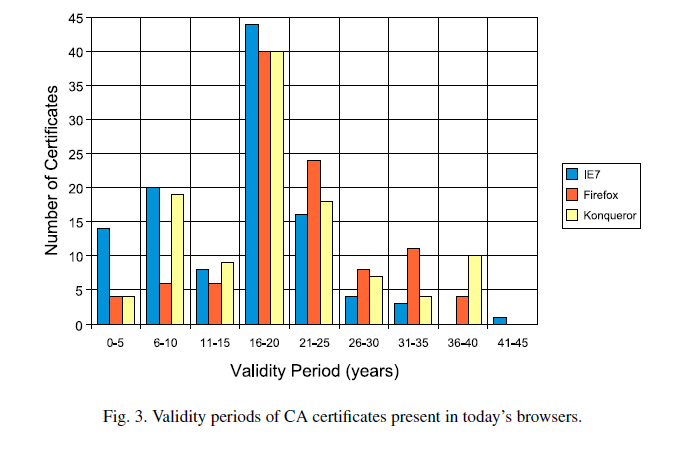
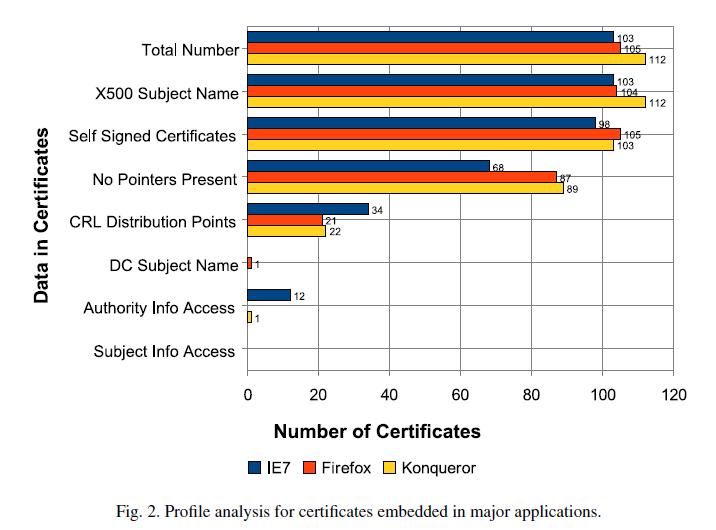
**PKI scalability issues - \cite{slagell2004pki}**

present the design and prototype of a new and flexible solution for automatic discovery

of the services and data repositories made available by a Certificate Service Provider (CSP). Addresses resource lookup problem: certificate retrieval, new service discovery. Three primary methods(to obtain pointers to PKI data) tradiationally: adopting specific certificate extensions, looking at easily accessible repositories, adapting existing protocols.

\*\*\* Introduction of the PRQP protocol. PKI Resource Query Protocol.





Finding the PKI needles in the Internet haystack - \cite{pala2010finding}

Reviews PKIX(X.509), SPKI, and PGP as base models.

In all of these systems there is a need to perform both efficient

enrollment and revocation.

**PKI seeks a trusting relationship - \cite{josang2000pki}**

Discusses Trust management of PKI infrastructure.

**Introduction to public key technology and the federal PKI infrastructure - \cite{kuhn2001introduction}**

Publication by NIST on todays PKI infrastructure and how it is applied.

**Ten risks of PKI: What you're not being told about public key infrastructure - \cite{ellison2000ten}**

Discusses Risks of PKI.

**Public Key Superstructure - \cite{wilson2008public}**

Paints optimistic picture of current PKI and builds on it. Introduction of Public Key Superstructure to describe a new way to knit together existing mature PKI components to improve utility and practicallity of Digital Certificates.

**Network security: private communication in a public world - \cite{kaufman2002network}**

Course textbook describing various types of PKI hierarchies.

**VeriKey: A Dynamic Certificate Verification System for Public Key Exchanges - \cite{sharifiVeriKey}**

**SSL Vulnerabilities**

**Proposal to authenticate the user to avoid MITM attacks**

SSL security is usually one way, server to user. Herzberg et al. propose a system, TPL, in which a certificate authority authenticates the user. \cite{herzberg2000access}

**A Scheme to improve security of SSL - \cite{huawei2009scheme}**

Proposes a new scheme (modify hand-shake protocol) which remedies the security defects of SSL as well as add valid SSL session key.

**Certified lies: Detecting and defeating government interception attacks against ssl (short paper) - \cite{soghoian2012certified}**

**Trust Darknet:Control and Compromise in the Internet's Certificate Authority Model - \cite{roosaTrustDarknet}**

**Browser(User) Vulnerabilities and Phishing**

**More on novice users and the importance of our “learning” ability in our plugin**

Security settings are generally difficult to configure. A complex learning approach, similar to the one used in our more simple (CA plugin name), was proposed by Sharifi et al. \cite{sharifiPersonalizedSecurity} The system works by monitoring user habits such as ignoring certificate warnings. The system will then either automatically choose to ignore such warnings, or prompt the user with a more thorough explanation of their risky behavior.

Users generally ignore or don’t notice SSL warnings. Maurer et al. propose and analyze a plugin to display larger warning messages \cite{maurerShiningChrome}

**“Do security toolbars really prevent phishing attacks” was listed twice by Yuri on accident, I will replace it with another article on my end.**

**Client-side defense against web-based identity theft - \cite{chou2004client}**

Discuss some aspects of common attacks fraudulent phishing/carding.

\*\*\*Introduces SpoofGuard: propose a framework for client-side defense: a browser plug-in that examines web pages and warns the user when requests for data may be part of a spoof attack.

Spoof site properties:

- Imitate appearance with logos

- suspicious URLs

- Fool user into entering sensitive information

- Copy HTML from honest site and make minimal changes to imitate

- Sloppiness and improper use of English

- HTTPS is not commonly used on spoof sites

Solution is three groups of tests: Stateless methods, stateful methods, and methods that evaluate outgoing html post data.

SpoofGuard Main modules:

Warnbar, ReflectionWind, UWToolBar, ConfigDlg, DomainDlg

**Phishing forbidden - \cite{agarwal2007phishing}**

• Users do not know which indicators are trustworthy.

• The browser indicators can be easily spoofed (e.g., by

including them in the page or painting over them with

chromeless windows).

• Users do not look outside their primary areas of interest.

Internal eye-tracking studies done by Yahoo! on login

pages showed that users see only the small rectangle

bounding the username and password fields of the page.

\*\*\* Yahoo! sign-in seal suggested\*\*\*

**Browser interfaces and extended validation SSL certificates: an empirical study - \cite{biddle2009browser}**

Extended Validation SSL study and IE 7- New design using “identity” and “confidentiality” icons.

**Trusted paths for browsers - \cite{ye2005trusted}**

Discusses problems with initial investigations i.e. spoofing.

\*\*\* Proposed solution of (SRD) Synchronized Random Dynamic boundaries. User’s browser displays graphical elements to the user.