Trusted Infrastructures for Identities

Barbara Fichtinger, Eckehard Hermann
University of Applied Sciences Hagenberg
Softwarepark 11, 4232 Hagenberg, Austria
{Barbara.Fichtinger,Eckehard.Hermann}@fh-hagenberg.at

Nicolai Kuntze, Andreas U. Schmidt
Fraunhofer Institute for Secure Information Technology SIT
Rheinstrasse 75, 64295 Darmstadt, Germany
{Nicolai.Kuntze,Andreas.U.Schmidt}@sit.fraunhofer.de

August 30, 2007

Abstract

The establishment of trust relationships across multiple identifier domains in identity management architectures enables a service provider in a certain domain to trust the decisions of an identity provider located in a foreign identifier domain. As a result, users do not have to create new credentials for every identifier domain they communicate with. This trust relationship can be established by using the infrastructure provided by the Trusted Computing Group. In this paper, a concept for the establishment of this trust relationship based on trusted computing technology is developed.

1 Introduction

At the Virtual Goods Workshop 2006 we proposed a method to price ratings using trusted computing, based on pseudonymous tickets [1]. The advancement of this proposal – a generic architecture of trusted ticket systems – was presented at the International Information Security Conference 2007 [2]. In addition to the generic architecture of trusted ticket systems, the paper analyzes further applications such as content protection in message push services. In the context of this research projects, the establishment of trust relationships in identity management architectures has been identified as another application field for trusted computing technology.

Due to the recent shift of the business processes to the virtual world and the increasing importance of service-oriented architectures and virtual goods, the significance of identity management has been steadily increasing. There are different approaches for the realization of an identity management architecture which have one thing in common: they always deal with users who want to access a certain service in their particular identifier domains. In this context, an identifier domain is defined as the area of the network where the user can access services by using a certain identifier and the respective credentials. As the management of an individual pair of identifier and credentials for each service provider leads to a high management overhead, identity management architectures offer ways to access multiple services in a domain with the same identifier and credentials.

If the user aims to access a service in another identifier domain, the conventional solution would be the creation of a new identifier and credential pair for the domain. However, it would be more convenient to use the already established identifier and credential pair from the other domain. This means that the foreign identifier domain would have to trust the decisions made in the other domain. For the establishment of this trust relationship, several conventional approaches – such as cross certification, spanning certificate authorities or mirroring of user databases – already exist. But unfortunately, all of them lead to substantial technical overhead, the more identifier domains are involved. Another approach for the creation of trusted infrastructures for identities is the usage of already existing architectures, such as the infrastructure provided by the Trusted Computing Group based on its Trusted Platform Module.

The paper is organized as follows. Chapter 2 gives an introduction to identity management and the necessary protocols in this area. In chapter 3, the background information about trusted computing is summarized. Based on this fundamental information, the concept for the establishment of trusted infrastructures for identities is developed in chapter 4. A detailed analysis of the problem analyzed in this paper is presented in the identically named master thesis [3].

2 Identity Management

Digital identity management deals with the creation, usage, management and possible destruction of digital identities [4, pp. 8ff.]. A digital identity can be defined as a set of permanent or long-lasting attributes associated with a subject or an entity [5, p. 36].

Basically there are three different approaches for identity management architectures (IMA) which are described in [6]. Figure 1 gives an overview of the different systems. In the *isolated IMA*, each service provider acts as identifier and credential provider for the user. Even though this approach is rather simple, it is complicated for the user as it is necessary to maintain different identifiers and credentials for each requested service. The *federated IMA* creates an identifier domain (circle of trust) consisting of different service providers that accept the identifiers and entitlements issued by other service providers within the domain. Different agreements, standards and technologies are necessary in order to enable identity federation. In a *centralized IMA* there is only one identifier and credential provider in each identifier domain (circle of trust) which has to be used by all the different service providers in the domain.

In the context of identity management, there are several aspects which play important roles during the design of IMAs. *Trust* is one of those fundamental aspects of identity management. In general, trust can be defined as the willingness of a certain party to depend on another party in a particular situation even though negative consequences are possible

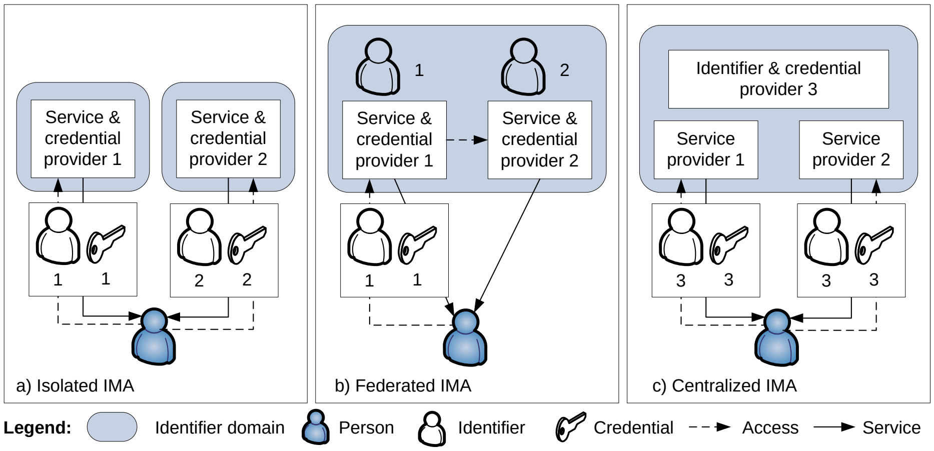


Figure 1: Identity management architectures. [6]

[7]. As analyzed in [4, pp. 15ff.], trust is always linked to a set of identity credentials and the associated attributes. In order to prove that an entity is authorized to use a claimed identity, credentials (passwords, digital certificates, biometrics etc.) should provide evidence. Trust relationships have the following properties:

- Trust is transitive only in very specific circumstances. If A trusts B and B trusts C, A may trust C.
- Trust is not symmetric. If A trusts B it does not follow that B trusts A.
- Trust cannot be shared. If A trusts B and A trusts C, it does not mean that B trusts C.
- Trustworthiness cannot be self-declared. Just because A says that the other parties should trust him does not mean that they have to.

Once a trust relationship is established, the participating entities form a trust or identifier domain. Federated and centralized IMAs try to create a certain trust or identifier domain in which the service providers trust each other or a central identity provider. In addition to the trust relationships inside these trust domains, it is also an issue to establish relationships between multiple domains.

Apart from the establishment of trust relationships, *access-control policies* which specify the entitlements of a certain subject for a particular resource have to be considered. Moreover, *privacy* aspects such as anonymity and pseudonymity of the user are important. Throughout the whole IMA, *authentication* and *authorization* of the user are other relevant aspects. *Integrity*, *non-repudiation* and *confidentiality* have to be especially assured during the transmission of protocol messages by the usage of suitable cryptographic algorithms.

For the transport of authentication and authorization credentials across identity and service providers in the same or different identifier domains, the XML-based Security Assertion Markup Language (SAML) offers the necessary functions [8, pp. 303ff.]. If an entity in an identifier domain requests resources in another domain, a trust relationship has to be established between the domains in order to accept the ticket in the foreign domain. Quite often, the identity and service providers in the identifier domains use different access-control systems. SAML addresses this compatibility problem by enabling a user to authenticate at the identity provider and accessing a service with the received ticket without having to authenticate at the service provider again. The whole SAML framework is based on *assertions* [9, p. 6]. These assertions are statements issued by SAML authorities about a subject that wants to access a resource. After a subject's identity has already been proven to the authentication authority, the *authentication assertion* contains the related information. Specific information describing the subject (credit limit, access level etc.) is stored in the *attribute assertion*. Details about the activities a subject is authorized to perform on a specific resource are communicated in the *authorization decision assertion*.

3 Trusted Computing

The trusted computing initiative has been established by the Trusted Computing Group (TCG) to develop a concept for attesting the trustworthiness of a platform [10, p. 9]. In order to guarantee this trustworthiness, there has to be a special component which forms the foundation of trust and validates the trustworthiness of all the other components. Realizing this foundation of trust with the help of a software component is not possible as it can easily be influenced. Therefore, the TCG has designed a hardware chip – the Trusted Platform Module (TPM) – that is mounted on the motherboard of a computer to form the foundation of trust.

A trusted platform has to implement several basic security features that are described in [11, pp. 5f.] and [12, pp. 31f.]. A *shielded location* is a protected storage area designed for sensitive information. These locations can only be managed by commands called *protected capabilities*. Mainly, these commands offer functions to control the *integrity measurement*, *storage* and *reporting functionality* of the TPM. Moreover, the trusted platform has to provide *attestation mechanisms* that allow an external party to verify the accuracy of a certain piece of information known to the TPM.

In order to realize these fundamental features, the TPM is equipped with basic software components (roots of trust) which have to be trusted because their malfunction cannot be measured and as a consequence might not be detected. The *Root of Trust for Measuring (RTM)* is a computing engine which offers functions to determine the integrity of a platform's configuration during the boot process. The calculations are performed by the Core Root of Trust for Measurement which is basically a BIOS extension that is executed first during the boot process. The TPM itself offers the necessary functions such as calculation of hash values and storage of the integrity values in the Platform Configuration Registers (PCR). Second, there is the *Root of Trust for Storage*, a computing engine which is responsible for the protection of keys and data objects. The whole key management is also situated in its field of duty. The reporting of data protected by the root of trust for storage is a task performed by the *Root of Trust for Reporting*. The necessary activity to perform trustworthy reporting tasks is called attestation. The *Trusted Building Blocks (TBB)* are those parts of the roots of trust that have neither shielded locations nor protected capabilities [11, pp. 6f.], which means that they have to be trusted by default. Examples for TBBs are the instructions for the RTM and TPM initialization functions.

To determine whether an application running on a certain platform can be trusted, a transitive trust relationship has to be established based on the hardware TPM with its TBBs and roots of trust [11, pp. 7f.]. If a trusted component of the platform measures the trustworthiness of another component, a transitive trust relationship is established between the first trustworthy component and the second. In other words, this means that the second component is accepted as trustworthy because the first component attests its trustworthiness. As a consequence, the trust boundary is extended and further on includes not only the first but also the second component [10, p. 9]. This process enables an external party to trust an application running on the platform due to the TPM and its roots of trust.

In addition to those basic integrity functions, the TPM can be used in any arbitrary application. Therefore, the TPM offers various functions such as generation of asymmetric and symmetric keys; calculation of signatures and hash values; asymmetric and symmetric encryption; encryption of cryptographic keys; creation of random numbers; and key management [13, p. 626]. In addition to these functions, the TPM manages several kinds of migratable or non-migratable keys. *Signing keys* are asymmetric keys which are used for signing arbitrary application data and messages. The *storage keys* are used for the encryption of data and other keys stored externally of the TPM. Moreover, there is the *endorsement key (EK)*, a non-migratable 2048 bit RSA key, which is created during chip manufacturing. As it is uniquely associated with a TPM, it can be used to verify its authenticity. In order to sign data originated by the TPM (e.g. TPM capabilities, PCR values), the *attestation identity keys (AIK)* (2048 bit RSA keys) have to be used.

Moreover, different certificate types (credentials) are required to use the keys in application scenarios. The *endorsement credential* issued by the TPM manufacturer attests that the creation and transmission of the EK to the TPM was performed according to the specification. Among other information, it includes the public part of the EK and the vendor certificate. Even though it contains only public information, it should be kept private as it uniquely identifies the TPM. The *attestation identity credential* attests and identifies the private part of the AIK. During the creation of the AIK, the TPM sends a request to a trusted third party, the privacy-CA, which includes the public part of the AIK as well as different credentials. The privacy-CA checks the credentials and issues the attestation identity credential which attests that the certain AIK belongs to a TPM which is conformable to the specification. As a TPM can create an arbitrary number of AIKs, it can use them as pseudonymous identities that cannot be linked by an external party except the privacy-CA.

4 Trusted Infrastructures for Identities

In traditional IMAs, a domain consists of multiple service providers and one or more identity providers depending on whether a federated or centralized approach is chosen. Tickets issued by an identity provider in a certain domain can normally be used to access services at service providers located in this domain. In order to use tickets across multiple identifier domains, trust relationships between the domains have to be established. Even though, traditional approaches – such as cross certification, spanning certificate authorities or the mirroring of user databases – lead to the establishment of mutual trust, the technical overhead is especially high, the more identifier domains are involved. Therefore, another approach is the usage of the architecture provided by the trusted computing technology. A user belonging to *identifier domain A* can use a ticket issued from *identity provider A* located in the same identifier domain to access a service at *service provider B* from the foreign *identifier domain B*. As a basic requirement, each identity provider has to be equipped with a TPM that forms the base for the establishment of trust. To decide whether the foreign ticket can be accepted, the following two requirements have to be fulfilled and verified by the ticket-receiving service provider:

1. At the moment of ticket issuing, the identity provider (in particular its configuration and system status) was trustworthy.

2. The identity provider is authorized to issue tickets for the identifier domain the service provider belongs to.

4.1 Trustworthy Status of the Identity Provider

A method to prove the trustworthy status of the identity provider is the integration of status information in the issued tickets. The TCG defines a process of providing reliable and authentic integrity data about the components of a trusted platform that can later be verified [10, pp. 26f.]. As a trusted platform consists not only of hardware components but also of software, integrity measurement has to include both. The foundation of the whole process are the measurement values. The *Runtime/Loadtime Measurements* describe the status of a running system. Loadtime measurements are taken during the boot-up sequence which means that they describe the pre-operating system state. On the contrary, runtime measurements generally refer to the integrity of the components during the operation of the platform. In addition to this general distinction, the loadtime measurements are taken by the RTM whereas an operating system part is responsible for the creation of runtime measurements. In order to verify the trustworthiness of a platform, there has to be a way to validate the runtime and loadtime measurements of a platform. The solution for this problem is the creation of *reference measurements* which describe the desired state of each component. Authoritative sources (e.g. the manufacturers of the specific component) are responsible for the creation of reference measurements. They collect and process (usually in XML format) the component integrity data and publish it in a reference manifest database. During the evaluation phase, the reference measurements are compared with the runtime and loadtime measurements.

Basically, the measurement value of a component is the SHA-1 hash value of the application and configuration data which is stored in a particular PCR [14, pp. 31ff.]. In total there have to be at least 16 PCRs available which are reserved for loadtime (PCR0-7) and runtime (PCR8-15) measurements. As there are more than 16 components demanding integrity measurement, the TCG specifies a special process to calculate the PCR values [10, pp. 26f.]. The old value of the PCR (initial value is zero) is concatenated with the new measurement value and hashed in order to get the new value for the PCR. Since the entries of the PCR change with every new measurement value, a mechanism to keep track of the old values of each component is required in order to evaluate the PCR values. These tasks are performed by the integrity measurement log which stores all the hash values together with additional information. During the attestation process, both, the values of the PCRs signed with the AIK and the integrity measurement log have to be delivered to the verifying platform.

4.2 Authorization of the Identity Provider and Establishment of Trust Relationships

The process of requesting AIK credentials from the privacy-CA is used to realize authorize the identity provider to issue trusted tickets (see figure 2). Based on the vendor certificates in the EK credentials or more specific on the public parts or serial numbers of the EK, the privacy-CA decides whether the identity provider is authorized to issue trusted tickets.

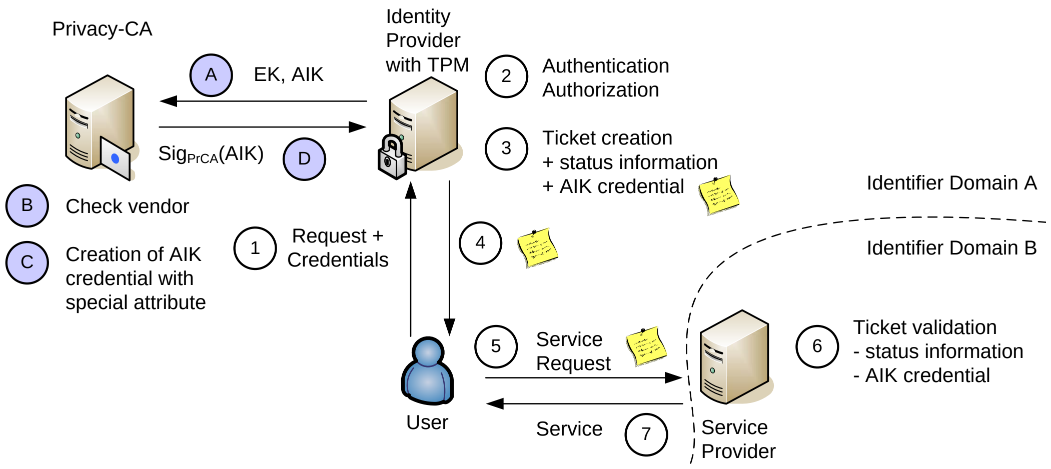


Figure 2: Trust establishment with AIK credentials and the privacy-CA.

As a basic requirement, each identity provider has to be equipped with a TPM that forms the base for the establishment of trust. Furthermore, the process of the establishment of the AIK credentials and the tasks of the privacy-CA have to be slightly adapted.

A. The identity provider sends a request consisting of EK credential and the public part of the AIK to the privacy-CA.

B. The privacy-CA checks the vendor certificate in the EK credential and decides whether TPMs issued by this vendor are allowed to issue trusted certificates.

C. In the case of a positive decision, the privacy-CA issues an AIK credential containing a special attribute attesting that the identity provider owning the certificate is authorized to issue trusted tickets.

D. Finally, the AIK credential is sent back to the identity provider and can be used during the identity management process.

In the identity management architecture, there are only minor adaptions necessary. The main task is the integration of authorization and status information in the created ticket.

1. User authentication consisting of request and presentation of the credentials at the identity provider.

2. The identity provider authenticates the user possibly with the help of an authentication server.

3. After the authentication, the identity provider is responsible for the authorization decision according to a predefined access-control policy. In order to embed the authorization information in the ticket, the AIK credential is used. Additionally, the status information describing the trustworthy status of the identity provider has to be included in the ticket.

4. The ticket is sent back to the user.

5. The user requests a service and includes the issued ticket.

6. During the validation of a ticket, the service provider can check whether the AIK credential in the ticket contains the necessary attribute and was issued by a trustworthy privacy-CA. Furthermore, the status of the identity provider during the issuing of the ticket has to be checked.

7. The service is either granted or refused based on the access-control policies of identifier domain B.

4.3 Protocol Messages

As the trusted computing infrastructure is used to establish trust relationships between multiple identifier domains, several protocol messages of the TCG specification have to be adapted and a syntax for the trusted ticket has to be defined. In the following, the protocol messages are described.

4.3.1 AIK Credential Request

The demand for the authorization to issue trusted tickets is included in the *Tcpa Identity Request*, the identity provider's request for AIK credentials at the privacy-CA. The regular ASN.1 format for this request message is specified in [15, p. 279]. The message starts with basic information such as version and used asymmetric and symmetric algorithms. In this basic, unencrypted information, the request for the authorization to issue trusted tickets is included as a boolean value. The remaining parts of the request (AIK, EK credential etc.) are cryptographically protected.

For the creation of this adapted request message, the identity provider's client application for the creation of AIK credentials is responsible. It has to assure that the boolean flag – indicating the request for an adapted AIK credential – is included in the *Tcpa Identity Request* whenever it is required. The reason for the integration of the flag in the unprotected part of the request is that it logically does not fit the content of the other parts of the request. Furthermore, the fact that the identity provider requests the authorization to issue trusted tickets is not sensitive. If an attacker alters the request message and removes or adds the request flag, no security problems arise for the identity provider because the attacker cannot benefit from this attack.

4.3.2 AIK Credential

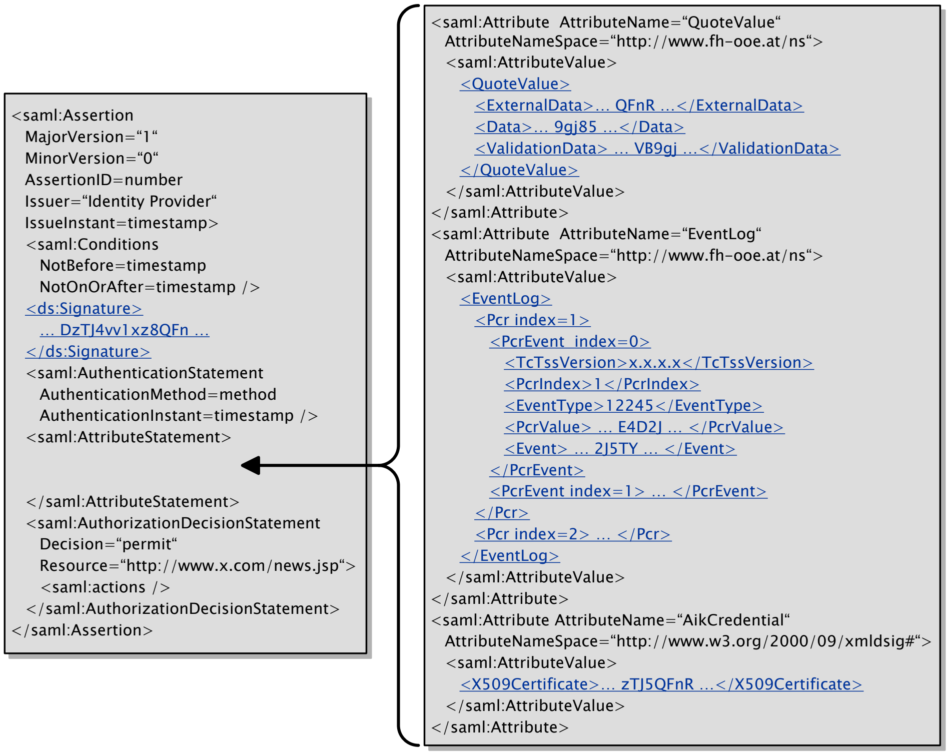
In the AIK credential issued by the privacy-CA, the authorization attribute for the identity provider, which allows the issuance of trusted tickets, has to be included. The TCG specifies the format for the credentials in [16] which is based on the message format of X.509 certificates and X.509 attribute certificates as specified in RFC 3280 [17] and RFC 3281 [18].

Even though the extended key usage field should not be used according to the TCG, this field will be used for the integration of the authorization information in the certificate. The extended key usage is defined by unique object identifiers assigned in accordance with IANA or ITU-T recommendation X.660. According to the object identifier repository [19], the key purpose with the object identifier 1.3.6.1.5.5.7.3 is already defined for the usage scenarios one to 16. The next free number in the tree is 1.3.6.1.5.5.7.3.17 which has to be newly registered for the issuing of trusted tickets (trustedTicketIssuing).

4.3.3 Trusted Ticket

The trusted ticket issued by the identity provider is specified as a SAML assertion including authentication, attribute and authorization decision statements (see figure 3). The authentication statement specifies details about the authentication process at the ticket-issuing web server and the authorization decision statement contains information about the actions, the user is allowed to perform at a certain service provider.

The necessary information which allows a ticket-receiving service provider to validate the ticket (authorization of the identity provider to issue trusted tickets and status information about the time of the ticket issuance) is included in the attribute statement. A signature



```
<saml:Assertion
  MajorVersion="1"
  MinorVersion="0"
  AssertionID=number
  Issuer="Identity Provider"
  IssueInstant=timestamp>
  <saml:Conditions
    NotBefore=timestamp
    NotOnOrAfter=timestamp />
  <ds:Signature>
    ... DzTJ4vv1xz8QFn ...
  </ds:Signature>
  <saml:AuthenticationStatement
    AuthenticationMethod=method
    AuthenticationInstant=timestamp />
  <saml:AttributeStatement>


  </saml:AttributeStatement>
  <saml:AuthorizationDecisionStatement
    Decision="permit"
    Resource="http://www.x.com/news.jsp">
    <saml:actions />
  </saml:AuthorizationDecisionStatement>
</saml:Assertion>
```

```
<saml:Attribute  AttributeName="QuoteValue"
  AttributeNameSpace="http://www.fh-ooe.at/ns">
  <saml:AttributeValue>
    <QuoteValue>
      <ExternalData>... QFnR ...</ExternalData>
      <Data>... 9gj85 ...</Data>
      <ValidationData> ... VB9gj ...</ValidationData>
    </QuoteValue>
  </saml:AttributeValue>
</saml:Attribute>
<saml:Attribute  AttributeName="EventLog"
  AttributeNameSpace="http://www.fh-ooe.at/ns">
  <saml:AttributeValue>
    <EventLog>
      <Pcr index=1>
        <PcrEvent  index=0>
          <TcTssVersion>x.x.x.x</TcTssVersion>
          <PcrIndex>1</PcrIndex>
          <EventType>12245</EventType>
          <PcrValue> ... E4D2J ... </PcrValue>
          <Event> ... 2J5TY ... </Event>
        </PcrEvent>
        <PcrEvent index=1> ... </PcrEvent>
      </Pcr>
      <Pcr index=2> ... </Pcr>
    </EventLog>
  </saml:AttributeValue>
</saml:Attribute>
<saml:Attribute AttributeName="AikCredential"
  AttributeNameSpace="http://www.w3.org/2000/09/xmldsig#">
  <saml:AttributeValue>
    <X509Certificate>... zTJ5QFnR ...</X509Certificate>
  </saml:AttributeValue>
</saml:Attribute>
```

Figure 3: Modified SAML assertion used as a trusted ticket.

with the private AIK over a hash value of the PCRs is stored in the QuoteValue field which internally consists of three different data blocks. In order to verify the PCR hash value included in the quoted data field and the current configuration of the platform, the integrity measurement log, which includes the measurement values as well as the PCR values, is transmitted to the ticket-receiving service provider in the EventLog field. The event log consists of multiple event values (PcrEvent) for each PCR (Pcr). For each event value, several information fields, such as TcTss version, PCR index, event type, PCR value and event value, have to be specified. Finally, the AikCredential field is used to store the AIK credential, which is required to validate the signature in the quoted value. For the encoding of the data in the XML elements, the base 64 encoding scheme according to RFC 4648 [20] is used.

In order to protect the integrity of the whole assertion, a digital signature according to the XML Digital Signature Standard [21] is required. A signing key created by the TPM and asserted by the private AIK is used to calculate the signature over the content of the assertion. The digital signature is stored in the field ds:Signature, which is already defined in the SAML specification, together with the public signing key.

4.4 Use Cases

In the identity management area, the purpose of this concept is the establishment of trust relationships between multiple trust domains. Without significant overhead, a service provider belonging to a certain identifier domain can trust the decisions of an identity provider in another domain. Above all, the service provider can be sure that the identity provider is trustworthy, which means that its applications and configurations are secure. Especially in situations where it is not possible to install overall protection mechanisms for the identity provider, the integration of status information allows a foreign service provider to estimate the status of the machine.

In addition to the primary purpose, the scenario can be adapted for the usage in other areas. An advantage of the scenario is that after the issuing of a ticket, it can be used anonymously for accessing a service at a certain provider. Theoretically, this concept can be combined with a payment system. Each ticket could have a particular monetary value or allow the access to a chargeable service. Unfortunately the problem arises that a ticket could be used multiple times at different service providers. In this case, non-repudiation has to be considered which means that, for example, the privacy-CA could take over the tasks of ticket devaluation. Especially in this scenario it is important to trust the authorization decisions of a foreign identity provider because money is involved. Moreover, the service provider has to perform accounting tasks with the identity provider that originally got the money from the user. In this case, the identity provider would be able to link the identity of the user with the accessed service at the service provider which could lead to privacy problems in case of a compromised identity provider. As a consequence, the authorization to issue tickets would have to be revoked.

5 Conclusion

In this paper, a concept for the establishment of trusted infrastructures for identities has been successfully presented. These trusted infrastructures enable the establishment of trust relationships across multiple identifier domains in the context of identity management by using the trusted computing architecture.

The major advantage of the proposed concept is the usage of the infrastructure provided by the TCG for the establishment of trust relationships between multiple identifier domains. This leads to a significant reduction of the initial implementation costs as no specific infrastructure has to be installed. In comparison to systems based on cross-certification, this concept does not require the installation of an additional PKI. In addition, it enables the embedding of status information about the machine in the tickets. As a consequence, not only information about the identity provider's authorization to issue trusted tickets is transmitted, but also detailed information about its current configuration.

However, the usage of the TCG infrastructure leads to problems concerning the scalability of the trust relationships between multiple identifier domains. It is the responsibility of the privacy-CAs to decide which identity provider is allowed to issue trusted tickets. A detailed concept which enables the grouping of trust relationships is not addressed in this paper. A possible solution would be the usage of different privacy-CA certificates for certain groups of trusting identity domains. Even though some of the required adaptions of the trusted computing architecture do not fully conform to the TCG specification (e.g. additional tasks for the privacy-CA, extension of the AIK credential), they are acceptable as they are only minor changes which can be realized without changing the global TCG specification (e.g. TPM architecture, EK certificate structure).

For future research work, the approach offers multiple starting points for extensions of the current architecture. One of these possibilities is an adaption of the protocol which allows the service provider to present its current platform status to the client if the client requests a service. This would allow the user to distinguish whether the server and the offered service can be trusted. In addition, the formulation of generic access-control policies focusing on multiple service providers offering similar services is another research topic. Depending on the application scenario, message replay attacks have to be kept in mind. Especially if clients can use their tickets in multiple identifier domains, it is a challenging exercise to implement protection mechanisms against this kind of attack. But above all, the implementation of integrity measurement mechanisms in current operating systems is one of the most important issues. As long as these mechanisms are not fully available, trusted platforms will not be able to create expressive event logs and as a consequence external parties will not be able to validate the status of a foreign platform.

As this paper was developed in the context of a master thesis at the University of Applied Sciences in Hagenberg, a more detailed analysis of the establishment of trust relationships across multiple identifier domains and the according reference implementation can be found in the full version of the thesis [3].

References

[1] Kuntze, N., Mähler, D., Schmidt, A.U.: Employing trusted computing for the forward pricing of pseudonyms in reputation systems. In: Axmedis 2006, Proceedings of the 2nd International Conference on Automated Production of Cross Media Content for Multi-Channel Distribution. (2006)

[2] Kuntze, N., Schmidt, A.U.: Trusted ticket systems and applications. In Venter, H., Eloff, M., Labuschagne, L., Eloff, J., von Solms, R., eds.: New Approaches for Security, Privacy and Trust in Complex Systems. Volume 232 of IFIP International Federation for Information Processing., Boston, Springer (2007) 49–60

[3] Fichtinger, B.: Trusted Infrastructures for Identities. Master's thesis, University of Applied Sciences Hagenberg, Hagenberg, Austria (2007)

[4] Windley, P.J.: Digital Identity. 1. edn. O'Reilly Media (2005)

[5] Camp, J.L.: Digital identity. Technology and Society Magazine, IEEE **23**(3) (2004) 34– 41

[6] Jøsang, A., Fabre, J., Hay, B., Dalziel, J., Pope, S.: Trust requirements in identity management. In: ACSW Frontiers ’05: Proceedings of the 2005 Australasian workshop on Grid computing and e-research, Darlinghurst, Australia, Australian Computer Society, Inc. (2005) 99–108

[7] McKnight, D.H., Chervany, N.L.: The Meanings of Trust. Trust in Cyber-Societies-LNAI **2246** (2001) 27–54

[8] Galbraith, B., Hankison, W., Hiotis, A., Janakiraman, M., Prasad, D.V., Trivedi, R., Whitney, D.: Professional Web Services Security. Wrox Press (2002)

[9] Hughes, J., Maler, E.: Security Assertion Markup Language (SAML) Version 1.1 Technical Overview, Committee Draft. Technical report, OASIS (2004)

[10] Trusted Computing Group, Infrastructure Working Group: Architecture Part II - Integrity Management, Version 1.0 (2006).

[11] Trusted Computing Group: TCG Specification - Architecture Overview, Version 1.2 (2006).

[12] Mitchell, C.: Trusted Computing. Institution of Engineering and Technology (2005)

[13] Eckert, C.: IT-Sicherheit. Konzepte, Verfahren, Protokolle. 3. edn. Oldenbourg (2004)

[14] Trusted Computing Group: TCG PC Client Specific Implementation Specification For Conventional BIOS, Version 1.20, Revision 1.00 (2006).

[15] Trusted Computing Group: Trusted Computing Platform Alliance (TCPA) Main Specification, Version 1.1b (2002).

[16] Trusted Computing Group: TCG Credential Profiles, Specification Version 1.0, Revision 0.981 (2006) .

[17] Housley, R., Polk, W., Ford, W., Solo, D.: Internet X.509 Public Key Infrastructure Certificate and Certificate Revocation List (CRL) Profile (RFC 3280). Internet Engineering Task Force, Network Working Group (2002)

[18] Farrell, S., Housley, R.: An Internet Attribute Certificate Profile for Authorization (RFC 3281). Internet Engineering Task Force, Network Working Group (2002)

[19] ASN.1 Consortium: ASN.1 Information site, Object identifier (OID) repository (2007) http://oid.elibel.tm.fr.

[20] Josefsson, S.: The Base16, Base32, and Base64 Data Encodings (RFC 3548). Internet Engineering Task Force, Network Working Group (2006)

[21] Bartel, M., Boyer, J., Fox, B., LaMiacchia, B., Simon, E.: XML-Signature Syntax and Processing W3C Recommendation. World Wide Web Consortium (2002).