Trust Management in Cross Domain Operations

Anders Fongen
Norwegian Defence Research Establishment (FFI)
PO Box 25, N-2025 Kjeller, Norway
Email: anders.fongen@ffi.no

***Abstract*—Protocols for communication across security domains need to be evaluated against their architectural properties, not only their security properties. The protocols have connectivity and capacity requirements, they have implications on system coupling, scalability and management. This paper investigates several trust management mechanisms from the perspective of a list of non-functional requirements. The conclusions have consequences for the organization of Identity Management Systems used in cross-domain applications.**

I. Introduction

During operations across borders of security domains there is a need to establish and manage trust between the communicating parties. The trust relates to several aspects of the operation, but in this paper the focus will be held on

- trust in the conduct of the operators
- trust in the integrity of the executing software

Other aspects, like trust in the genuineness and integrity of the data exchanged between the parties, may be relevant but will not be discussed here.

Besides the aspects of trust, there are seeral non-functional requirements which should be observed when implementing a domain interconnection, like:

1) The domains require separate subject registries and certificate providers.
2) Subject registries and certificate providers shall be a common resource for many application for. the support of both service-centric and object-centric authentication.
3) Registration of foreign subjects shall not be necessary.
4) The subjects own their credentials and shall decide their dissemination.
5) Least possible privileges are granted between the domains.
6) The connectivity between the domains shall be limited and controlled.
7) The traffic volume across the interconnection point shall be minimized for scalability reasons.

This list of non-functional requirements will hereafter be referred to as NFR#1 to NFR#7.

Although these requirements are not always mandatory, they are based on sound and reasonable principles of loose coupling, scalability, manageability and autonomy and are similar to well established guidelines of system construction in general.

Another principle more directly related to security operations is that the requestor of an operation has the "burden of proof", i.e., the requestor should provide all necessary proofs of authenticity and authorization for the operation to be executed. This principle is founded in Rivest's seminal paper [14], but it will be shown that the principle conflicts with the scalability and optimization requirements, and must be balanced according to this dilemma.

Finally, the principle of *symmetric* control of authenticity and authorization forms a basis for the proposed mechanisms. Trust between a requestor (client) and a responder (service) needs to mutually established. With the threats from man-in-the-middle attacks, phishing attacks and fraudulent services this principle should be self-evident, although most well-established Single Sign On protocols disregard it.

The **contribution** of this paper is an architectural model for inter-domain operations and a set of communication protocols and data structures which provides a good balance between security, scalability and management cost. The model is based on several years of research into cross-domain identity management, service discovery, messaging and service invocation in tactical networks where communication resources are scarce, cf. Section V

The remainder of the paper is organized as follows: Section II will give a detailed discussion on the structure of credentials used in cross domain authentication. Section III will follow with an analysis on how tokens of authorization may be shared and interpreted across domain borders. Section IV will give a short discussion on how credentials can provide attestation of software integrity. The prototype used for experimentation will be described in Section V. Some related work are shortly described in Section VI. Finally, the paper presents its conclusions in Section VII.

II. Authentication mechanisms

On a small scale, users can be authenticated with a password. This arrangement is responsible for our daily nuisance of an increasing number of passwords that need to be maintained. A central registry used by a Single Sign On (SSO) service improves the situation, but still only offers client authentication. HTTPS connections offer a limited authentication of services which relies on the user observing the service URL with due diligence.

A mutual authentication operation can be based on operations with public keys and certificates issued by a trusted third party called the *Identity Provider* (IdP). In its simplest form, a digital signature with ample replay protection or a challenge-response mechanism involving the private key of both parties is sufficient. Authentication relies on the *proof of possession*

Dette er en postprint-versjon / This is a postprint version.
DOI til publisert versjon / DOI to published version: 10.1109/MILCOM.2015.7357565

principle, in which only a subject who possesses the private key can perform the demonstrated operation.

In the context of this paper, a *domain* is constituted by an IdP and all subjects (called *domain members*) to which it issues certificates. A public key certificate binds a public key to a subject identity for a limited time with the digital signature of an IdP. The public key of the IdP is known and trusted within the domain (called a *trust anchor*) and the certificate can therefore be *validated* by anyone. *Verification* of a digital signature therefore involves certificate validation as well as verifying the signature value and the message hash value.

Formally, a public key certificate of subject $x, x \in a$ can be written as $(Id_x)_a$, indicating that it is issued and signed by a IdP of domain a. A public key certificate in encrypted form is written as $(Id_x)_a^x$ indicating that it is encrypted with the public key of subject x. The certificate of IdP a is self signed and may be written as $(Id_a)_a$.

In a cross-domain environment, i.e., operations between members of different domains, a certificate $(Id_y)_b$ issued to $y, y \in b$ cannot be directly validated by members of a since they have a different trust anchor. Three solutions exist to solve this problem:

1) an often proposed solution is to issue a *cross certificate* from a to b, expressed as $(Id_b)_a$ so that subjects with $(Id_a)_a$ as their trust anchor can use the certificate pair $[(Id_y)_b, (Id_b)_a]$ to construct a *signature path* from y to a.
2) all members of domain a can add $(Id_b)_b$ to their trust anchor collection
3) IdP for a can issue a *guest certificate* $(Id_y)_a$ to certify the public key of y for members of domain a

Solution no. 1 and 2 violates NFR#5, since they mandate members of b to validate any certificate issued in a, i.e., creates infinite and unconditional trust from a to b. Solution no. 3 allows the IdP to decide which individual members of b should have new certificates issued. This decision should be based on rules, since a separate registry of approved subjects would violate NRF#3. Solution no. 2 is also disregarded for the remainder of the paper from reasons of scalability.

A. Certificate revocation status

It is common practice to issue certificates with relatively long lifetime, e.g., 1 year, despite the fact that they may be deemed invalid (*revoked*) before their expiration. A compromised private key is the most prominent reason for revocation. To ensure that revoked certificates are taken out of circulation the validating subject is required to obtain the revocation status in a separate operation. Revocation information can be published as a revocation list, which blacklists all revoked certificates and is widely disseminated on a regular basis, or through an online service called *validation authority* (VA) which issues *Proof of Validity* (PoV), signed by a trust anchor and given limited lifetime. The lifetimes of both these data structures are termed the *revocation latency* which constitutes the acceptable timeframe for a revoked certificate to stay in operation. There is a direct trade-off between revocation latency and network traffic volume.

The distribution of revocation data in a single domain operation generates considerable headache [3], [15], [2], and even the principle of burdening the validating subject with the duty of obtaining auxiliary information with the possible dilemma if the information is unreachable has been intensely criticized [13]. Cross-domain operation exacerbates the problems with regards to the scalability and connectivity, since potentially large revocation lists may need to pass the interconnection point, and a VA arrangement requires that extremely sensitive services are made reachable for an unknown and untrusted foreign population of clients.

Figure 1 contains an interaction diagram for a validation operation based on revocation lists. They are fetched on an on-demand basis and cached for the remainder of their lifetimes, so they are not retrieved for each operation. The caching mechanism and their size make revocation lists unfit for inclusion in the signature, so they are retrieved on demand by the validating client (possibly through a content delivery network).

Certificates are sent from the IdP on demand. The request contains the *distinguished name* of subject x (shown as dn_x in Figures 1-4), and the response is encrypted with the public key of the certificate in order to make the response useful only to the owner of the certificate, and to maintain NFR#4. An interaction diagram is a convenient way to identify the inter-domain traffic (across the dashed vertical line) and the required connectivity (the services which need to be reachable from the foreign domain).

PoV items, on the other hand, are suitable candidates for inclusion in signature structures. Since they relate to a single certificate and are significantly smaller than revocation list they can be proactively retrieved by the signer and sent together with the certificate in each signed message. A deeper analysis conducted in [5] shows that this consumes less bandwidth than an on-demand retrieval and caching mechanism on the validator's side.

Figure 2 shows the interaction diagram of this process. The PoV item which attests $(Id_x)_a$ is denoted $(PoV_x)_a$ and is fetched from the VA as often as necessary, which is less often than the chosen revocation latency. The important difference from Figure 1 is that the interconnection point is only used by traffic between x (client) and y (service), so the VA or IdP is never exposed to requests from other domains, cf. NFR#6 and NFR#7.

The same arrangement needs cross certificates on both sides, $(Id_a)_b$ and vice versa, in order to validate the certificates and PoV items. It is therefore still in violation of NFR#5 similarly to other cases which employ cross certificates.

B. Revocation free certificates

A radical improvement both in terms of architectural neatness and traffic properties is obtained through the use of certificates that make do without revocation arrangement. They are simply re-issued by the IdP for each revocation

Dette er en postprint-versjon / This is a postprint version.
DOI til publisert versjon / DOI to published version: 10.1109/MILCOM.2015.7357565

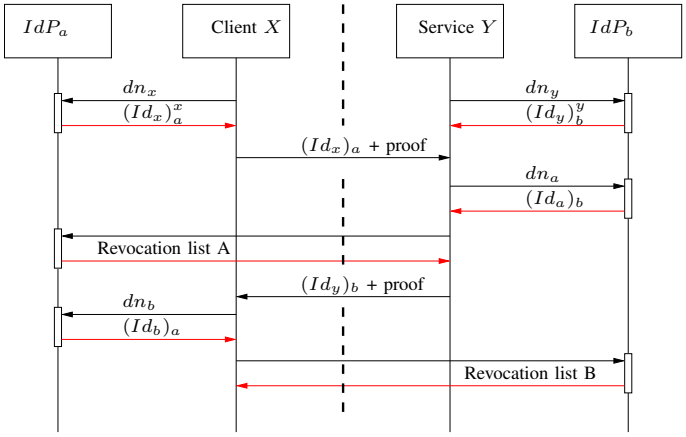

Fig. 1. The exchange of certificates and revocation lists during a cross-domain authentication using traditional PKI protocol elements. Red arrows indicate data items which are suited for a caching arrangement.

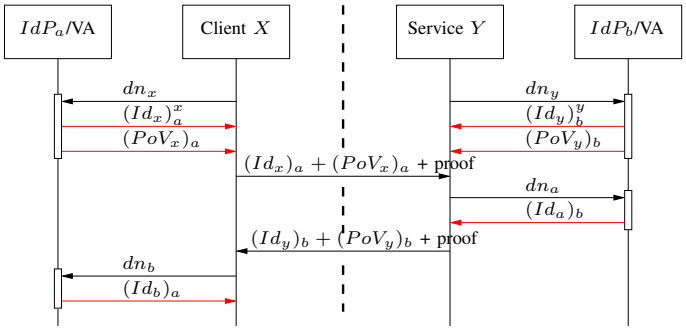

Fig. 2. The exchange of certificates and prefetched *Proofs of Validity* during a cross-domain authentication limits the traffic and connectivity across the inter-domain connection point (vertical dashed line). Red arrows indicate data items which are suited for a caching arrangement.

latency interval. A certificate can thus be validated without any auxiliary information source (of revocation status) and the burden of proof lies with the signer which has to obtain new certificates from the IdP as the old ones expire.

In a cross-domain operation, the validation of revocation free certificates requires either the presence of cross certificates or guest certificates, as outlined in the beginning of this section. Cross certificates are effective, since one is enough to validate every certificate from a given domain, but are in violation of NFR#5. Guest certificates create a different situation, since the issuing IdP can make individual policy decision at its discretion. Guest certificates can be requested by the signing client (a request which has to cross the domain interconnection point, contradicting NFR#6 and NFR#7) and used in the authentication operation, or they can be obtained (on demand and cached) on the validator's end, which shifts the burden of proof away from the signer. There is consequently a trade-off between the burden of proof principle on one hand and the principles of low traffic and limited connectivity across the domain interconnection point on the other. This trade-off will be further investigated in the course of this paper.

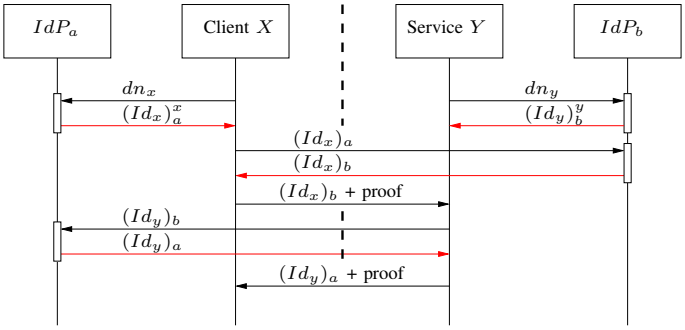

Fig. 3. The exchange of revocation free credentials and guest credentials during a cross-domain authentication operation. This model observes the "burden of proof" principle, but generates much traffic and requires extensive connectivity across the inter-domain connection point (vertical dashed line).

The network economy associated with revocation free certificates have been thoroughly calculated in [7], where the advantage of this scheme has been proved for single-domain operations. The rest of the paper will therefore study the applications of revocation free certificates both for authentication, authorization control and integrity control.

C. Cross or Guest Certificates?

Of the three listed solutions on cross-domain validation given in the beginning of Section II, solution no. 1 and 3 will be studied in more details with regard to their impact on inter-domain connectivity and traffic (which need to be minimized according to NFR#6 and NFR#7). It will be shown that the "burden of proof" principle, i.e., that the signer/requestor needs to obtain all necessary proof for the authentication operation, affects networking load in a negative way, and that improved scalability and separation between domains can be obtained if that principle is relaxed.

Figure 3 shows how guest certificates may be used for cross-domain validation between client $x, x \in a$ and server $y, y \in b$ when the burden of proof principle is observed. The guest certificates are denoted $(Id_x)_b$ and $(Id_y)_a$, respectively, and are obtained by x and y through invocations across the inter-domain connection point. Two observations may be made from the interaction diagram: (1) The IdP service need to be made reachable to clients outside the domain, and (2) that guest certificates are used (validated) only in the domain where they were issued, and are sent back and forth between the domains with no added value.

An alternative way is shown in Figure 4, where the validator (either x or y) requests the guest certificate from the IdP on behalf of its counterpart. Although it shifts the burden of proof over to the validator's side, it reduces the network traffic across the interconnection point and shields the IdP from foreign access. The figure also shows that the guest certificate $(Id_x)_b$ is returned to "its owner" during the authentication operation. This is done from scalability reasons; the guest certificate will be cached there and subsequently used in operations into the same domain. A remaining advantage of the protocol in Figure 3 is that guest certificates can be proactively issued while the

Dette er en postprint-versjon / This is a postprint version.
DOI til publisert versjon / DOI to published version: 10.1109/MILCOM.2015.7357565

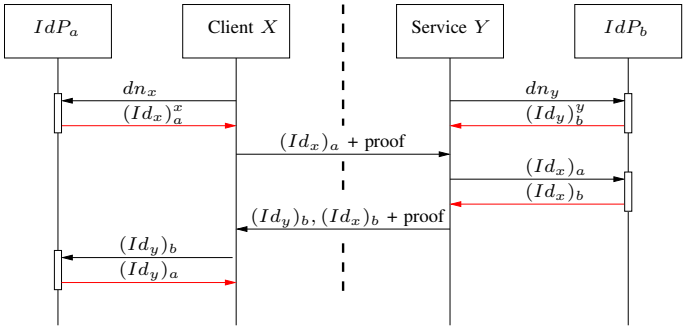

Fig. 4. The exchange of revocation free certificates during a cross-domain authentication operation. Guest certificates are never exchanged, but obtained by the peer as a part of the validation process. This model violates the "burden of proof" principle, but generates less traffic and requires minimum connectivity across the inter-domain connection point (vertical dashed line).

IdPs are reachable, so that the authentication can take place even without a communication path to the IdPs. This may offer an advantage in tactical mobile networks.

In these two figures, cross certificates, $(Id_a)_b$ and $(Id_b)_a$, could replace guest certificates and support cross-domain validation, although in violation of NFR#5. Cross certificates offer slightly better scalability properties, since they may be cached on the validator's side and used in subsequent validation operations from any requestor in that domain.

D. IdP service interface

The IdP issues certificates that certify the public key of subjects, and the public key of other domains to which it has a *trust relationship*. For this purposes the IdP should have access to a registry of approved subjects and trusted domains. In the system prototype made for the study of the ideas in this paper [6], a traditional PKI (Public Key Infrastructure) has been established where key generation and identity management take place. The IdP takes the public key and identifier from the PKI-based certificate, adds a short expiration time (similar to the *revocation latency* discussed in Section II-B) and signs the structure with its own private key. In Section III the addition of authorization attributes into the certificate by the IdP will be discussed.

Formally, the IdP may be denoted as a function:

$$\begin{aligned} IdP_a(dn_x) &= (Id_x)_a^x \quad \text{if } x \in a \text{ (domain member)} \\ IdP_a((Id_x)_b) &= (Id_x)_a \quad \text{if } b \in a \text{ (trust relationship)} \\ IdP_a(dn_b) &= (Id_b)_a \quad \text{if } b \in a \text{ (do.)} \end{aligned}$$

Observe that the $x \in a$ (subject x in domain a) expression has a different semantics than $b \in a$ (domain b trusted by domain a) although their notations are similar.

The reason why the prototype chose to employ a PKI for management of keys and identities is investment protection: if a PKI is already in operation, it represents a significant investment in development and management which should be employed for as many applications as possible.

III. Authorization control

An authenticated part is not necessarily authorized to conduct a given operation. Authentication is merely a first step to identify the authorization of a subject. In the same manner as authentication, authorization should be a symmetric property: The service will decide if a request from a client can be executed, and the client will decide if the response can be accepted (or invoked in the first place). In a messaging system, the sender will decide if the addressee may receive the message, and the receiver will decide if the message is accepted. However, even modern access control standards, like XACML, do not offer mutual authorization control, and require two independent systems to be working back-to-back for mutual control.

Several models for access control has been proposed and implemented. This paper will focus on the Attribute Based Access Control model (ABAC), which is constituted by a set of subject attributes and a boolean function over these attribute values (and possibly environmental properties like location and time etc.). The ABAC model lends itself well to integration into an identity management system, since attributes may be added to public key certificates and share the same trust properties as the public key. Most of the discussion on trust relations, guest- and cross certificates etc. given in Section II applies to a cross-domain ABAC mechanism as well.

The introduction of attributes into certificates raises concern over their interpretation in other domains. Here are some situations to avoid:

- Different IdPs use different attribute names to describe equivalent properties.
- IdPs use different attribute values to describe comparable properties like rank, degree, clearance level etc.
- IdPs use same attribute names in a conflicting manner.

These situations all have the potential to cause false positives or false negatives in the authorization control, with the loss of availability or confidentiality as a result.

It seems clear that attributes need harmonization, replacements and conversions when they are passed between domains. Alternatively, the set of boolean control functions can be extended to accommodate differences in attribute names, syntax and semantic content.

Another aspect of subject attributes is *confidentiality*. Attributes represent subject authorization which may well be of sensitive nature, and the limited trust between domains could mandate restrictions on which attributes may be disclosed to other domains.

The limited trust between domains, see NFR#5, suggests that domains will:

- be reluctant to trust unconditionally the attributes assigned by a foreign IdP
- avoid including non-essential attributes in the certificates which are to be used in a foreign domain

These conditions suggest that guest certificates are used, rather than cross certificates, since the former allows an IdP to inspect local certificates for suspicious attributes and modify them

Dette er en postprint-versjon / This is a postprint version.
DOI til publisert versjon / DOI to published version: 10.1109/MILCOM.2015.7357565

according to local policy. For the latter, the IdP must be able to issue certificates with varying attribute sets, depending on if they are intended for use in the local or in a foreign domain.

Although attributes can be organized in several different manners to accommodate these particular requirements, the following paragraphs will describe the chosen implementation in the prototype system:

Three types of attributes are considered in a cross-domain ABAC model:

1) Attributes local to the owner's home domain. These attributes should never be taken into regard in other domains.
2) Attributes common to all domains. These attributes need a globally harmonized vocabulary and semantic interpretation.
3) Attributes specific to one domain. These attributes are assigned during guest certificate issue.

These three categories of attributes are termed la, ca and ga, respectively, and they are separated by namespace prefixes. When attributes are evaluated, ga takes precedence over ca, which again takes precedence over la. Only ga is modified during guest certificate issuance. Since one guest certificate is used in only one domain, ga will never contain attributes specific to more than one domain.

For a formal definition of attribute processing the IdPs, their interface definition given in Section II-D must be augmented to show the attributes of the certificate.

A certificate for subject x issued in domain a which contains all three attribute types is denoted $(Id_x\{la, ca, ga\})_a$. Three different certificate variants can now be issued by an IdP:

$$\begin{aligned} IdP_a(dn_x, true) &= (Id_x\{la, ca\})^x_a && \text{if } x \in a \\ IdP_a(dn_x, false) &= (Id_x\{ca\})^x_a && \text{if } x \in a \\ IdP_a((Id_x\{ca\})_b) &= (Id_x\{ca, ga\})_a && \text{if } b \in a \end{aligned}$$

Having separate certificates for local and remote operations complicates caching and discovery arrangements somewhat, and an alternative single call to an IdP which returns both $[(Id_x\{la, ca\})^x_a, (Id_x\{ca\})^x_a]$ may reduce the number of IdP invocations, but this has not been tested in the prototype.

When an IdP is issuing a guest certificate, it will study the attributes of the given certificates. If it contains any ga attributes, the operation will be dismissed as this is already a guest certificate, or it is a forgery of some kind. Any la attributes are removed. The block of ca attributes should be inspected for suspicious or inconsistent content, and add ga attributes to adapt the attribute set to the requirements of the local ABAC processes. National classification and clearance values are typical candidates for adaption.

A. Authorization control in messaging systems

Although the figures suggest that the operations take place in a service invocation environment, the models presented in Sections II and III applies equally well to a messaging environment. In a messaging system, the sender and the receiver may have mutual authorization requirements that they both must fulfill in order for the message to be delivered. In this way, the sender may protect the confidentiality of a message, while the receiver protects the integrity, in the sense that only messages from authorized senders will be accepted. The requirements, in the form of access rules, may be separately given to a message router or attached to the message itself. The actual ABAC operations will take place in the message router on behalf of the communicating parties.

IV. Integrity control

As mentioned in the introduction, trust also relies on the integrity of software and hardware involved in the operations. So far, the discussion has taken into regard the authenticity and authorizations of the subjects involved in a transaction, based on the assumption that authorizations are given to subjects based on their clearance, training, duties etc. The conduct of a transaction may still be threatened by compromised software, e.g. malware, and even faulty hardware.

To meet the need for integrity control, a certificate can provide attestation that its owner operates from a "trusted" computer, e.g., one that has been checked for software integrity at boot time. The TPM (Trusted Platform Module) can offer a trusted boot, and also make certain cryptographic operations available only on the condition that the associated integrity check is passed. The TPM can generate a temporary key pair which will be attested by the IdP and used in subsequent operations (in lieu of the "personal" key pair). The trust chain for the integrity proof relies on: (1) That the temporary key pair is created only subject to a successful trusted boot, (2) that the computer possesses a key pair to prove that it is a part the domain's inventory and (3) that the TPM possesses a key pair issued by a trusted certificate issuer (e.g., Verisign) to prove that it is genuine.

This arrangement has been described in detail in [12]. The effect is an elevated trust in the authentication operation which may be shared across domain borders and affect the authorization control.

V. Prototype implementation

The principles proposed in this paper have been developed over several years of experimentation on a prototype identity management system called Gismo IdM. Gismo IdM has provided security services for experimentation on cross domain sevice invocations[8], publish subscribe distribution[11], service discovery[10], tuplespace coordination[9] and integrity protection[12]. All the services use the same set of base protocols for authentication and (where needed) authorization control. Gismo IdM uses revocation free, short lived credentials which include public key and subject attributes, and are used for authentication as well as authorization control.

Gismo IdM is open source, written in Java and uses serialized objects in the communication protocols. It runs on any JavaSE enabled platform including Raspberry Pi and has been ported to run on Android.

Dette er en postprint-versjon / This is a postprint version.
DOI til publisert versjon / DOI to published version: 10.1109/MILCOM.2015.7357565

VI. Related Work

Several protocols exist for *federated* authentication and access control, which implies that one IdP and the issued credentials can be used for many services within the domain of that IdP. Of more interest to the topic of this paper is how cross-federations are supported with these protocols.

Kerberos (RFC4120) is a popular protocol for federated authentication and is based on symmetric cryptography. An Authentication Server (AS) provides the authenticated client with proof which can be shown to a Ticket Granting Server (TGS) which returns to the client a ticket for a specific service. The TGS may do this subject to authorization control, given that the TGS has a user authorization database. The TGS must provide tickets for every service, no ”universal” ticket exists liken to a certificate.

The TGS can provide a ticket to another TGS in a different domain, provided that there exists a trust relationship between them in the form of a shared key. The foreign TGS can provide tickets for services in the foreign domain, but authorization control is impossible without copying the user authorization database between the domains. In the model described in this paper, no parallel to the TGS exist, the AS service provides the client with all necessary credentials for subsequent operations, also operations in other domains. The scalability of this model is therefore likely to be better than Kerberos. Authorization control in Kerberos, if any, takes place in a TGS, while the proposed model allows any service to implement their own control based on the supplied subject attributes. Kerberos provides a rudimentary form for mutual authentication, but an optional authorization control only applies for clients. Kerberos is a service oriented protocols and cannot be used in a messaging system.

SAML SSO profile [4] provides federated authentication of browser based clients. The protocol authenticates clients, but not services. The protocol relies on a central IdP which issues identity and authorization proofs in the form of *SAML Assertions*, which can be validated by services. SAML Assertions is functionally similar to certificates as it contains necessary information for both authentication and authorization control. Like Kerberos, SAML SSO requires the client to invoke IdP operations for every service it wants to contact, thus creating more network transactions than the proposed model and relying more on the reachability of the IdP.

XACML [1] is a framework for ABAC authorization control. It consists partly of a syntax for expressing access rules (to be evaluated over an attribute set), and partly an architecture for delegating access decisions to centralized servers. For this reason, the access rule syntax has elements which refer to resources and contexts, elements to resolve rule conflicts and elements which express conditions that the client must fulfill prior to access. XACML is service oriented in its nature and does not lend itself well to messaging systems. In the proposed model, where the services make authorization control themselves, the rule expression syntax may seem overspecified, but would otherwise be possible to use.

VII. Conclusions

The non-functional properties of cross domain trust mechanisms has been the topic of this paper. Different scenarios for the exchange of temporary credentials between domains have been described and compared for their connectivity and scalability properties. In a military tactical network one should also regard at which phase of an operation the network resources are available.

The conclusions from the analysis in this papers are:

- Cross domain trust should rely on revocation free credentials
- Public key and subject attributes should be included in the same certificate.
- Guest certificates are preferred over cross certificates
- The “burden-of-proof” rule should be relaxed for the purpose of reducing traffic volume and connectivity requirements across the intra-domain connection point.

References

[1] OASIS eXtensible Access Control Markup Language. https://www.oasis-open.org/committees/tc_home.php?wg_abbrev=xacml. Online, Accessed March 2015.

[2] Andre Aarnes, Mike Just, Svein J. Knapskog, Steve Lloyd, and Henk Meijer. Selecting revocation solutions for PKI. In *Proceedings of NORDSEC 2000 Fifth Nordic Workshop on Secure IT Systems*, Reykjavik, Iceland, 2000.

[3] David A. Cooper. A model of certificate revocation. In *Proceedings of the 15th Annual Computer Security Conference*, December 1999.

[4] John Hughes et. al. *Profiles for the OASIS Security Assertion Markup Language (SAML) V2.0*. OASIS Standard, 2005.

[5] Anders Fongen. Scalability analysis of selected certificate validation scenarios. In *IEEE MILCOM*, San Jose, CA, USA, Oct 2010.

[6] Anders Fongen. Federated identity management in a tactical multidomain network. *Int. Journal on Advances in Systems and Measurements*, Vol.4, no 3&4, 2011.

[7] Anders Fongen. Optimization of a public key infrastructure. In *IEEE MILCOM*, Baltimore, MD, USA, Nov 2011.

[8] Anders Fongen. Protected and controlled communication between military and civilian networks. In *Military Communication and Information Systems Conference (MCC)*, Gdansk, Poland, 2012.

[9] Anders Fongen. Data-centric authorization and integrity control in a linda tuplespace. In *ACM Symposium on Applied Computing (SAC'15)*, Salamanca, Spain, 2015.

[10] Anders Fongen and Trude Hafsøe Bloebaum. Trusted service discovery through identity management. In *IEEE MILCOM*, San Diego, USA, 2013.

[11] Anders Fongen and Federico Mancini. Identity management and integrity protection in publish-subscribe systems. In *IFIP IdMan 2013*, London, UK, 2013.

[12] Anders Fongen and Federico Mancini. The integration of trusted platform modules into a tactical identity management system. In *IEEE MILCOM*, San Diego, USA, 2013.

[13] Peter Gutmann. PKI: It's not dead, just resting. *Computer*, 35(8):41–49, 2002.

[14] Ronald L. Rivest. Can we eliminate certificate revocations lists? In *Financial Cryptography*, pages 178–183, 1998.

[15] Adam J. Slagell, Rafael Bonilla, and William Yurcik. A survey of PKI components and scalability issues. In *Proceedings of the 25th IEEE International Performance Computing and Communications Conference, IPCCC*, Phoenix, AZ, 2006.

Dette er en postprint-versjon / This is a postprint version.
DOI til publisert versjon / DOI to published version: 10.1109/MILCOM.2015.7357565