Journal of Universal Computer Science, vol. 16, no. 9 (2010), 1139-1151
submitted: 10/10/09, accepted: 21/12/09, appeared: 1/5/10 © J.UCS

A Trusted Computing Identity Collation Protocol to Simplify Deployment of New Disaster Response Devices

Peter Danner
(Institute for Applied Information Processing and Communications (IAIK)
Graz University of Technology, Austria
peter.danner@iaik.at)

Daniel Hein
(Institute for Applied Information Processing and Communications (IAIK)
Graz University of Technology, Austria
daniel.hein@iaik.at)

Abstract: The use of modern computing equipment by emergency service units in a disaster area assures increased efficiency during disaster response. Emergency devices must be easy to use and secure. Trusted Computing is a promising approach to help protect the software integrity of commodity emergency devices and thus increase their security. To efficiently use Trusted Computing in an emergency scenario it is necessary to establish an initial trust relationship between the emergency infrastructure providers and a user, her devices, and the software running on those devices. Currently, this requires physical presence of the involved entities. In this paper we propose a remote protocol that employs electronic identity facilities and Trusted Computing to aggregate the identity of a user, the identity of her devices and a set of trusted software states as well as the users facilities and skills. Such a protocol alleviates the need for physical presence. Thus, the protocol facilitates deployment of new electronic emergency equipment, while maintaining a high level of security. We belief that such a protocol is an important step in the process of introducing new capabilities for disaster response.

Key Words: trusted computing, disaster response, electronic identity, eID, TPM
Category: D.4.6, D.2.9, H.3.2, H.3.4

1 Introduction

Portable computing and communication devices offer the potential to increase the efficiency of emergency management systems and disaster response. For example, it has been proposed to employ mobile agent systems in disaster response [Schurr et al. 2005, Arunachalan et al. 2007]. Such applications require powerful security mechanisms. Recent results show that Trusted Computing is a promising technology to secure mobile agent systems [Balfe & Gallery 2007, Gallery et al. 2009]. Furthermore, during disaster response emergency service units often operate under adverse conditions and it cannot be assumed that a preset communication infrastructure is available. Therefore, the need for secure ad-hoc networking arises. Trusted Computing can help ensure the integrity and security of an ad-hoc network [Xu et al. 2007, Sandhu & Zhang 2005]. In addition to ensuring the *trusted state* of the mobile agent system, it is important to affirm the identity of the user of each device in the system. To establish these two properties a hardware security token, the *Autonomous Attestation Token (AAT)* [Hein & Toegl 2009], has been proposed.

One of the core concepts of Trusted Computing is establishing the *trusted state* of a platform. To achieve this end, Trusted Computing as propagated by the *Trusted Computing Group (TCG) [TCG 2003]* measures the state of a platform and stores this measurements in the *Trusted Platform Module (TPM)*. The TPM then uses a cryptographically secured reporting mechanism to attest the state of the platform to a verifier. This platform configuration report is called a *Quote*. By means of this *Quote*, the verifier decides if the attested platform is in a *trusted state*. A relying party can then base further interaction with the attester on this decision. The outlined process is called *Remote Attestation*.

In an Internet scenario the remote verifier is a powerful entity with access to a wide range of computational and informational resources. In a crisis scenario such a service might not be available. The AAT assumes the role of a local verifier. It regulates access to the communication infrastructure and the mobile agent system, by only granting access if the host system is in a *trusted state* and the user can provide a proof of presence. This proof of presence could be simply entering a short password or a Personal Identification Number (PIN). Thus, the AAT protects against unauthorized users and possibly maliciously modified platforms.

Provisioning the AAT with the information required to perform the trust decision and to authenticate a user is a major challenge. The setup of the AAT must be in compliance with a predefined procedure. This setup process must gather specific information from the applicant. An AAT must only be deployed after successful verification of this information.

On the one hand, it must be determined that an applicant's platform(s) posses a TCG standards conforming TPM, which is correctly integrated into the platform. The AAT enforces that the platform software configuration is in a *trusted state*. Therefore, it is necessary to measure the platform(s) state(s) when executing in a trustworthy software configuration.

On the other hand, the identity of an applicant and her authority to use the emergency communication network and services must be established. *Electronic Identities (eIDs)* provide a way to prove a persons identity, without requiring physical presence and verification of paper-based identity documents. In addition to identifying an applicant, eIDs have the potential to prove the role of an applicant in a crisis, for example that she is a doctor.

For further discussion we assume the existence of an *Emergency Service Infrastructure Provider (ESIP)*. This organization provides a protected communication network and access to services and information for disaster response. The scope of an ESIP could vary from a single emergency service unit, for example a metropolitan police to an international consortium of national and international emergency service organizations. One of the dominating factors for acceptance of the AAT will be its ease of use. This does not only pertain to an actual crisis situation, but also concerns the ease of distribution of the AAT. It must be simple for emergency service organizations and personal to obtain an AAT from an ESIP, and for the ESIP to manage the AAT. Therefore, the need for a remote protocol for AAT provisioning arises. In this paper we propose such a protocol.

An ESIP must establish four different facts to provision and issue an AAT. The first is the identity of the user. The second is her authorization to use the emergency service network and services. Third, it must be established that the device of the user is equipped with a TPM and last, the actual identity of the emergency software components on the target platform (measurements) must be collected. We propose a protocol that combines Trusted Computing and eIDs to collate the necessary information for the ESIP with a high level of assurance.

Outline

The next section of this paper discusses related work relevant to the protocol introduced herein. [Section 3] provides a concise description of Trusted Computing and eID. The core of this work, the protocol, is introduced in [Section 4]. Its security properties are discussed in [Section 4.2]. The paper gives an outlook on future work in [Section 5] and concludes in [Section 6].

2 Related Work

To reliably report the state of a platform by Attestation, it is necessary to map the state of a platform to a set of SHA-1 values. In [England & Loeser 2008] Paul England (convincingly) states that this mapping is impractical. He introduces several mechanisms to overcome this problem. One mechanism is to use a read only software image. This approach was chosen by Franklin et al. for their CA-in-a-Box [Franklin et al. 2005]. This poses the question of how to distribute this image to the user.

In [Weigold et al. 2008] Weigold et al. argue that the level of men-in-the-middle and malicious software attacks is increasing and that therefore no trust can be put into an end-user's machine. They propose the Zurich Trusted Information Channel, a USB device capable of exposing malicious modifications of critical information before it is sent over a cryptographically secured channel.

A way and a protocol using external keys to marry smart cards with TPMs is introduced in [England & Tariq 2009]. England and Tariq mutually exchange keys to enable secure communication between a TPM and a smart card. The aim of this marriage is to extend the "trusted" execution area to programs residing on a smart card. They use this to enhance the TPM with new TPM commands for testing purposes. In England and Tariq's case the smart card with its programs has to be examined physically before the coupling happens. The benefit of our protocol is that each evaluation can be done remotely by providing appropriate certificates or electronic data. A further difference of our protocol is that we use a special certificate issued by an acceptable registration authority and bind this to a third piece of hardware - the AAT. This AAT contains the trusted states for the platform(s) in which it can be used.

Klenk et al. are aiming to prevent identity theft by using the facilities of a TPM. In [Klenk et al. 2009] they bind a smart card's eID key to a TPM key. They correlate an OpenID user ([Ferg et al. 2007]) to a real physical user possessing an eID protected by a smart card. The OpenID provider is responsible for the Remote Attestation in the initial trust establishment process. If the trust decision is positive, the proof of the user's identity is transfused into the TPM of the requesting machine. After this process the user does not have to use her smart card each time she authenticates herself. In contrast we use the eID's signature to confirm the correctness of the platform and application data during the collection process.

Electronic Software Distribution (ESD) is concerned with the reliable delivery of software components to complex networks of embedded systems found for example in airplanes. Software delivery to such critical devices requires a high assurance level that the correct software, from the correct source, was timely delivered, and is compatible with the software of the other components. Maidl et al. introduce a formal security analysis of such a system in [Maidl et al. 2008]. Their system relies on a Software Signer Verifier which requires protection from manipulation.

Microsoft Windows BitLocker Drive Encryption is a data protection mechanism that can use a TPM. BitLocker trusts the user to accept the software configuration of the platform. A more in-depth analysis of BitLocker is described in [Türpe et al. 2009].

3 Background

3.1 TPM

The TPM [Trusted Computing Group 2007] is a tamper resistant hardware module similar to a smart card, and shipped with a variety of commodity PC platforms. It provides basic cryptographic services like a SHA-1 hash function and an RSA engine, and also a protected non-volatile storage area. Likewise, for mobile platforms, the *Mobile Trusted Module (MTM)* [Trusted Computing Group 2008] has been specified. The MTM standard defines different sets of features (profiles) that implementors can choose from. The TPM and all profiles of the MTM allow to implement a secure chain of measurements which enables the mapping of a platforms software configuration into a set of *Platform Configuration Registers (PCRs).*

The mapping of the platform state into a set of PCRs requires a Root-of-Trust for Measurement (RTM). The RTM is a trusted component that assumes control over the platform in a well defined state, for example after power-on. Depending on the nature of this initial component it performs a specific set of hardware initialization steps and then hands over control to the next component. Before this control hand-off, the initial component measures the next component, establishes it's unique identity in the form of a SHA-1 hash, and stores this hash to a PCR protected by the TPM. The next component in the chain proceeds to use the same *measure-before-invoke* paradigm and thus builds a chain of SHA-1 measurements. In order to limit the amount of storage space for the SHA-1 values the only way to write to a PCR is the PCR_EXTEND operation which works as follows:

$$PCR_i = SHA\text{-}1(PCR_i|x),$$

where i is the number of the PCR, SHA-1 is the SHA-1 hash function and x is a SHA-1 hash of the measured component. By virtue of the invertibility of the hash function it becomes computationally infeasible to fake a chain of measurements extended into a PCR.

Two different RTMs for desktop PC platform have been proposed. The *Static Root-of-Trust for Measurement (SRTM)* and the *Dynamic Root-of-Trust for Measurement (DRTM).* The SRTM is a component of the system's BIOS, which gains control early during the boot process and measures the BIOS before giving control to it. The BIOS repeats this with the MBR, the MBR with the boot loader etc.. This approach has several drawbacks, the foremost of which is that it includes the BIOS and a number of other hardware dependent low level components in the *Trusted Computing Base (TCB)*, the set of platform software that must be trusted. As a rule the TCB should be kept as small as possible to minimize the possible points of failure and simplify verification.

The DRTM is the newer and more powerful concept. This concept requires the CPU, the chipset and the TPM to cooperate in an effort to bring a running system into a well defined state and from thereon create a Measured Launch Environment (MLE). In this case an Authenticated Code Module (ACM) which is supplied by the chipset manufacturer serves as the RTM.

The default policy for PC platforms is to measure the configuration of the platform and report it to a challenger on request. There is no restriction on the software that can be executed by the user, but it has become very difficult to hide which software is executed on the system, during an attestation. This is called *Authenticated Boot.* In case of mobile MTM equipped platforms the default policy is to abort the boot process if a non-satisfactory software component is found. This is called *Secure Boot*.

Mixed forms of Secure and Authenticated Boot are also possible. Intel's *Trusted eXecution Technology (TXT)* [Grawrock 2009] is a technology that enables a DRTM. TXT introduces *Launch Control Policies (LCPs)*. A LCP allows to set a specific target measurement value for the initial MLE. If the actual identity of the MLE does not match this preset value the boot process is terminated.

The platform configuration report (Quote) of a TPM is signed with an *Attestation Identity Key (AIK)*. Each TPM is supplied with a unique RSA key-pair, the *Endorsement Key (EK)*. Signing all Quotes with the same unique private key poses a serious privacy issue. In anticipation of this, the TCG standards prohibit signing a Quote with the EK. Instead an AIK has to be used. The AIK is an RSA key pair and is backed by an AIK certificate. The AIK certificate asserts that a Quote originates in a valid TPM.

TPMs are not shipped with AIKs, but are capable of creating new AIKs. In order for the AIK to be credible it must be certified. For a *Certificate Agency (CA)* to be able to issue AIK certificates without necessarily having physical access to the target platform, the TCG conceived three different credentials. The TPM EK certificate assures that the EK was correctly created and is protected inside a TPM. The platform certificate certifies that the TPM is correctly integrated into the host platform. Finally, a third certificate, the conformance certificate, establishes that the platform correctly adheres to the TCG standards.

In combination the EK, platform, and conformance certificates provide a powerful assurance that a platform is equipped with a correctly integrated TPM, and the system conforms to TCG standards. The TCG specified a cryptographic protocol for creating and certifying AIKs. The first part of the AIK protocol consists of creating a new RSA key under the protection of the TPM, collating the three certificates with the TPM signed public AIK, and then sending the whole package encrypted under the public key of the CA, to the CA. The CA verifies the certificates and decides if the request is valid. If the request is accepted, the CA returns the encrypted AIK certificate. To decrypt the certificate the private EK of the requesting TPM is necessary. The AIK protocol is explored in greater detail in [Pirker et al. 2009].

3.2 Electronic Identity - eID

Paper identity documents have long been used to prove the identity of individuals. Recently many countries have introduced electronic identities in the form of an ID-card which should be a smart card with cryptographic functions.

In the EU the basis for all national signature acts is the Directive for Electronic Signatures 1999/93/EC[European Parl. and the Council of the EU 1999].

All member states within the EU are enforced to adopt the directive into national law. The directive's purpose is to facilitate the use of electronic signatures and to contribute to their legal recognition. In connection with eID, electronic signatures are a key factor enabling strong and unambiguous authentication. In terms of the directive, so called advanced electronic signatures are required to be

1. uniquely linked to the signatory
2. capable of identifying the signatory
3. created using means that the signatory can maintain under his sole control
4. linked to the data to which it relates in such a manner that any subsequent change of the data is detectable.

An advanced electronic signature is created by a secure signature-creation device and is based on a qualified certificate. A qualified certificate is issued by a governmental registration office, after an evaluation process, and is subject to strict technical requirements. The signature created under these circumstances meets the legal requirements of a handwritten signature and therefore can be considered equivalent.

During the issuing process, when the qualified certificates are generated, the identity of the citizen must be proven with a paper based identity document and often is double checked with legal registers.

Through the two factors, knowledge and possession a certificate can assure the identification of an individual. Possession of a certificate is equivalent to controlling its private key. Key protection is usually ensured by storing it on a smart card, which also hosts the qualified certificate. Knowledge pertains to knowing the secret signing PIN. Only the legitimate user of a signature card must know the signing PIN.

4 The AAT Remote Release Protocol

The novelty of our protocol is to gather *trusted states* for an AATwithout requiring physical access to the target platform or platforms. During the process the user has to prove her identity and this also allows preparing the AAT with different privileges for different emergency response scenarios and for varying roles on different machines.

The data gathering process is automated by means of a specifically prepared setup software image. The setup software image establishes the measurements for the target platform software configuration, the identity of the user, and if the platform is indeed equipped with a TPM.

The platform configuration measurements can be gathered by the setup image because it uses a similar configuration to the actual emergency software. Thus, the setup software can establish a base pool of platform software configuration measurements from which the actual PCR values for the emergency software configuration can be extrapolated. This pragmatic approach is necessary, because even when using a DRTM, slight variations in the platform configuration complicate estimation of the exact PCR values.

To authenticate and authorize the applicant her identity has to be determined. Furthermore, person related application data must be evaluated. This can be handled using eID smart cards and their electronic signatures. By using this method of authentication and authorization the assurance level for the identity is backed by a country's specific national facilities for eID.

It must be established that the device of the user has indeed a TPM and that it is correctly integrated into the platform. One mitigation strategy is to let a qualified entity evaluate and certify the TPM and platform:

1. Some vendors fabricate hardware that comes with the EK certificate for its TPM, a platform certificate and a conformance certificate. Currently, to the best of our knowledge, only Infineon provides an EK certificate for it's TPMs.

2. A trusted qualified administrator with physical access to the device assesses and certifies it.

3. The user performs the evaluation and guarantees its fitness.

The ESIP can decide to refrain from demanding the platform certificate and the conformance certificate and accept lower levels of assurances.

4.1 The protocol in detail

[Figure 1] depicts the flow of the *AAT Remote Release Protocol*. The involved parties include the applicant (citizen), who must own an eID card and is responsible for the user interaction (middle column). The left hand side represents the hardware devices for which an AAT is requested. On the right side the ESIP (registration authority) responsible for the software image, AIK cycle, application examination and AAT-deployment can be found.

A prerequisite of the protocol is that the TPM ownership has to be taken before, or during the first three steps. The protocol description in detail is as follows:

1. The applicant downloads the software setup image from her registration authority.

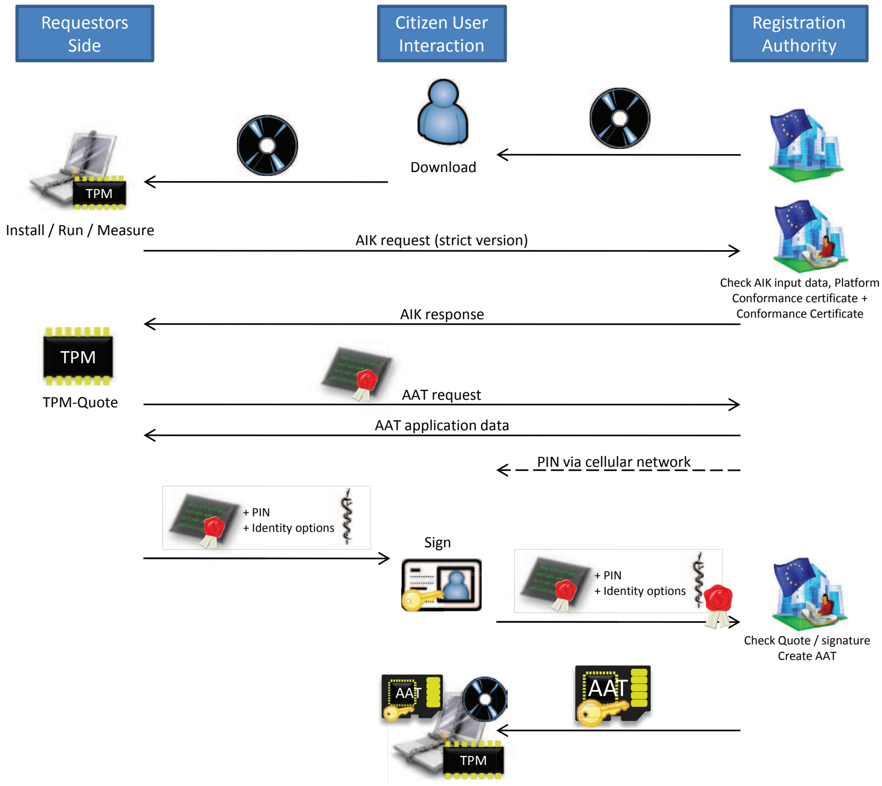

Figure 1: The AAT Remote Release Protocol

2. The applicant verifies the integrity and trustworthiness of the software image by checking its digital signature.

3. The applicant configures a DRTM launch of the image and starts it. These three steps can be handled by an additional installer.

4. The image boots into the base system, which performs an AIK certificate request. The request must contain EK, platform and conformance certificates.

5. The ESIP answers with the AIK certificate response. A nonce for the Quote in the next step is generated by the ESIP and part of the response.

6. The System is quoted to the ESIP together with applicant related application data.

7. The ESIP examines the quote and the application data. It sends an SMS to the user stating that the application has been confirmed. The SMS contains a PIN which serves as out-of-band confirmation.

8. The user signs the summary where she attests that everything with her platform and the application data for the privileges is O.K.. The signature encompasses the quote, the nonce, the PIN from the SMS and all additional data to be stored on, or which are a base for other credentials on the AAT.

9. The ESIP verifies authenticity and authorization of applicant.

10. The ESIP creates an AAT and sends the AAT via a postal service.

After reception of the AAT, the user is prepared to utilize it on the machines she requested and through the provision of credentials, protected by the AAT, she is empowered to help in an emergency response.

4.2 Security Analysis

The primary challenge of the protocol is to establish the identity of the applicant and the platform's conformance to TCG standards with an assurance level high enough to satisfy security requirements for disaster response.

Electronic ID cards provide a high assurance level for identification. The protocol must prevent replay and it must be established that the signature software is under the control of the user. Therefore, a trusted viewer is needed. This can be best guaranteed with a trusted platform. Revocation checking of signature certificates has to be done during the examination process and is an advantage of our protocol. Checking the validity during a crisis where no infrastructure is available is not practical. One solution is to perform these checks at a time when the IP-connectivity is reliable. Therefore, the protocol examines the identity and other data with the remote protocol before a disaster occurs.

The following challenges arise if the platform of the applicant has been subverted by an attacker:

1. Attacks on the setup image. Possible solutions are to either distribute the software by different means (postal service), or to perform the signature check on a trusted machine.

2. Attacks on the quote. Mitigated by TPM certification.

3. Attacks on AIK-cycle. Mitigated by TPM and platform certification.

Platform conformance verification is important for several reasons. For example, the platform could be faulty or the TPM/CPU/Chipset cooperation required for Trusted Computing could simply not work. The TCG originally proposed three certificates to certify the correct operation of the TC components for PC platforms: The EK, platform and conformance certificates. To the best of our knowledge, no PC platform manufacturer provides a platform certificate and the conformance certificate has been not been updated in the current credentials specification. Even with the assurance granted by these three certificates, there is no guarantee that it has not been maliciously tampered with. An attacker with hardware access could exchange the platform against one without a TPM and forward all TPM requests to a device in a *trusted state* which is equipped with a TPM. That is the TPM variant of the Grandmaster postal-chess problem. This is a hardware attack problem. It has to be considered that physical access to hardware is always a problem.

The user could be tricked into starting a corrupted image. An image could be changed and the signature verification could be circumvented, or a wrong image could be passed on by downloading from a phishing site. It is a challenge to distribute software securely. The long-existing problem where trusted connections between the communication parties (ESIP and requesting platform) are required has to be considered. We assume that the ESIP is trustworthy and not subverted. The ESIP provides only signed software for download which can be checked. The simplest solution for a save delivery would be an out-of-band channel (postal-service) that delivers a read-only medium to the user. Otherwise, the user must ensure herself that she can examine the downloaded image.

5 Future Work

A valuable extension of the protocol is the processing of additional identity elements during the application process. This enables rights-management and allows for collecting and storing additional privilege data to an AAT. These privileges can be per machine, or per AAT. There are different kind of situations, needing different kind of the alluded solutions. With an extended version of our protocol, a mapping from roles to different machines becomes possible. For example, it becomes possible that a user can identify herself as a doctor on one host platform, whereas on a differnt platform, she might only have the access rights of a general emergency response person. Such identity elements are already available in some european countries. They can be in the form of certificates or signed data structures. In Austria the *Federal Ministry of Health* issues health service provider tokens *HSP-Tokens*[Danner et al. 2008, Danner & Knall 2008] that will be used in future extensions of the protocol. HSP-Tokens contain roles and validity duration of them as well as other constraints and are very useful in emergency response.

One of the major obstacles remaining is the management of platform changes and platform software updates for already deployed AATs. Easy to use and secure solutions have to be found.

6 Conclusion

In conclusion, our proposed protocol promises a way to collate identity information for both individuals, and machines and their software configuration remotely, by means of Trusted Computing and eID. The aggregated information can then be used to grant access to emergency communication networks and software services. In conjunction with a hardware token like the AAT it becomes possible to enforce trust decisions even in ad-hoc scenarios. With our protocol it becomes possible to prepare AATs for future disaster response operations early enough and without the necessity of physical presence of the hardware and the individuals who's identities have to be proven. Through our protocol, the administrative expenses to prepare for a disaster response can be optimized. In contrast to conventional processes, it increases electronic quality of data and assures a high level of security through remote attestation and examined privilege authorization. These crucial points are absolutely indispensable in crisis scenarios.

Acknowledgments

The work reported in this paper was supported by the European Commision through project SECRICOM, FP-7, contract no. FP7-SEC-218123.

References

[Arunachalan et al. 2007] Arunachalan, B., Light, J., & Watson, I. (2007). Mobile agent based messaging mechanism for emergency medical data transmission over cellular networks. In *Communication Systems Software and Middleware, 2007. COMSWARE 2007. 2nd International Conference on* (pp. 1–6).

[Balfe & Gallery 2007] Balfe, S. & Gallery, E. (2007). Mobile agents and the deus ex machina. *21st International Conference Advanced Information Networking and Applications Workshops (AINAW '07)*, 2, 486–492.

[Danner & Knall 2008] Danner, P. & Knall, T. (2008). GDA-Token Berechtigungskonzept - Version 1.0.1.

[Danner et al. 2008] Danner, P., Rssler, T., & Knall, T. (2008). GDA-Token Spezifikation - Version 1.0.1.

[England & Loeser 2008] England, P. & Loeser, J. (2008). Para-virtualized TPM sharing. In *Proceedings of TRUST 2008*, LNCS: Springer Verlag. in print.

[England & Tariq 2009] England, P. & Tariq, T. (2009). Towards a programmable tpm. In *Trust '09: Proceedings of the 2nd International Conference on Trusted Computing* (pp. 1–13). Berlin, Heidelberg: Springer-Verlag.

[European Parl. and the Council of the EU 1999] European Parliament and the Council of the European Union. Directive 1999/93/EC of the European Parliament and of the Council of 13 December 1999 on a Community framework for electronic signatures, OJ L 13, 19.1.2000, p. 12.

[Ferg et al. 2007] Ferg, B., Fitzpatrick, B., Howells, C., & other (2007). OpenID Authentication 2.0. Final Specification.

[Franklin et al. 2005] Franklin, M., Mitcham, K., Smith, S., Stabiner, J., & Wild, O. (2005). Ca-in-a-box. In *Public Key Infrastructure Second European PKI Workshop: Research and Applications, EuroPKI 2005*.

[Gallery et al. 2009] Gallery, E., Nagarajan, A., & Varadharajan, V. (2009). A property-dependent agent transfer protocol. In *Trust '09: Proceedings of the 2nd International Conference on Trusted Computing* (pp. 240–263). Berlin, Heidelberg: Springer-Verlag.

[Grawrock 2009] Grawrock, D. (2009). *Dynamics of a Trusted Platform: A Building Block Approach*. Number ISBN 978-1934053171. Intel Press, Intel Corporation, 2111 NE 25th Avenue, JF3-330, Hillsboro, OR 97124-5961: Richard Bowles.

[Hein & Toegl 2009] Hein, D. M. & Toegl, R. (2009). An autonomous attestation token to secure mobile agents in disaster response. In *The First International ICST Conference on Security and Privacy in Mobile Information and Communication Systems, MobiSec 2009*: Springer-Verlag.

[Klenk et al. 2009] Klenk, A., Kinkelin, H., Eunicke, C., & Carle, G. (2009). Preventing identity theft with electronic identity cards and the trusted platform module. In *EUROSEC '09: Proceedings of the Second European Workshop on System Security* (pp. 44–51). New York, NY, USA: ACM.

[Maidl et al. 2008] Maidl, M., von Oheimb, D., Hartmann, P., & Robinson, R. (2008). Formal security analysis of electronic software distribution systems. In M. D. Harrison & M.-A. Sujan (Eds.), *Computer Safety, Reliability, and Security 27th International Conference, SAFECOMP 2008*, number 5219 in Lecture Notes in Computer Science: Springer-Verlag.

[Pirker et al. 2009] Pirker, M., Toegl, R., Hein, D., & Danner, P. (2009). A PrivacyCA for anonymity and trust. In L. Chen, C. J. Mitchell, & M. Andrew (Eds.), *Trust '09: Proceedings of the 2nd International Conference on Trusted Computing*, volume 5471 of *LNCS*: Springer Berlin / Heidelberg.

[Sandhu & Zhang 2005] Sandhu, R. & Zhang, X. (2005). Peer-to-peer access control architecture using trusted computing technology. In *SACMAT '05: Proceedings of the tenth ACM symposium on Access control models and technologies* (pp. 147–158). New York, NY, USA: ACM.

[Schurr et al. 2005] Schurr, N., Marecki, J., & Tambe, M. (2005). The future of disaster response: Humans working with multiagent teams using DEFACTO. In *AAAI Spring Symposium on AI Technologies for Homeland Security*.

[TCG 2003] TCG (2003). Trusted computing group website. http://www.trustedcomputinggroup.org/.

[Trusted Computing Group 2007] Trusted Computing Group (2007). TCG TPM specification version 1.2 revision 103.

[Trusted Computing Group 2008] Trusted Computing Group (2008). TCG mobile trusted module specification version 1.0 revision 6.

[Türpe et al. 2009] Türpe, S., Poller, A., Steffan, J., Stotz, J.-P., & Trukenmüller, J. (2009). Attacking the bitlocker boot process. In L. Chen, C. J. Mitchell, & A. Martin (Eds.), *Trusted Computing Second International Conference, Trust 2009*, volume 5471 of *Lecture Notes in Computer Science*: Springer-Verlag.

[Weigold et al. 2008] Weigold, T., Kamp, T., Hermann, R., Höring, F., Buhler, P., & Michael, B. (2008). The zurich trusted information channel - an efficient defence against man-in-the-middle and malicious software attacks. In P. Lipp, A.-R. Sadeghi, & K.-M. Koch (Eds.), *Trusted Computing - Challenge and Applications First International Conference on Trusted Computing and Trust in Information Technologies, TRUST 2008*, number 4968 in Lecture Notes in Computer Science: Springer-Verlag.

[Xu et al. 2007] Xu, G., Borcea, C., & Iftode, L. (2007). Trusted application-centric ad-hoc networks. In *Mobile Adhoc and Sensor Systems, 2007. MASS 2007. IEEE Internatonal Conference on* (pp. 1–10).