

A Model for Identity in Digital Ecosystems

Mark McLaughlin
TSSG

Waterford Institute of Technology
Waterford, Ireland
Email: mmclaughlin@tssg.org

Paul Malone
TSSG

Waterford Institute of Technology
Waterford, Ireland
Email: pmalone@tssg.org

Brendan Jennings
TSSG

Waterford Institute of Technology
Waterford, Ireland
Email: bjennings@tssg.org

Abstract—In digital ecosystems, where they are no central authorities or single points of failure or control, entities form dynamic relationships to facilitate transactions and knowledge sharing. These relationships are contextual, evolving and not subject to central moderation. The devolved structure of these emergent environments pose unique challenges for identity.

In this paper, we build on previous work on identity in digital ecosystems by offering a closer examination of identity requirements and outlining a flexible and extensible identity model framework that uses OASIS SAML-like protocols and metadata to perform the equivalent of operations such as Single Sign-On (SSO).

We also introduce an identity model software toolkit, *IdentityFlow*, that can be used to implement and execute identity protocols; and outline concrete use-cases for SSO protocol implementations.

I. INTRODUCTION

A Digital Ecosystem (DE) consists of diverse, distributed entities that sometimes compete and sometimes collaborate, interacting with each other to negotiate, transact and share knowledge.

There are a number of significant challenges for identity in a DE. Firstly, one of the main requirements is that there be *no single point of failure or control*[1]; therefore centralised authentication and identity management is impossible¹. Secondly, since relationships between entities in a DE vary over time, being sometimes collaborative, sometimes competitive, and sometimes both in separate contexts, the *trust* that backs identity claims must be dynamic and contextual². Thirdly, DE environments are technologically heterogenous, with potentially many types of platforms and infrastructures underlying entity interactions. Fourthly, the scope of an asserted identity, or the *identity context*, is not prescribed, nor is the mechanism for defining it³.

In this paper, we provide strong theoretical foundations and a model for identity in a DE that address these challenges. We show how our identity model uses ‘operations’ that provide protocols to verify identity claims and to perform the equivalent of operations such as SSO. Since federations cannot be considered stable or global in scope in a DE, SSO itself is

¹Neither central servers nor recognised authorities can play a significant role in a DE.

²As apposed to static and absolute.

³Contexts can be created from webs of trust, local identity providers, federations of contexts, etc.

limited in scope in terms of time and context, and perhaps *dynamic sign-on* might be a more appropriate term than SSO.

We introduce a software project, *IdentityFlow*⁴, and a design process for building operations. Finally, we give concrete use-cases for operations that can be employed in realistic DE environments.

Following on from work in [2], we use SAML messaging and metadata and SAML-inspired protocols (or profiles) to pass assertions backing identity claims between entities playing the role of identity provider on behalf of subjects and relying parties. The belief that an entity places in an assertion is based on the trust between identity providers and the trust between identity providers and relying parties. We remain agnostic of specific identifier standards, such as X.509 certificates or XRI.

This paper is organised as follows, in Section 2 we examine current identity work applicable to DEs. In Section 3 we outline theoretical foundations for identity in a DE and an identity and operation model. In Section 4 we outline an operation building process that can be used to provide protocols that assert identity claims in a DE. In Section 5 we give two use-cases for operations for DEs built on two separate underlying technical infrastructures and a third on a heterogenous, or mixed, underlying infrastructure. In Section 6 we give our conclusions and our plans for future work.

II. STATE OF THE ART

There is currently much work being conducted in the area of online identity, from a social as well as from a computer science viewpoint[3]. Many definitions have been put forward for digital identity[4],[5],[6],[7] and partial identities [8],[4],[9],[7]. There is only limited agreement on these definitions; when we talk of identities in this work, we define them similarly to partial identities in [8],[4]: ‘that which represents a person in a particular context in the online world’, where not otherwise specified. There is also much work being conducted in the related areas of trust and reputation[10],[11]. In this work we use *trust* to refer to reliability trust in [10].

Identity management in DEs draws inspiration and influence from the intersection of distributed/decentralised identity management[12],[13] and user-centric identity management[6],[14] The implications of trust in the latter

⁴<http://identityflow.sourceforge.net/>

is explored in [15]. The initial work in DEs was performed by [2],[16], which form a precursor to this work. Roughly speaking, the literature tends to regard identity management as a) a protocol, or set of protocols, for performing SSO (a network perspective) and b) a method of aggregating and storing user credentials (a user perspective). In this paper, we consider an identity model which gives a formal model drawing on [4] and an identity management model (from a network perspective) drawing on [2],[14] as well as a generalised identity protocol framework (which includes SSO).

From a technology standpoint, [2] looks at the federation standards SAML v2.0[17] and WS-Federation[18] with regard to DEs and concludes that the latter is overly complicated and rigid to accommodate ‘unstable coalitions’ in a DE. [13] claims that the three most used ‘protocols’ for federated identity are SAML, OpenID[19] and Windows CardSpace[20]. In terms of the network perspective, SAML and OpenID are relevant. OpenID is a lightweight SSO protocol for the WWW. SAML is a more complex and flexible set of protocols and bindings, which uses XML metadata messaging, and is therefore a good candidate for building identity protocols in a DE. Although SAML was built with stable federations in mind, SAML metadata can also be used to make (and dissolve) unstable federations[2]. We extend this approach and make extensive use of SAML metadata and SAML-inspired protocols in our operation model.

We do not prescribe the use of particular identifiers to express identity in our model. This is because we do not to impose needless restrictions on our model implementations. However, Extensible Resource Identifiers (XRI)s[21][22] can encode descriptive and contextual information within a URI, provide a clean separation between identification and addressing, and provides support for relative, contextual addressing. These attributes make XRI a good choice for identifiers in DEs. SPKI[23] is a distributed mechanism for associating X.509 identities with an identifier (public key). Relationships between entities are leveraged to provide trust and trust transitivity. However, since this trust is inherently both static and uncontextualisable, SPKI does not lend itself to our purposes.

Our model does not require entities to hold any more (and hopefully less) digital credentials or identifiers that the entity held previously. [2] propose the aggregation of credentials in a user profile, which is stored on the DE. However, by mandating the extension of SAML protocols and messaging between Service Providers (SPs) and Identity Providers (IdPs), as well as between IdPs, the requirement to hold additional credentials is obviated. This is because the user need never supply credentials directly to an SP since SPs can consume SAML assertions directly. This does require legacy SPs to be modified to consume and interpret assertions, but we argue that the gains in disposing of the overhead of maintaining a user profile, and the complication of dealing with collections of heterogeneous credentials, outweigh this drawback.

Fig. 1. A portion of a DE illustrating entities participating in various identity contexts (not necessarily as the same actor).

III. IDENTITY AND OPERATION MODELS

Because there are no single points of failure or control in a DE, we cannot assume a central naming registry, such as Domain Name Service (DNS) on the WWW, much less a central registry of identities. In certain instances, such registries may be admitted, and for certain semantic specification may be required, but if a DE is explicitly reliant on such registries then it is inherently flawed by definition. Local and provisional naming should be utilised wherever possible, and therefore identities should be similarly local and provisional.

A. Locality in Identity Contexts

We define ‘locality’ in identity contexts by the following requirements:

- 1) The number of entities participating in the context should be ‘small’⁵.
- 2) The scope of the context should be concisely and narrowly defined⁶ (delimited by purpose and time).
- 3) Identities should be (as far as possible) unique or pseudo-unique in an identity context.

The lack of a central registry and requirement 3, enforce requirements 1 and 2, since only by making the context ‘local’ can naming and identity be (effectively) unique. This is important since many applications assume uniqueness of identity. Where contexts themselves are named, e.g. by a namespace, names⁷ should be chosen that are pseudo-unique.

Fig. 1 shows entities in a DE participating in identity contexts. Identities only apply in individual identity contexts, therefore entities do not have a single canonical identity but a set of identities.

B. Construction of the Scope

We identified two mechanisms that combine to determine the scope of identity contexts:

- 1) Locally co-ordinated identity: specifies the set of entities that share an IdP for a certain identity they hold that trust the asserted identity of the other entities in the set implicitly.

⁵Small enough such that the context does not become dominant in the DE.

⁶Contexts should be established for a particular purpose.

⁷A name can be an identifier, URI, random number, etc.

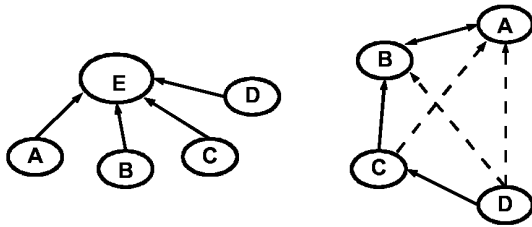


Fig. 2. Locally co-ordinator identity (left), and trust transitive networks (right).

- 2) Trust transitive networks: specifies the interconnected network of entities whose identities are trusted (sufficiently) by other entities with whom it has a trust relationship.

In (1), single, local identity providers, or co-ordinators, provide identity for a portion of the context. In (2), a web of trust uses contextual trust transitivity, with trust ratings indicating a (directional) measure of trust between individual entities in the context of identity referral and/or provision, extends the context. Trust values can be made to vary dynamically based on reputation or experience[24]. Via the two mechanisms outlined, the scope of the identity context can be said to emerge from a dynamically inter-connected network of trust relationships. Pre-existing IdP managed environments and trust networks can be combined to create evolving identity contexts from loosely federated identity domains, using trust as the glue.

In Fig. 2, left, A, B, C and D are locally co-ordinator by E, whilst, right, A, B, C and D are connected in a trust transitive network.

C. The Identity Model

The identity model is valid for identity contexts, regardless of how the scope is constructed. Following on from the development in [2], we outline the basic actors and elements in identity interactions. We define an identity operation as a contingent network protocol that verifies or asserts a claim, e.g. to verify that the ‘speaker’ has the identity that was claimed.

- 1) *Subject*: Often User or User Agent. The subject of the identity operation.
- 2) *Relying Party (RP)*: Often Service Provider. The party that relies on the result of an identity operation.
- 3) *Identity Provider (IdP)*: Credential Provider in [2]. The party that is asserting a claim in an identity operation.
- 4) *Identity Context*: The context in which the above actors interact and in which their identities are valid and pseudo-unique.
- 5) *Digital Ecosystem*: The DE. The set of all interrelated entities in the ecosystem. Entities behave as one of the above actors during operations in DEs.

We use generic actor designations to recognise the fact that actors can interact for many reasons to complete any operation, hence ‘subject’ rather than ‘user’ and ‘relying party’ rather than ‘service provider’. Entities in a DE may play the role of different actors at different times and in different contexts.

D. The Operation Model

The *Operation Model* is a meta-model for operations. The identity model, above, orientates the operation model around identity operations.

- 1) *Actor*: A role that an entity plays in an operation. (The common actors are given in the identity model.)
- 2) *Connection*: A relation between two actors that also provides an abstraction of a uni-directional communication between them.
- 3) *Profile*: A scheme that dictates how a portion of a protocol is conducted, by specifying a contingent ordering of connections.
- 4) *Binding*: The transport definition and logic adopted by connections in a given profile.
- 5) *Operation*: A specification of an operation, including the profile(s) required to conduct it.

The use of profiles are inspired by the SAML v2.0 specifications. We require an operation model rather than a prescribed set of profiles since DE environments are dynamic and heterogeneous, rather than a collection of stable federations. In webs of trust, in particular, complicated interactions may be required in order to produce a result. e.g. an RP might require three separate IdPs to vouch for the identity of a subject. (See section III-F for further examples.)

E. The Role of Trust in Operations

Operations specify how actors should communicate to perform an identity related task. Trust can be built into operations by requiring redundant consent or assent from other parties, where the input of these parties would not be required in a ‘trusting environment’. Trust can also be evaluated dynamically between participants during an operation, which we can mandate must be sufficient in order for the operation to succeed.

One of the variables in the connection relation can refer to a trust threshold. If trust levels between the two actors (in the direction of the connection) are lower than the threshold, the communication delivered by the connection can be considered untrusted, and the operation can either fail or be deemed unsuccessful. Alternatively, or in addition to this mechanism, trust transitivity can be used to aggregate the trust between actors participating in an operation to produce an overall score, which must exceed a certain minimum trust threshold for the operation to be considered successful.

Fig. 3 shows how trust relationships can support SSO (identity assertion) type operations.

F. Examples of Operations

In all operations, one or more parties (usually an IdP) make a claim about a subject to an RP, usually in the form of an assertion statement. Any claim or assertion of that form that can be settled by passing messages between the actors involved, can be verified by an operation, provided there is sufficient trust (see section III-E). Below we outline briefly three examples of likely real world applications. (These examples are developed further in use-cases in section V.)

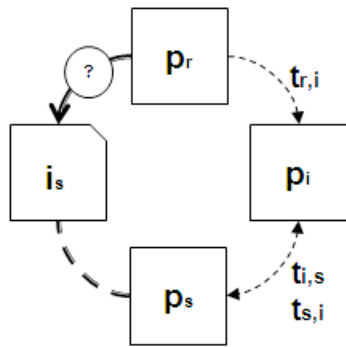


Fig. 3. Trust relationships are leveraged so that p_r can trust an assertion of p_s 's identity (i_s) made by p_i .

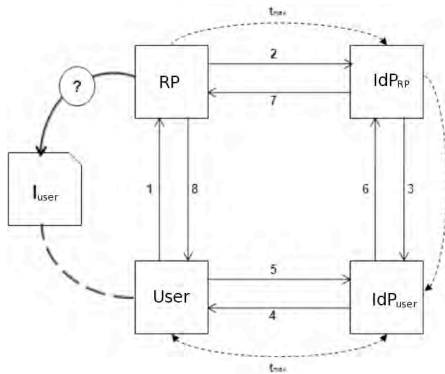


Fig. 4. A sample operation fitting the example in section III-F3

IV. OPERATION DESIGN

Operations are designed to verify or assert claims in environments, such as in a DE, where central servers and authorities cannot be leveraged to forge long standing federations. Operations may be designed by security experts, application developers or domain specialists.

A. Design Process

Operations are contingent protocols, which can be modeled as graphs with actors as nodes and connections as directed edges. Where two edges lead away from a node, this indicates that the protocol can fork at this point, based on dynamic factors that emerge during protocol execution.

1) *Design Specification*: A domain specialist can outline a scenario, naming the actors involved and the set of interactions that must follow between them, and this description can be easily rendered into an operation graph.

2) *Profile Design*: This operation graph can then be transposed into a profile for use in an operation by a security or software expert. Profiles contain sets of connections that are executed contingently for performing all or a portion of the operation protocol flow. Profiles can be executed in series, parallel or conditionally, and profiles may be nested. This enables the profile designer to design a profile in isolation from other profiles, and for the work of the overall operation design to be divided among multiple designers with the appropriate interest or expertise⁹.

3) *Operation Design*: The operation specifies appropriate definition information and a top-level profile, which is the starting point of the protocol flow execution. The operation profile(s) specify all actor interactions involved in the protocol flow (since operations are not incomplete specifications but fully implementable).

4) *Binding Design*: Once the operation graph has been translated into a set of profiles, a binding must be specified for each profile to provide the network transport for connection messages. Bindings are indicative of the underlying 'type' of actors involved and the means by which they normally communicate. Bindings are analogous to SAML bindings. Since each profile can specify a different binding, profiles can specify portions of the protocol that operate in entirely different environments, leading to operations that function in highly heterogeneous environments.

The same actor can be involved in multiple profiles using different bindings in a single operation by implementing an 'interceptor'¹⁰ for each binding environment. Currently all bindings use SAML metadata: SAMLRequests, Responses, Assertions, Statements, etc., to verify and assert identity claims.

B. Software Platform

As part of the OPAALS project, we produced a sourceforge software project, *IdentityFlow*, which allows designers to build

1) *IdP asserts to Service Provider that subject is in a group*: The Service Provider (SP) relies on the IdP to assert that a given subject, a user or service in a composed service, is a member of group A (membership of group A is also maintained by the IdP). Again the operation co-ordinates the communications of the actors such that the IdP responds to a request from the SP verifying a claim made by the subject in the form of an assertion.

2) *Trust recommendation request in web of trust*: A small farmer F establishes a DE presence for the first time and wants to do business with a co-operative S. S has no established relationship with F, but has a trust relationship (in the context of referrals) with G and H, who know F and will vouch for F. S decides that it trusts G and H sufficiently such that if both vouch for F, S will trust that the farmer F is a faithful business person. The operation co-ordinates the set of connections between all parties that establish this.

3) *Single Sign-On for local co-ordinator identity context*: A user is logged into his⁸ IdP and wants to participate in some activity that requires that his identity be recognised in a legacy domain X. The user's IdP vouches for the user to the IdP of X according to some protocol flow governed by the operation. Fig. 4 illustrates a sample operation fitting this example, where the RP co-ordinates the activity and IdP_{RP} is the IdP of X.

⁸User-chosen in the sense of user-centric.

⁹A thorough description of this process is beyond the scope of this work; see <http://identityflow.sourceforge.net/> for further details.

¹⁰An *IdentityFlow* concept.

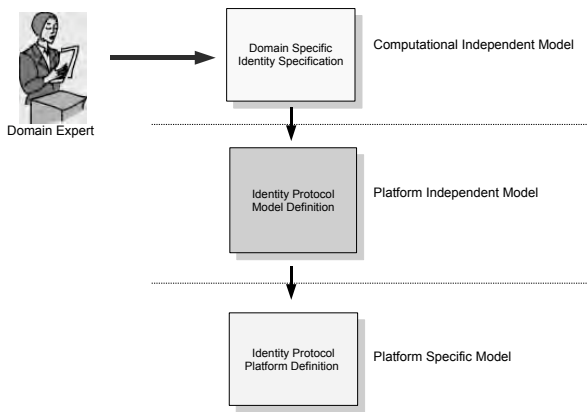


Fig. 5. A general illustration of the MDA approach used by *IdentityFlow*.

operations, profiles and bindings, and execute the operation protocol flow. HTTP GET/POST redirect bindings and a sample SSO operation is available at the time of writing and a JXTA¹¹ binding is in development.

IdentityFlow is currently implemented in Java and uses OpenSAML v2.0 libraries. An MDA¹²-like approach is used (see Fig. 5), where the highest level of abstraction is the operation model given in section III-D. The next level lower is the interconnection of actors and connections to form profiles, and profiles to form an operation, via the process outlined above. The lowest level is the implementation of these elements. The correct connection implementations can be injected for the chosen binding.

V. USE-CASES

We outline the following use-cases to demonstrate our identity model and operation design and execution framework.

- 1) Case 1: The group membership assertion scenario outlined the example in section III-F1.
- 2) Case 2: The trust recommendation scenario outlined in the example in section III-F2.
- 3) Case 3: The SSO scenario outlined in the example in section III-F3.

A. Case 1

We imagine a scenario where trusted participant machines, indicated by group membership, collaborate in a sensitive distributed computing project, and expose web service (SOAP) interfaces for communication.

In this case, we let the entities be web services (SOAP) that communicate using a SOAP binding. Shown in Fig. 1, the entities in this scenario are represented by server machines which participate in dynamically composed distributed computing services. The web service represented by the '?' is the subject, the ones with the tick are verified members of a given

¹¹<https://jxta.dev.java.net/>

¹²<http://www.omg.org/mda/>

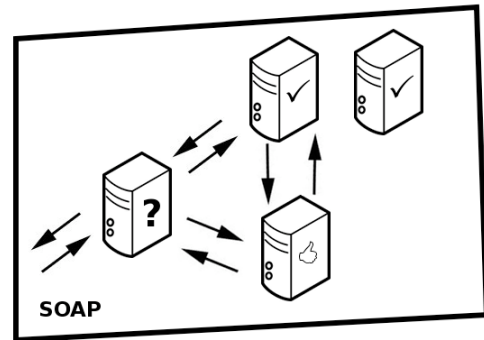


Fig. 6. An illustration of use-case 1.

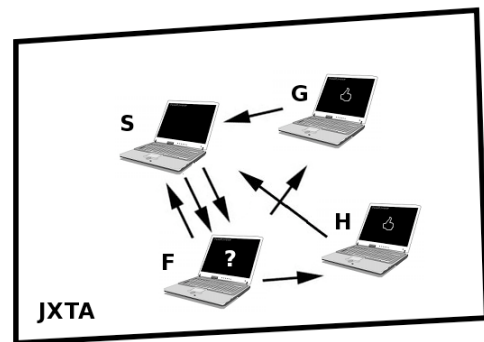


Fig. 7. An illustration of use-case 2.

group, where the one of the left is the RP. The one with the 'thumbs up' operates as the IdP for the subject in this context. The rule here is that if a trusted IdP verifies the subject's group membership to one current participant, the subject can participate in composed services.

B. Case 2

We imagine that a small farmer in India wants to establish a business relationship with a co-operative, operating in a DE, where the co-operative will only deal with farmers who are recommended by their peers and current members. The co-op uses a JXTA P2P DE infrastructure.

In this case, the entities are peers in a JXTA PeerGroup, which communicate using a JXTA (unicast pipe) binding. Based on the example, we show in Fig. 2 how the peers, representing farmers online might interact. The rule here is that if one designated peer, representing S, receives recommendations from two other peers (G and H) on F's behalf, F is authorised join the co-op.

C. Case 3

We imagine that a user has established an identity on a user-centric IdP (IdP_u) and wishes to consume a service using the

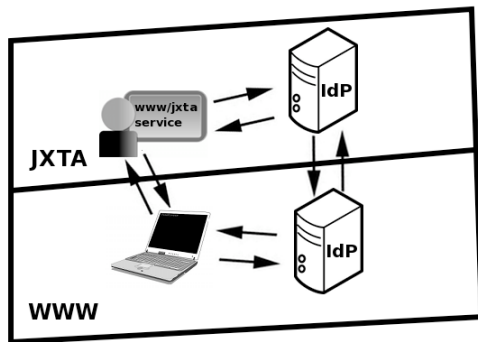


Fig. 8. An illustration of use-case 3.

same identity in a legacy domain where identity is provided by an IdP (IdP_x) which has a trust relationship with his own IdP. The user's navigable environment is the WWW, although the SP and IdP_x have a presence on the WWW and a JXTA PeerGroup.

In this case, the user uses a web browser and his IdP is on the WWW, while the SP and the SP's IdP can intercept operation connections from both the WWW, as websites, and a JXTA PeerGroup, as peers. This is accomplished by implementing an 'interceptor' for both the WWW and the JXTA environment so that operation connections can be received from both. Fig. 3 shows how an SSO operation consisting of the four actors might interact. The rule here is that if there is sufficient trust between IdP_x and IdP_u , the assertion of the user's identity is accepted by the SP (since the other trust relationships are assumed).

VI. CONCLUSIONS AND FUTURE WORK

We have given a model of identity and a framework for building identity protocols in highly decentralised environments such as digital ecosystems. We have also highlighted a software implementation that verifies the framework and provides a basis for further protocol implementations.

The *IdentityFlow* project is currently being used to develop protocols for authentication and authorisation for the OPAALS¹³ Open Knowledge Space (OKS). In the future, we see much potential in combining our operation model with work on distributed transactions. More work must also be done to consider accountability in a DE with regard to the multiplicity of partial identities that may be in use, representing the same real world (legal) entity.

ACKNOWLEDGMENT

This work is supported by the European FP6 Network of Excellence OPAALS.

¹³<http://www.opaals.org>

REFERENCES

- [1] F. Nachira, "Towards a network of digital business ecosystems fostering local development." [Online]. Available: <http://www.digital-ecosystems.org/doc/discussionpaper.pdf>
- [2] H. Koshutanski, M. Ion, and L. Telesca, "Distributed identity management model for digital ecosystems," in *Emerging Security Information, Systems, and Technologies, 2007. SecureWare 2007. The International Conference on*, 2007, pp. 132–138.
- [3] R. Halperin and J. Backhouse, "A roadmap for research on identity in the information society," *Identity in the Information Society*.
- [4] U. Glasser and M. Vajihollahi, "Identity management architecture," in *Intelligence and Security Informatics, 2008. ISI 2008. IEEE International Conference on*, 2008, pp. 137–144.
- [5] K. Cameron, "The laws of identity." [Online]. Available: <http://www.identityblog.com/?p=354>
- [6] A. Josang and S. Pope, "User centric identity management," in *Asia Pacific Information Technology Security Conference, AusCERT2005, Australia*, pp. 77–89, 2005.
- [7] A. Pfitzmann, M. Hansen, T. Dresden, and U. Kiel, "M.: Anonymity, unlinkability, unobservability, pseudonymity, and identity management a consolidated proposal for terminology. version 0.26," 2005.
- [8] M. Hansen, P. Berlich, J. Camenisch, S. Clau, A. Pfitzmann, and M. Waidner, "Privacy-enhancing identity management," *Information Security Technical Report*, vol. 9, no. 1, pp. 35–44, 2004.
- [9] E. Damiani, S. D. C. di Vimercati, and P. Samarati, "Managing multiple and dependable identities," *Internet Computing, IEEE*, vol. 7, no. 6, pp. 29–37, 2003.
- [10] A. Josang, "Trust and reputation systems," *Foundations of Security Analysis and Design IV*, 2007.
- [11] A. Josang, R. Ismail, and C. Boyd, "A survey of trust and reputation systems for online service provision," *Decision Support Systems*, vol. 43, no. 2, pp. 618 – 644, 2007, emerging Issues in Collaborative Commerce.
- [12] D. Weitzner, "Whose name is it, anyway? decentralized identity systems on the web," *Internet Computing, IEEE*, vol. 11, no. 4, pp. 72–76, 2007.
- [13] E. Maler and D. Reed, "The venn of identity - options and issues in federated identity management," *IEEE SECURITY & PRIVACY*, vol. 6, no. 2, pp. 16–23, Apr. 2008.
- [14] T. E. Maliki and J. Seigneur, "A survey of user-centric identity management technologies," in *Emerging Security Information, Systems, and Technologies, 2007. SecureWare 2007. The International Conference on*, 2007, pp. 12–17.
- [15] A. Josang, J. Fabre, B. Hay, J. Dalziel, and S. Pope, "Trust requirements in identity management," in *Proceedings of the 2005 Australasian workshop on Grid computing and e-research - Volume 44*. Newcastle, New South Wales, Australia: Australian Computer Society, Inc., 2005, pp. 99–108.
- [16] M. Ion, A. Danzi, H. Koshutanski, and L. Telesca, "A peer-to-peer multidimensional trust model for digital ecosystems," in *Digital Ecosystems and Technologies, 2008. DEST 2008. 2nd IEEE International Conference on*, 2008, pp. 461–469.
- [17] *Assertions and Protocols for the OASIS Security Assertion Markup Language (SAML) V2.03*, OASIS Security Services (SAML) TC, OASIS Open, 2005.
- [18] *Web Services Federation Language v1.1 (WS-Federation)*, Hal Lockhart, Steve Anderson, Jeff Bohren, OASIS Open, 2006.
- [19] "OpenID specification," <http://openid.net/specs.bml>, 2006. [Online]. Available: <http://openid.net/specs.bml>
- [20] D. Chappel. (2006, April) Understanding Windows CardSpace. [Online]. Available: <http://msdn2.microsoft.com/en-gb/library/aa480189.aspx>
- [21] *Extensible Resource Identifier (XRI) Syntax V2.0, Committee Specification*, OASIS XRI TC, OASIS Open, 2005.
- [22] *Extensible Resource Identifier (XRI) Resolution V2.0, Committee Specification*, OASIS XRI TC, OASIS Open, 2008.
- [23] C. Ellison, B. Frantz, B. Lampson, R. Rivest, B. Thomas, and T. Ylonen, "SPKI Certificate Theory," RFC 2693 (Proposed Standard), Feb. 2009. [Online]. Available: <http://www.ietf.org/rfc/rfc2693.txt>
- [24] J. McGibney and D. Botvich, "A trust overlay architecture and protocol for enhanced protection against spam," in *ARES '07: Proceedings of the The Second International Conference on Availability, Reliability and Security*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 749–756.