Ensuring Low Cost Authentication with Privacy Preservation in Federated IMS Environments

K. Bekara[#1], Y. Ben Mustapha[#2], S. Bouzefrane[*1], K. Garri[*2], M. Laurent[#3] & P. Thoniel[$1]

[#] *Institut Telecom,Telecom SudParis, CNRS Samovar UMR 5157*
9 rue Charles Fourier, 91011 Evry, France
{[1]kheira.bekara, [2]yosra.ben_mustapha, [3]Maryline.Laurent}@it-sudparis.eu

[*]*CEDRIC Lab CNAM*
292 rue Saint Martin, 75141, Paris Cédex 03, France
[1]samia.bouzefrane@cnam.fr, [2]garri_k@auditeur.cnam.fr

[$] *NTX Research SA*
30 rue de Saussure - 75017 Paris - France
[1]thoniel@ntx-research.com

Abstract— **Federated Identity Management Systems (IMS) is a promising system where an increasing number of e-services will be made available in the future for users' convenience. However in this environment, users are required to manage several identities (ID cards) and a great number of personal data. As such, simplification of users' involvement is highly needed while increasing the users' confidence, and guaranteeing security. This paper proposes a low-cost authentication solution which leads to a reduction of users' identities, even across several circles of trust, while maintaining high-level security. Also it proposes a privacy preserving technique to automatically control that privacy preferences of the users are satisfied during electronic transactions. This leads to defining new entities in the federated IMS, an innovative privacy policy language XPACML, and a practical-oriented privacy policy comparison middleware.**

***Keywords*— Digital identity management, privacy, PKI, authentication.**

I. Introduction

The proliferation of e-services (e.g. e-commerce, e-health, e-government) within the emerging digital Identity Management Systems (IMS) makes Internet an undeniable convenient and powerful tool for users. In the future, most of the administrative services will be put online for the convenience of users who will be serviced in almost real time, at any time, any day, and from anywhere. However, one must keep in mind all the inherent dangers that could result from the misuse of these attractive e-services. To list a few of them, an attacker successfully spoofing a legitimate user will have then power to cause really bad damages like illegal bank transfers, e-shopping with someone else's credit card, collecting very sensitive personal information of spoofed users like their social security number, their amount of wages, etc.

This paper is addressing two critical research directions of the IMS. (1) The authentication operations must implement strong enough security to deny unauthorized access to users' account. (2) Privacy of personal data attached to users' account must remain under the full control of the users.

In this paper, we are mostly interested in the federated IMS model (e.g. Liberty Alliance, Shibboleth) [1,2] which is the model that best permits national institutions (banks, government…) to jointly collaborate within a security and legal framework to increase the range of available online services. In the federated IMS model, digital identities of a single user across different service providers are linked or mapped by the use of pseudo IDs to keep institutions as much independent as possible, and to help preserving privacy of users.

This paper refers to the federated IMS vocabulary and following entities [3]: the Identity Providers (IDPs) whose role is to manage the individuals' identities, and proceed to online authentication, the Service Providers (SPs) which offer online services to the authenticated individuals or users, the Attribute Providers (APs) which supply the users attributes to any authorized entities while not compromising users' privacy. The federated IMS also defines a Circle of Trust (CoT) as the set of IMS entities establishing trust relationship among each other through a set of business agreement and technological framework. However, for the users to maintain full control on their personal data, and authentication credentials, the paper also incorporates some technical elements from the user-centric IMS model [3]. That is, it is assumed that users are equipped with an ID selector likely to the Microsoft CardSpace solution [4]. Each ID of the selector is associated with authentication credentials and the privacy preferences of the users.

This paper is organized as follows. Section 2 presents the existing works related to users' authentication and privacy within federated IMS models. Section 3 gives an overview of our scientific contribution for better security support in IMS. Section 4 introduces the main actors involved in our architecture. Sections 5 and 6 detail our PKI-based authentication approach and privacy preserving solution. Section 7 gives conclusions and future perspectives.

II. Related Works

The notion of trust and PKI management (Public Key Infrastructure) as defined in [5, 6, 7, 8] has been used in IMS such as using PKI within a federated architecture. Tran et *al*., in [9], targeted particularly a pan-European multi Circle of Trust environment. A more sophisticated work introduces a formal semantics based calculus of trust that explicitly represents trust and quantifies the risk associated to trust in PKI and identity management [10]. Le and Bouzefrane, in [11], addressed the interoperability issues between Liberty Alliance and CardSpace, while Jorstad et al. in [12] tried to integrate the current SIM authentication used in GSM with both Liberty Alliance and CardSpace such that it can be used for Internet services. However, all these research works targeted generally a special identity management system using a special physical support. The PKI-based approach proposed in this paper enables managing several circles of trust, whatever their technical solution adopted to implement their identity management system. Moreover, with our solution, the user may access services after the registration and authentication phase thanks to any physical support (smart card, USB key, cell phone, etc.).

In addition to authentication, the privacy principles and requirements based upon the European legislation [13] and the OECD guidelines [14] are enforced by technical means. These means target the minimization of the amount of personally identifiable data that are collected, as well as the enforcement of the privacy agreement between data collectors and personal data owners. Most of the works of the literature are theoretical and are hardly applicable to Internet needs. For instance, several frameworks for privacy support are proposed [15, 16, 17]. Most of the practical results are related to privacy policy languages to support the final objective of privacy policy negotiation during electronic transactions with service providers. W3C developed the P3P (Platform for Preferences Privacy) specification [18] to enable SPs expressing transparently their privacy policy in a standard machine-readable format. P3P permits SPs to express their privacy policies thanks to three XML tags: PURPOSE (why the SP is requesting data?), RECIPIENT (who will share the collected data with SPs) and RETENTION (how long data will remain stored at the SP?). These policies are assumed to be processed automatically by enabled P3P web browsers during online transactions. Other languages were developed like APPEL and XPref to better express the user's preferences, but none of these languages is adequate to support privacy policy negotiation. XACML (eXtensible Access Control Markup Language) [19] is a flexible language firstly developed by OASIS as an Access Control Policy language and an Access Control request/response language based on XML. A number of works [20, 21, 22, 23] conclude that preserving privacy using XACML seems to be an interesting solution to define both the user's preferences and the SP's privacy policy. However none of them proposes the necessary changes and extensions to the basic specifications.

III. Overview Of Our Research Work

The objective of our approach is to offer a secure and privacy preserving environment for the users to perform electronic transactions with confidence. In the federated model of IMS, any entities within the same CoT are assumed to be configured with a certificate that can be checked by other entities of the CoT, so secure communications can be established between any of the entities. Note that some other trust relationships are necessary as described in section 5. The SP is assumed to have configured his web server with its own privacy policy. This policy is expressed into our own XPACML language (eXtensible Privacy Access Control Markup Language) [24, 25]. XPACML is an extension to XACML where P3P basic tags - Purpose, Recipient, and Retention – Data Type and Service Type are defined. That is, we consider the privacy preservation problem as a problem of access control to user's personal data. Further details are given in section 6.

When joining for the first time the CoT, the user has to enrol himself to the CoT. Two entities are involved. The first entity is a proximity agency called Local Registration Agency (LRA) that allows processing the user enrolment and generating credentials. The second entity is assigned to each CoT and is called Electronic Notary (EN) server used to check the credentials. The user is also asked to configure his privacy preserving tool with his privacy preferences. Defining privacy preferences is a cumbersome task. For each of his ID card, his personal data (address, name, birth date…), for each service category, the user has to express his P3P preferences in terms of Purpose, Recipient and Retention. As a convenient support for configuring preferences, we designed a user interface that enables three configuration levels according to the expertise of users (high-level, middle-level and fine-grained). The resulting privacy preferences are stored into XPACML format file.

During a transaction, as depicted in Fig. 1, the user first connects to the SP of a CoT. He is first required to authenticate to the SP. To do so, the user holds his key pair and his certificates (certificate of public-key and certificate of public-key ownership) generated by the LRA. The certificate of public-key ownership published by a LRA on the EN server is used to check the validity of the self-signed certificate of public key. After authentication completion, the user is asking the SP to send its privacy policy, and the XPACML privacy policy is sent to the user's client. The client then has to select one of his identities that enable the transaction through his ID selector. Right after selecting his ID, the privacy middleware of the user gets the XPACML privacy preferences of the user for that ID and checks (for the attributes awaited by the server) the compatibility between the SP's privacy policy against the privacy preferences permitting/denying delivery of attributes under user's P3P conditions. In case of matching, the transaction is accepted, otherwise, a negotiation process is launched aiming to build automatically counterproposals for the SP. The negotiation process and the middleware devoted to are explained in section 6.

IV. Architecture And Main Actors

Our proposal refers to the following entities:

- *User entity*: The user, as a consumer or citizen, is linked to a circle of trust. First of all, he has to register himself within a LRA to access the system. This is the enrolment phase. Afterwards, he is able to communicate with the system to manage his own account (update, revoke or renew). Of course, he is also able to check any public-key certificates on any EN servers delivered by any Registration Authority (RA) of any circles of trust.

- *Local Registration Agency (LRA)*: This entity manages clients' enrolment by generating public-key certificates and publishing/deleting public-key ownership certificates onto the EN server. A public-key ownership certificate is used to ensure the authenticity of the public key.

- *Electronic Notary (EN) server*: The EN server registers the ownership certificates. It is requested by other actors to authenticate the public-key certificates. It stores the public-key ownership certificates issued by LRA. The communication between EN and LRA is secured thanks to SSL certificates managed by an internal PKI. The EN server is requested to verify the users' public-key certificates delivered by its own LRAs.

- *Service Providers (SP)*: The SPs are servicing services to users and are responsible for making sure of the identity of the users before granting them access to the local resources. As such, providers are asking EN server to check the public key certificates claimed by users.

- *Identity Providers (IDP)*: The IDP, one per circle of trust, stores public-key certificates generated by *LRA*. The publication of these certificates is done automatically thanks to a software module integrated within the user's identity selector.

V. PKI 2.0 protocol

The authentication protocol proposed here is called PKI2.0. It differs from the standard PKI by avoiding the use of commercial certification authorities. In fact, PKI2.0 protocol proposes to replace certification authorities by local agencies that have access to an electronic server acting as a notary to check the user credentials. These agencies are attached to a particular circle of trust. PKI2.0 consists of two phases. The enrolment phase involves the registration of new users and the generation of keys and certificates. The verification phase that assumes the publication of new public-key ownership certificates on the EN server in order to check the validity of the user certificates.

The principle is detailed in the following sub-sections.

A. Enrolment Phase

The enrolment operation is divided into three steps:

1) *Key-pair generation*: To get a PKI 2.0 certificate, each user must have a pair of keys generated by a LRA. At this stage, the user gets two keys: a public key K_{pub} and a private key K_{priv}.

2) *Certificate generation:* The PKI 2.0 protocol recommends for each user two pairs of keys, one pair is dedicated to self authentication and electronic signature and the other pair to encryption. These pairs of keys have to be stored in a secure way, especially for the private keys. The X.509v3 public-key certificates are self-signed and stored in plaintext. The first one is an authentication/signature certificate. The second one is dedicated to encryption.

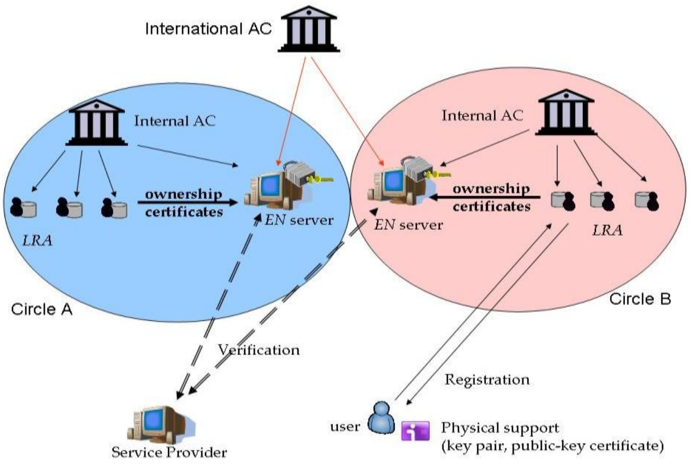

Fig. 1 Architecture overview

3) *Enrolment operation:* The user enrolment process is done directly within a LRA by first checking the identity of the user, then by generating public-key certificate and public-key ownership certificate. In fact, once the user identity is checked, the registration agent launches the public-key certificate generation after having generated the corresponding key pair. This certificate contains: third party nationality (FR), third party type (registration agency), third party circle of trust, time-stamping (validity period of the certificate), user identity, public key, self-signature (with the user's private key). The user is responsible for certifying the ownership of his public key with no certification authority involvement. With this purpose in mind, PKI 2.0 adds another certificate, called "public-key ownership certificate", which does not contain the user's public key. Instead, it contains the hash value of the public key (by using a hash function like MD5, SHA-1 or RIPE-MD). Hence, a certificate of public-key ownership contains: nationality of the third party (FR), type of the third party (registration agent), third party circle of trust, time-stamping, user identity, public key hash value. Once generated, this certificate is encrypted with the user's private key.

B. Publication of public- key ownership certificate

The public-key ownership certificate allows checking the authenticity of the public key through the hash value of it which is inserted into the certificate. This certificate must be published onto the *EN* server. This is done thanks to an SSL communication with the *EN* server. The SSL authentication is done mutually by using SSL certificates generated by an internal PKI. Once the mutual authentication is achieved successfully, a message is sent to *EN* server with the following information:

$$\left\{ Id_{LRA_i}, M, \left\{ H(M) \right\}_{Kpriv_{LRA_i}} \right\}_{Kpub_{EN}}$$

Where Id_{LRAi}: is the identity of the LRA_i,

M: contains the ownership certificate encrypted with the private key of the user, and other information like : serial number, version, signature algorithm.

H(M): the hash value of *M*. All the information is encrypted with the public key of *EN* server.

The use of the signature enables to guarantee the message integrity. The encryption with the public key of *EN* server guarantees confidentiality as only the appropriate *EN* server will decrypt the message. Upon receiving the message, the *EN* server begins decrypting the message using its private key, and then it checks the signature. To do this, the *EN* server reads the identity of the sending *LRA* because it has a register of all the public-key certificates of its local agencies, indexed with their serial number. Then, *EN* server begins extracting the certificate that contains the public key of the concerned *LRA*, in order to verify the signature. *EN* server computes the hash value of *M* and compares it to the one received. If they match, *EN* server stores the certificate of ownership in its database; otherwise an error message is notified to the *LRA*.

C. Public key certificate verification

As shown in Fig.2, this operation enables a user, an IDP, or a SP to check the validity of a public key certificate in real time in order to access an Internet service or to exchange data with another actor. The verification process is done as follows. First, the public key certificate to be verified is sent to the *EN* server which address is included in the certificate. An SSL session enables to authenticate and protect next exchanges between EN and the requesting entity. Then the *EN* server extracts the serial number which is the same serial number as the one included within the public-key ownership certificate. The *EN* server searches in its database whether the public-key ownership certificate is registered. If so, *EN* server decrypts the public-key ownership certificate with the public-key extracted from the received public-key certificate. Then the hash value is extracted and compared with the one computed with the public-key. The public key is contained in the received certificate. In case of matching, the verification is successful.

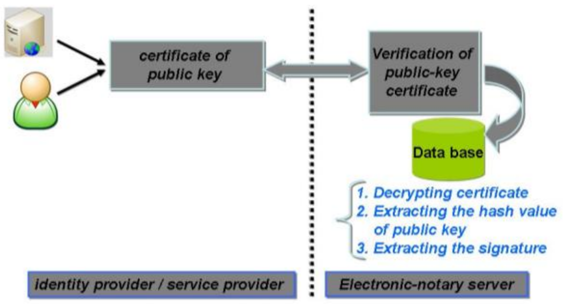
certificate of public key
Verification of public-key certificate
Data base
1. Decrypting certificate
2. Extracting the hash value of public key
3. Extracting the signature
Identity provider / service provider
Electronic-notary server

Fig. 2 Checking a public-key certificate

VI. Preserving users' privacy

After the authentication is completed, the user might perform electronic transactions while automatically preserving his privacy. In our solution, the privacy middleware of the user is in charge of negotiating the privacy policy to be enforced during an electronic transaction for the SP privacy policy and the user privacy preferences to be satisfied as much as possible. The SP policy is expressed into our own XPACML language (eXtensible Privacy Access Control Markup Language) [24] which affords the SP to define its own list of required/optional data attributes according to their categories, along with the associated P3P basic tags proposed by the P3P platform: Purpose, Recipient, and Retention. We defined a preliminary ordered classification of the privacy policy P3P tags. Like the IMS user-centric model described in [26], our client user is equipped with an ID selector. For each of his card, for each data category and for each data attribute, the user has to configure his privacy preferences according to our P3P tag classification. That is, each data attribute is assigned three separate ordered vectors related to the retention (e.g. no-retention, indefinitely), list of possible recipients (e.g. ours, delivery) and purpose tag types (e.g. current, develop, pseudo-decision). For each of these vectors, the user is required to set two borders thus partitioning the vector's set of elements into three disjoint sub-sets: Ideal, Negotiable, and Unacceptable. These sets contain tags that the client considers to be:

- Ideal: the attribute is permitted to be delivered to SP under these tag conditions.
- Negotiable: the attribute is preferred not to be delivered to the SP but delivery can be afforded if required by the SP, and if the related privacy risk is lower than a certain threshold set by the user.
- Unacceptable: the client conveys that a negotiation should fail before accepting a privacy policy containing that tag.

Furthermore, the user is required to set a sensitivity level of each attribute. This helps the middleware during the policy negotiation to evaluate the acceptable/unacceptable risk to deliver attributes under certain P3P conditions.

At the beginning of a transaction, the SP is asked to communicate its privacy policy under the XPACML format. Then the policy negotiation can start at the user's. Each policy negotiation can be seen as a set of synchronized unitary negotiations, one for each attribute, and decomposed as such. The whole negotiation is successful if all the unitary negotiations are successful. The next discussion is focusing on unitary negotiations only.

The middleware is first checking whether the attribute and P3P tags asked by the SP are part of the Ideal set of the user. If so, the attribute is accepted for delivery. Otherwise, two cases are considered. (1) For attributes tagged as optional within the SP policy, the attribute is not delivered. (2) If the attribute is required by the SP, and is part of the Negotiable set, a risk function is computed by the middleware taking into account the sensitivity assigned by the user to this attribute, and the distance between the P3P tags required by the SP and the Ideal / Unacceptable borders. This distance is evaluated thanks to our ordered P3P tag classification with the following principle: Closer is the tag to the Unacceptable border, higher is the risk; Closer is the tag to the Ideal border, lower is the risk. If the risk function associated to a tag gives a risk value lower than the risk threshold set by the user, the attribute can be delivered. Otherwise, we have a policy conflict, and the middleware searches for the tag value satisfying the risk function threshold condition and which is the closest to the required tag (according to our classification). As such, the middleware obtains a degraded policy for the required attribute. The attribute associated to the tag value forms the privacy counterproposal that is sent back to SP for approval/refusal. The negotiation protocol enables per-session privacy contract negotiations that are guaranteed to complete within a maximum of three negotiation rounds. Note that the negotiation function is an advanced feature of the privacy middleware. Both the SP and the user are required to be equipped with such privacy middleware so they are able to handle the privacy policy and counterproposals.

VII. Conclusion

This paper proposes a low-cost authentication with privacy preservation solution for IMS environments. Our PKI-based system is scalable at low cost as certification authorities are replaced by registration authorities. Moreover, our privacy middleware helps the user preserving their privacy by offering a better control over the delivery of their personal data. The users are assumed to define their own privacy preferences thanks to our designed user interface. The privacy middleware is then able to handle the privacy negotiation process by comparing the SP's privacy policy for some requested data against the user's preferences.

Our proposed solution has been implemented based on the user-centric model. This proves the feasibility of the solution. The implementation details are not given in the paper due to paper length limitation.

Acknowledgment

This research is part of the project called FC2 (Federation of Circles of Trust -www.fc2consortium.org). Kheira Bekara and Khaled Garri are partly financially supported by the DGCIS (Direction Générale de la Compétitivité de l'Industrie et des Services). We are thankful to students Thanh Ha Nguyen, Sigfrid Blasius Ngowi, Nesrine Jlidi, and Mohamed Mameri, who contribute in the implementation of the software tools.

References

[1] Liberty Alliance ID-FF 1.2 Specifications, Liberty Alliance Project; www.projectliberty.org/liberty/resource_center/ specifications/liberty_alliance_id_ff_1_2_specifications.

[2] S. armody, "Shibboleth Overview and Requirements", http://shibboleth.internet2.edu/docs/draft-internet2-shibboleth-requirements-01.html

[3] A. Jøsang, S. Pope, "User Centric Identity Management", Proceedings of AusCERT Asia Pacific Information Technology Security Conference 2005, Gold Coast 2005, pp. 77-89.

[4] Windows CardSpace "Geneva", http://connect.microsoft.com/site642/content/content.aspx?ContentID=10104

[5] J. Linn, "Trust Models and Management in Public-Key Infrastructures" RSA Laboratories, Tech. Rep., 2000. ftp://ftp.rsasecurity.com/pub/pdfs/PKIPaper.pdf

[6] M. Blaze, J. Feigenbaum, J. Lacy, "Decentralized Trust Management", in Proceedings of the 1996 IEEE Symposium on Security and Privacy, May 1996, pp. 164–173.

[7] U. Maurer, "Modeling a public-key infrastructure", in Proceedings of the 4th European Symposium on Research in Computer Security (ESORICS 96), ser. Lecture Notes in Computer Science, vol. 1146, September 1996, pp. 325–350.

[8] R. Perlman "An Overview of P K I Trust Models", Journal of IEEE Networks, Nov/Dec. 1999, pp. 38-43.

[9] D.V. Tran, P. Lokstad, D. Van Thanh, "Identity Federation in a Multi Circle-of-Trust Constellation", Report of Telektronikk 3/4.2007, pp. 103-118.

[10] J. Huang, D. Nicol, "A calculus of trust and its application to PKI and identity management", Proceedings of the 8th Symposium on Identity and Trust on the Internet, 2009, pp. 23-37.

[11] H.-B. Le et S. Bouzefrane. Identity management systems and interoperability in a heterogeneous environment. In Int. Conf. on Advanced Technologies for Communications, Hanoi, Oct., pp. 243-246, IEEE, 2008.

[12] I. Jorstad, D. Van Thuan, T. Jonvik, D. Van Thanh, "Bridging CardSpace and Liberty Alliance with SIM authentication", Proceedings of the 9th Symposium on Identity and Trust on the Internet, 2010, pp. 12-25.

[13] Directive 95/46/EC of the European Parliament and of the Council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data. Official Journal L No.281, 23 Nov. 1995.

[14] Organization for Economic Co-operation and Development, "Guidelines on the Protection of Privacy and Transborder Flows of Personal Data", Sept.1980.

[15] M. Hansen et al., "Privacy-Enhancing Identity Management," Information Security Tech. Report, vol. 11, no.3, 2006, pp. 119–128.

[16] G.-J. Ahn and J. Lam., "Managing privacy preferences for federated identity management", Digital Identity Management, pp; 28–36, 2005.

[17] M. Alsaleh and C. Adams., "Enhancing consumer privacy in the liberty alliance identity federation and web services frameworks", Privacy Enhancing Technologies, 2006, pp. 59–77.

[18] "P3P: Platform for Privacy Preferences", http://www.w3.org/TR/P3P11/.

[19] "eXtensible Access Control Markup Language (XACML) version 3.0", http://www.oasis-open.org/committees/tc home.php?wg abbrev=xacml.

[20] D. K. W. C. Vivying S. Y. Cheng, Patrik C. K. Hung, "Enabling web services policy negotiation with privacy using xacml", in IEEE 40th Hawaii International Conference on System Sciences, 2007.

[21] H.-Y. G. Yuh-Jong Hu and G.-D. Lin, "Semantic Enforcement of Privacy Protection Policies via the Combination of Ontologies and Rules", IEEE International Conference on Sensor Networks, Ubiquitous, and Trustworthy Computing, 2008.

[22] A. Anderson, "Web Services Profile of XACML (WS-XACML) Version 1.0", August 2007.

[23] L. Bussard, M.Y. Becker, "Can Access Control be Extended to Deal with Data Handling in Privacy Scenarios?", November 2009, http://research.microsoft.com/apps/pubs/default.aspx?id=105065.

[24] K. Bekara, Y. Ben Mustapha, M. Laurent, "XPACML eXtensible Privacy Access Control Markup Language", The Second International Conference on Communications and Networking (ComNet'2010), Tozeur, Tunisia, Nov. 2010.

[25] A. Davoux, J.-C. Defline, L. Francesconi, M. Laurent-Maknavicius, K. Bekara, R. Gola, J.-B. Lezoray, V. Etchebarne, "Federation of Circles of Trust and Secure Usage of Digital Identity", eChallenges e-2008, Stockholm, Sweden, October 2008.

[26] T. Yu, N. Li, and A. I. Anton, "A formal semantics for p3p", in ACM Workshop on Secure Web Services, Fairfax, VA, October 2004.