Trustworthy Identity Management for Web Authentication

Ramasivakarthik Mallavarapu
Aalto University, School of Science and Technology
kmallava@tkk.fi

Abstract

Identity theft today is one of the major security issues on the Internet. Phishing and pharming are two of the most prominent web based attacks typically employed in order to direct users to fake websites for stealing critical user information. In addition to the web based attacks, malicious code residing on a user's computer may launch malware attacks including attacks based on web browsers. An important reason for increase in the onslaught of identity fraud is due to the inefficiency of existing identity management architectures in thwarting threats involving identity fraud. An average Internet user's inability to distinguish a legitimate website from fraudulent ones is also a significant reason in establishing identity fraud as a major Internet crime. In order to improve security incorporated in to web authentication, there is a need to examine architectures that provide trustworthy identity management. In this paper, we present and analyze some of the secure identity management architectures that could serve as alternatives to the traditional identity management architectures.

1 Introduction

Identity fraud has emerged as a significant threat not only in affecting a user's privacy but also in leading to sophisticated attacks that might eventually incur heavy financial losses [2]. Phishing and pharming have surfaced as two of the most important attacks in stealing user credentials apart from various other attacks like cross-site-scripting and request forgery. The attacks often result in compromise of the computing platform. Although web browsers and operating systems have been trying to improve the user interfaces to make the user aware of such attacks, they have not been entirely successful in countering them [12]. In addition to these attacks, malware and rootkit based attacks are being implemented at such an extent that the operating system may not detect its presence which leads to compromise of the entire system. For instance, attacks based on BIOS of a system may go undetected by the operating system. Although these attacks are not very common, they pose an interesting security challenge of building trust from a trusted entity that cannot be tampered. Hence there is a need for examining trustworthy identity management architectures [7].

As more and more enterprises have significant presence on the internet, strong identity management and authentication mechanisms are considered as an essential requirement. The focus has been to simplify the identity management process both for the end users and also for the enterprises. Federated identity management architectures provide businesses, a way of separating authentication from business logic. Federated identity management architectures like Single sign-on etc provide efficient identity management to enterprises but the client computers may be affected with malware which compromises security. In order to verify the authenticity and integrity of such transactions, with focus on the end users, the issue of trust usually boils down to the web browser or the underlying operating system. Both, web browser and operating system are based on software and unfortunately have been prone to attacks by various malicious programs over the years. To address these issues, trustworthy identity management architectures that primarily rely on a trusted entity might serve as an interesting alternative to the traditional identity management architectures. In this paper we examine architectures based on a trusted proxy and on the Trusted Computing Hardware [5].

The remainder of the paper is organized as follows. Section 2 provides the necessary background information about TPM. Section 3 details the relevant threats and security objectives. Section 4 describes a few of the identity management architectures involving trusted entities as a proxy between web server and the un-trusted machine. Solution based on mobile phone as a trusted entity is also presented. Section 5 details the analysis of various solutions described in this paper. Section 6 concludes.

2 Background

In this section we present a brief introduction of Identity Management and Trusted Platform Module.

2.1 Identity Management

Digital identity management refers to representing and identifying entities in computer networks. Identity management is fundamental in implementing complex security functionalities like authorization and access control. Digital identities are managed by the service providers by assigning a one-one key-value pair with each user having a unique identifier that represents the user and a shared secret known only to the user and service provider. Service providers usually maintain a database with a one-one mapping of user identifiers with their respective secrets. With the ever increasing presence of enterprises on the internet, the issue of digital identity management has become an important aspect of security from an enterprise's perception as well as the end user utilizing the services offered by the enterprise. In order to simplify the identity management process and reduce the re-

dundant identities, several architectures have been proposed to separate identity management from business logic. Few identity providers catering to a large number of enterprises also favors the end users by considerably reducing the number of secure credentials that a user needs to be reproduce in order to authenticate to the service provider [8]. In this paper, architectures based on Trusted Computing Hardware and proxy based identity management architectures are discussed in detail.

2.2 Trusted Platform Module

The Trusted Platform Module (TPM) as specified by the Trusted Computing Group (TCG) [13] provides security mechanisms like protected storage, integrity measurement and support for cryptographic computations. It is implemented as a hardware chip and is mounted on a platform. The communication between TPM and all the other devices is facilitated through a hardware bus. In order to create a sequence of trust, there has to be a trusted component that verifies the trust associated with all the other components. Every component in the system is measured for trustworthiness against a pre-loaded configuration. The process continues recursively until all the essential components are measured for trustworthiness. The combination of all the trusted components in a system is referred to as *Trusted Computing Base* (TCB). In case of a typical computer system, the process could start with *Core Root of Trust for Measurement* (CRTM), an extension of BIOS that verifies the integrity of BIOS operations during the boot process. CRTM verifies the *BIOS*, BIOS verifies the *Boot Loader*, which in turn verifies *Operating System Kernel* and finally the *Operating System* (OS) [13, 7].

TPM serves as the base component from where the chain of trust may begin, as it provides functionalities to measure a platform's trustworthiness. TPM provides two pairs of asymmetric keys, *Endorsement Key* (EK), *Storage Root Key* (SRK). SRK is used to encrypt all the other sensitive information that must be stored on a TPM, while EK is meant for digitally signing the content with a private key which is unique to a TPM [4]. *Sealing* refers to the process of encrypting sensitive information by the TPM using non-migratable keys unique to that particular TPM. TPM contains *Platform Configuration Registers* (PCR) that are used to store the hash values of specific platform configurations which are an important part of integrity measurement. PCR can only be accessed through `extend` operation that calculates the hash value of the platform configuration along with the existing PCR value. The new PCR value overwrites the current PCR value. A sealed value can only be unsealed by providing an authorization data that includes the current platform configuration. Unsealing of sensitive information is bound to the platform configuration and the authorization data. The TPM verifies for a match between the current platform configuration values and the values of platform configuration at the time of sealing. Only in case of a perfect match, the information is unsealed [5, 13, 7].

The TPM includes a cryptographic engine that offers some basic security features such as random number generation, encryption/decryption, key generation and hash value generation. Apart from these, TPM also provides shielded locations for tamper resistant storage [12]. Shielded locations may only be accessed through commands called *protected capabilities*. TPM offers ways to attest the trustworthiness of a platform through attestation mechanisms. *Attestation* is an essential function of TPM that asserts the trustworthiness of a platform to a remote party. Attestation process also involves a certification authority, referred to as privacy CA. Privacy CA is a trusted third party that verifies the integrity of a platform in question. In order to achieve confidentiality of sensitive information and integrity, TPM has components called Root of Trust; one each for storage, measurement and reporting. *Root of Trust for Measurement* (RTM) is a way of measuring a platform's integrity. The integrity measurement operations are performed by the CRTM [13, 9].

2.3 Root of Trust for Measurement

Root of Trust for Measurement aims at ensuring a trusted boot sequence. *Static Root of Trust for Measurement* (SRTM) constitutes the measurement of a normal boot sequence from BIOS to all the components till the OS is loaded. SRTM builds trust from the boot time and performs load time measurements. This approach has had a few disadvantages as run time measurements might differ from the load time measurements [9, 13]. *Dynamic Root of Trust for Measurement* (DRTM) does not operate at boot time but provisioning the initialization of RTM at any point of time as required. This opens up the opportunity of performing run time measurements in contrast to the load time measurements performed by the SRTM. An important feature is the introduction of dedicated instruction to enable this particular feature. Intel [3] and AMD [1] have introduced specific instructions to make use of DRTM for measuring run time integrity. The instruction re-initializes the CPU and brings the CPU back to a secure state in order to enable a secure execution environment [6].

3 Threat model

Threat modeling refers to identifying threats affecting a system and the subsequent risks associated with those threats. Threat modeling is an iterative process that requires careful attention during the entire life cycle of an application. We define threats and security objectives that might be applicable to authentication in the web [11].

3.1 Threats

- **Identity theft** Identity theft refers to stealing of valuable user credentials. Phishing and pharming are two specific attacks that can be categorized under identity theft.
- **Malware** Malware residing on a user's computer or a remote user using an un-trusted computer to perform secure transactions may result in capture of user credentials by the underlying malware.
- **Unauthorized Access** Unauthorized access to a trusted device like a trusted proxy or a mobile phone creates an

opportunity for the attacker to launch attacks that may lead to severe damage [12].

3.2 Security Objectives

- **Strong passwords** Strong passwords are essential to counter password guessing attacks and dictionary attacks.
- **Secure execution** Secure execution environment for a trusted entity is vital in order to ensure that malicious entities do not corrupt the functioning of trusted entities.
- **Protection of long term secrets** In an un-trusted environment protection of long term secrets may involve usage of short lived or one time passwords to mitigate the loss of confidentiality of long term secrets.
- **Trust relationship** Establishment of trust between a trusted user on an untrusted computer and a trusted proxy or web server is a significant security challenge with imminent risks of attacks like session hijacking, Man-in-the-middle attack and other re-direction attacks.

4 Trustworthy Identity Management

Traditional identity management architectures often employ a username, password combination to authenticate a user on the web. Due to limited technical knowledge about information technology and web security, an average user is ill-equipped to detect simple identity fraud. Personal computers increasingly have become victims of various kinds of attacks including attacks involving spying on the user's computer or stealing credentials by redirecting users to fake websites. To ensure confidentiality of sensitive information and integrity of transactions on the Internet, there is a need to examine alternate architectures that provide trustworthy identity management based on a trusted entity [10]. A few such architectures are presented in this section.

4.1 TruWallet

TruWallet is a wallet based authentication mechanism for secure web authentication. Wallet based approach provides a generic framework to counter the attacks prevalent in web authentication. A wallet provides a dedicated agent that performs secure authentication by operating in an isolated environment. TruWallet proposes an architecture that makes minimal usage of Secure Sockets Layer (SSL) Public Key Infrastructure (PKI) based approach. TruWallet also includes a migration protocol that uses attestation functionality of TPM to facilitate secure transfer of secrets from one wallet to another [12].

4.1.1 Architecture

TruWallet architecture is based on a security kernel which is an essential part of the TCB providing trusted services and isolated compartments. Compartmentalization ensures that objects of one compartment may not access objects or memory of other compartments. This kind of isolation helps trusted applications like TruWallet perform secure transactions residing amongst insecure applications, for instance, a compromised web browser. TruWallet makes use of TPM as a trusted way of storing long term secrets. The major components of TruWallet are trusted wallet acting as a web proxy, security kernel that provides a secure environment for the wallet and a secure user interface between user and the wallet [12].

TruWallet acts as a secure proxy between an un-trusted web browser and the server. The wallet handles two separate SSL sessions one with the server and the other with the web browser, thus acting as a secure proxy. Secure GUI is used as an interface between the user and the system by ensuring a trusted path between user and the application. It also tries to increase the visibility of attack scenarios by displaying name of the application with which the user is currently interacting with, on a reserved area on the screen. Security of the wallet is protected by restricting access to the compartment in which the wallet resides. Compartmentalization ensures that the malicious code running on a different compartment cannot affect the compartment where the wallet resides [12].

4.1.2 Secure User Authentication

Registration phase refers to the user creating a new account on a website. During the registration phase, the wallet creates a mapping between the website and the appropriate credentials that authenticates the user. At the end of the registration phase, SSL server finished message is used an additional shared secret (ss). As the server finished message is encrypted by the server using the shared secret key of the corresponding SSL session, the server finished message in plain text form serves the purpose of *ss*.

During the login phase, SSL is only used to provide a confidential channel and the shared secret ss from the registration phase is used to authenticate the server. To verify the authenticity of the server, a challenge response protocol is used where in the wallet sends all the SSL messages received by the wallet, referred to as transcript, as a challenge. The server must compute the hashed message authentication code (HMAC) of the transcript using the shared secret as the key value. The response R:$HMAC_{ss}$(transcript) will be checked against the transcript using the shared secret *ss* by the wallet at the client side. This verification is performed at the end of a successful SSL session between the wallet and the server. The wallet proceeds with the authentication only on a successful verification of server authenticity [12].

4.1.3 Secure Wallet Data Migration

The confidentiality of wallet data is bound to the integrity measurements of the TCB and data is sealed by the TPM. In order to transfer the wallet data in a secure way, the integrity of both the source and the target platform needs to be verified. Attestation functionality of the TPM may be used in order to assert the trustworthiness of both the platforms in question. A secure channel is established between the two wallet instances that paves the way for verification of a platform's integrity. The process involves attestation of PCR val-

ues of platform, certification of a privacy CA, a trusted third party that certifies that the TPM is genuine and finally asymmetric key pair exchange that is bound to the configuration values of a platform. The platform needs to be in a secure state in order to be able to decrypt the private key of the asymmetric key pair which is used to encrypt the data in the wallet. Successful decryption of the private key will only ensure secure data migration between the two wallet instances [12].

4.2 Delegate

Delegate is a proxy-based architecture for secure web authentication from un-trusted computers. Delegate mandates the use of a trusted proxy server and a trusted personal communication device in its architecture at all times. The major components of Delegate are

- **Un-trusted computer**
- **Trusted proxy**
- **Trusted mobile device**
- **Web server**

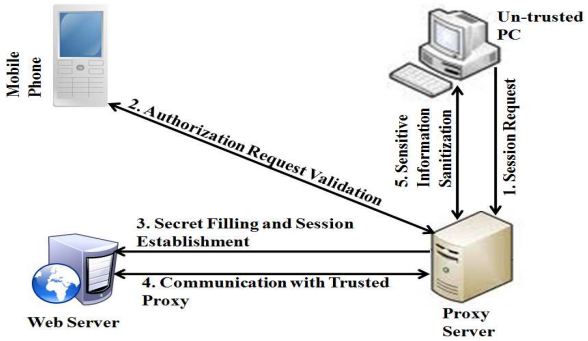

Figure 1: Delegate

As shown in Figure 1, the trusted proxy acts as the man-in-the-middle, intercepting and routing packets between web server and un-trusted computer. The role of a proxy is not just confined to establish a secure path between the web server and the un-trusted computer, but also to sensitize confidential information that is being sent to the un-trusted computer. Delegate makes use of a user's personal mobile device to communicate to the user, authorization and confirmation requests that arise from the un-trusted computer. Delegate tries to minimize the sensitive information that a user needs to enter from an un-trusted computer by performing the following functionalities [11].

- **Secret filling** User on an un-trusted computer will not enter any sensitive information that may lead to the compromise of long term secrets. Credentials are filled by the trusted proxy by matching the URL of the requested service with the appropriate secret from the database of passwords.
- **Authentication** The user must authenticate to the trusted proxy in order to utilize the services offered by the proxy. The user must enter a one-time password or a PIN which is communicated across to the user using the user's mobile phone. The user can also send confirmation to the proxy using the mobile phone. After authenticating to the proxy, the proxy acts as the trusted man-in-the-middle directing traffic between the web server and the un-trusted computer.
- **Removing sensitive information** Apart from the above mentioned functionalities, removing sensitive information is an essential part of Delegate framework. The sensitive data or metadata is filtered out by the proxy to make sure that no vital information reaches the un-trusted computer.

4.3 Mobile phone based TPM solution

Mobile phones have increasingly become powerful personal devices with significant improvements in processing capabilities. Also mobile phones are personal devices that are usually carried by the users. Trusted Computing Hardware embedded on to the mobile phone hardware provides an opportunity to rightly utilize the computing capabilities and also at the same time take advantage of the trust factor induced from a personal communication device like the mobile phone. Wallet based solutions based on minimal security kernels when operated on mobile phones open up newer avenues in the process of devising trustworthy identity management architectures with mobiles as central focus [11, 12].

Wallet based solution that exists on a mobile phone secured with the TPM could serve as the trusted entity that the user always carries with him/her. Protection of long term secrets can be achieved using the cryptographic functionalities provided by the TPM. Since the mobile phone is always within the accessible range of the user, it is easier to detect attacks. This kind of solution might scale well with users on home computers or for roaming users. In this section, we detail and analyze two such architectures.[11, 8].

4.3.1 Mobile Phone as Authentication Agent

In this model, we consider mobile phone acting as a trusted entity for authentication on behalf of the user on an un-trusted computer. Figure 2 depicts the identity management architecture with mobile phone as a trusted authentication agent. After the registration phase, the user's mobile phone and the server share a long term secret which may be used in computing the challenge and response messages.

4.3.2 Mobile Phone as Trusted Authentication Agent

The authentication process begins with the user on an un-trusted computer making a session request with the server. The server sends a challenge along with other credentials such as the server certificate, nonce and time stamp to mitigate the replay attack. All the components are encapsulated in to a single file (challenge file) for abstracting the user from all the lower level details. The response from the user side is also provided in the form of a single file (response file). The response file is the encapsulation of the response message, encrypted password and other essential components like the

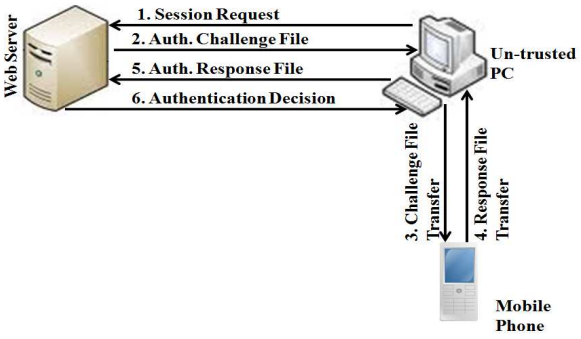

Figure 2: Mobile phone as authentication agent

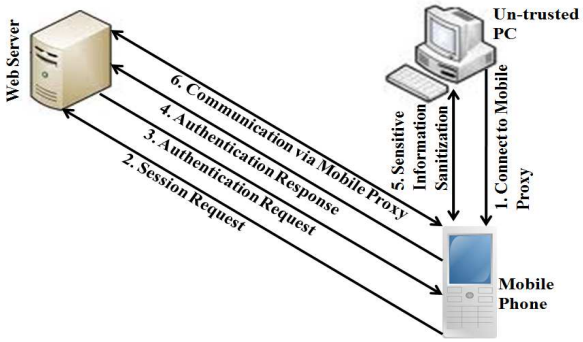

Figure 3: Mobile phone as trusted proxy

nonce, optional client certificate, time stamp etc. The process of providing the necessary credentials is always performed on the user's mobile phone in a secure environment and never on the un-trusted computer. TPM on the mobile phone provides a secure execution environment for performing the authentication on behalf of the user. The encrypted passwords are decrypted only for specific platform configurations ensuring a secure environment. The user must always transfer the challenge file to the mobile phone. At this step the user may require to authenticate to the mobile phone. The required response to the challenge and other necessary credentials are computed after the verification of server identity. Finally the response file is generated which is transferred back to the un-trusted computer and then sent to the server. After a successful authentication, the user communicates directly with the web server. Thus during the authentication stage, user's mobile phone acts as a trusted authentication agent [12].

4.3.3 Mobile Phone as Trusted Proxy

In this architecture, user's mobile phone plays the role of a trusted proxy re-directing traffic between web server and the user's un-trusted computer. Mobile phone handles the entire communication between the web server and the un-trusted computer by identifying authentication requests and providing authentication credentials to the web server. In order to prevent the leakage of sensitive information to the un-trusted computer, the proxy also sanitizes sensitive information arriving from the web server. Sanitization of sensitive information and preventing user from using the authentication credentials from an un-trusted computer may considerably reduce the risk of malware attacks on the un-trusted computer. Equipped with a TPM, mobile phone could provide a secure execution environment while operating on authentication credentials or sanitizing information. The model is quite similar to Delegate, with the only exception that the trusted proxy in this case is a mobile phone. Figure 3 depicts a high level architecture of this model [11].

The mobile phone may be pre-configured to act as the proxy for a few selected websites. The authentication credentials for those websites may be stored in a secure way on the mobile phone storage. TPM's cryptographic functionalities may be used for secure storage and retrieval of authentication credentials. Unsealing operation is bound by the PCR values at the time of sealing, ensures retrieval of credentials only under specific platform configuration values. In order for the user to connect to the proxy, the proxy could run a web service that can be accessed through a URL. This method of connection establishment could be fairly simple even for a naive user as it is equivalent to connecting to any other website. The user must authenticate to the mobile phone in order to make a successful connection. Once the connection is made, the mobile phone acts as a proxy for the entire range of communication between the user on an un-trusted computer and the web server. The authentication process is handled by the proxy on behalf of the user on the un-trusted computer.

After a successful authentication, the communication between the web server and the un-trusted computer is handled by the proxy, filtering sensitive information from the prying eyes of malware that might reside on the un-trusted computer. Sensitive information, for instance could be the social security number, bank account number etc. Thus users on un-trusted computers may perform important transactions without losing any sensitive information to the attackers.

5 Discussion

In this paper, we have detailed and discussed a few architectures that base their authentication process on some trusted entities. In this section we discuss various issues concerning the feasibility, compatibility with existing systems and security provided by each of these architectures. We also detail some of their advantages and disadvantages.

TruWallet architecture provides a framework for enabling secure user authentication. The main advantage of this model is compartmentalization that ensures secure execution. The access to wallet requires authentication that helps provide better security. Migration protocol defined in the TruWallet architecture provides a secure way of transferring wallet information by verifying the target platform's integrity, thus countering the threat of malware on the target platform. The architecture requires software changes on the server side and presence of TPM on the client side. The model does not mention about accommodating roaming users on un-trusted computers. TruWallet as a generic framework for wallet based approaches introduces a promising architecture that scales well with computers equipped with a TPM [12].

Delegate is a proxy based architecture that mandates the

use of a trusted proxy and a trusted personal communication device. Trusted proxy redirects traffic between the web server and the user on an un-trusted computer allowing the user to perform limited operations on the un-trusted computer. Critical operations like providing credentials are all handled by the proxy on behalf of the user. The proxy also sanitizes sensitive information before redirecting traffic from the web server to the user. This approach permits users to use un-trusted computers as the user never enters his/her secret credentials on un-trusted computers. Delegate employs the user's mobile phone to validate certain requests. The feasibility issues involved with this approach are that every user is mandated to have a trusted proxy and all the access to the web happens through the proxy. This can create a single point of failure as a non-functioning proxy would not let the user access the web. Delegate also requires the users to set up a trusted proxy server. The proxy server services could be offered by a third party trusted entity or the users themselves can configure a proxy server. Users without technical expertise in IT might find it extremely difficult to manage a trusted proxy server [11].

Mobile phone for secure authentication treats the mobile phone as an authentication agent that provides the necessary credentials for authentication. The complexity of the entire process could be simplified by encapsulating all the required information in the form of a single file. Mobile phone equipped with a TPM, provides secure storage of credentials by enabling decryption of encrypted credentials only under certain specific platform configuration values. Each time the user wishes to authenticate to the web server, the file transfer happens between the un-trusted computer and the mobile phone. File transfer may involve a wired transfer from the un-trusted computer to the mobile phone. Other ways of information exchange could be Bluetooth, WLAN etc. Feasibility of method of transfer and the overall time taken in order to complete the process of authentication needs to be sorted out in order to improve the feasibility and seamlessness of the architecture. The architecture allows the user to perform transactions on un-trusted computers as well. This architecture requires a change at the server side for supporting challenge response style of protocols and encapsulation of authentication data in to a single file.

Mobile phone as a trusted proxy between an un-trusted computer and the web server is an interesting model that requires the mobile phone to act as a web server. The user on the un-trusted computer connects to the mobile phone through a URL. Major issues with this kind of approach are

- Mobile phone must always have an IP address, preferably a static IP address.
- In order to connect to the mobile phone through a URL, there must be a unique domain name for the mobile phone.

However the positive aspect of this architecture is compatibility with the existing system. It does not require any major changes on the server side. Energy efficiency is one other issue crucial for mobile devices and mobile phones are no exception. In order to make the solution more efficient, the traffic re-direction may be used only for selected websites involving crucial transactions. Instead of relaying traffic for every website that the user visits, a few important websites may be a more feasible and efficient solution.

6 Conclusion

Digital identity management has grown in to a major security challenge that requires sophisticated approach to counter the growing threats. Traditional identity management architectures have not been entirely successful in dealing with the new security challenges in the Internet. Hence there is a need for a paradigm shift in the field of identity management. New approaches and novel solutions have evolved over time that envisages trusted entities with a crucial role of building a trustworthy relationship between a trusted user on an un-trusted computer and the web server. In this paper, we proposed, detailed and discussed a few architectures that rely on Trusted Computing Hardware and/or employ a trusted proxy to perform web authentication in un-trusted environment. The use of Trusted Computing Hardware like the TPM looks like a promising solution in building architectures with a generic framework to counter newer threats in the web. TPM based solutions for mobile phones with mobile phone as a trusted entity could induce greater levels of trust as a personal device is perceived as trustworthy by the users.

References

[1] AMD64 Virtualization: Secure Virtual Machine Architecture Reference Manual . Technical report, Advanced Micro Devices. , May 2005.

[2] 2009 Internet Crime Report . Technical report, Internet Crime Complaint Center., 2009. `http://www.ic3.gov/media/annualreport/2009_IC3Report.pdf`.

[3] Trusted eXecution Technology (TXT) — Measured Launched Environment Developer's Guide . Technical report, Intel Corporation., December 2009.

[4] Understanding Keys and Passwords Used by the TPM. 2010. `http://technet.microsoft.com/en-us/library/cc755038.aspx`.

[5] Barbara Fichtinger and Eckehard Herrmann and Nicolai Kuntze and Andreas U. Schmidt. Trusted Infrastructures for Identities. In *Virtual Goods: Technology, Economy, and Legal Aspects. Proceedings of the 5th International Workshop for Technical, Economic and Legal Aspects of Business Models for Virtual Goods, Koblenz, October 11-13, 2007*. Nova Publishers, 2008.

[6] S. Bugiel. Using TCG/DRTM for application-specific credential storage and usage . Master's thesis, Technical University of Denmark and The Royal Institute of Technology, the Netherlands, June 2010.

[7] S. Garriss, R. Cáceres, S. Berger, R. Sailer, L. Doorn, and X. Zhang. Towards trustworthy kiosk computing. In *In Workshop on Mobile Computing Systems and Applications*. Society Press, 2007.

[8] A. Jøsang, J. Fabre, B. Hay, J. Dalziel, and S. Pope. Trust requirements in identity management. In *Australasian Information Security Workshop*, volume 44, 2005.

[9] B. Kauer. Oslo: Improving the security of trusted computing. In *16th USENIX Security Symposium*, pages 229–237, 2007.

[10] A. Pashalidis and C. J. Mitchell. Impostor: A single sign-on system for use from untrusted devices.

[11] Ravi Chandra Jammalamadaka and Timothy W. van der Horst and Sharad Mehrotra and Kent E. Seamons and Nalini Venkasubramanian. Delegate: A Proxy Based Architecture for Secure Website Access from an Untrusted Machine. In *Twenty-Second Annual Computer Security Applications Conference, December 11-15*, 2006.

[12] Sebastian Gajek and Hans LÃűhr and Ahmad-Reza Sadeghi and Marcel Winandy . TruWallet: Trustworthy and Migratable Wallet-Based Web Authentication. In *ACM 978-1-60558-788-2/09/11*, November 2009.

[13] Trusted Computing Group . TPM Main Part 1 Design Principles . In *Specification Version 1.2*, 2007.