The Role of Identity in Pervasive Computational Trust

Jean-Marc Seigneur
Trinity College Dublin
Jean-Marc.Seigneur@cs.tcd.ie

Christian Damsgaard Jensen
Technical University of Denmark
Christian.Jensen@imm.dtu.dk

Abstract. A central element in the human notion of trust is to identify whom or what is under consideration. In the digital world, this is harder to achieve due to more or less trustworthy technical infrastructure between interacting parties. However, we argue that uncertain identification may enhance privacy protection. Pervasive computing – digital and real world becoming one – has also good sides: context-awareness of computing systems allow for auto-configuration of privacy protection and trust-based decision making based on context. Proliferation of sensor technology threatens privacy though, as trust inherently conflicts with privacy. We present the role of identity and how identity can be managed in a trust-based security framework, in order to balance these concerns, and present a discussion of our design and implementation choices.

1 Introduction

Weiser's vision of ubiquitous/pervasive computing [43] will only become true when computing capabilities are woven into the fabric of every day life, indistinguishable from it. The goal is to enhance the environment and help people in their daily activities. However, the current state of the art in pervasive computing does not properly address security and privacy [3, 22]. In fact, serious privacy issues, e.g., illegitimate monitoring, can arise in such an environment due to the proliferation of sensor technology, e.g., the increasing reliance on CCTV surveillance cameras and RFID technology. The ability of computing systems to identify and adapt to their environmental context is called context-awareness [8].

Privacy can be seen as a fundamental human right "to be left alone" [4] or a basic need (according to Maslow's hierarchy of needs [26]) for a private sphere protected against others. Regardless of the definition, different mechanisms have been proposed to protect the privacy of people due to information technology. The most common mechanisms are either legislative or technological, depending on whether privacy is seen as a right which should be protected by law or a need which should be supported by the devices that are used to access the online world. We do not consider the general privacy threat of pervasive sensors but focus on the technological aspects of privacy protection in trust/risk-based security frameworks (TSF), especially techniques to control the dissemination of personal information at the level of identity. It is important that these frameworks maintain a trade-off between privacy and trust. We use TSF in its broad sense: any TSF can be used (even though the TSF being developed in the SECURE [32] project is an example of an advanced TSF).

In the human world, trust exists between two interacting entities and is very useful when there is uncertainty about the outcome of the interaction. The requested entity

uses the level of trust in the requesting entity as a mean to cope with uncertainty, to engage in an action in spite of the risk of a harmful outcome. Trust can be seen as a complex predictor of the entity's future behaviour based on past evidence. In the literature, divergent trust definitions are proposed. McKnight and Chervany [27] show that these definitions can fit together and underline that the notion of dispositional trust has its importance. Interactions with uncertain result between entities also happen in the online world. So, it would be useful to rely on trust in the online world as well, especially since real and digital world merge. The goal of TSF is to provide a computational version of trust. Researchers are working both theoretically and practically towards the latter goal. Others have shown how trust can be formalized as a computational concept [18, 25]. The aim of the SECURE project [2, 32] is an advanced TSF formally grounded and usable. The basic components of a TSF (depicted in Figure 1) should expose a decision-making component that is called when a requested entity has to decide what action should be taken due to a request made by another entity, the requesting entity.

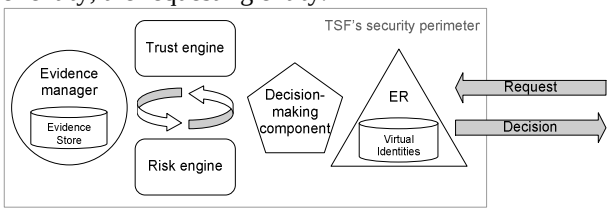

Figure 1: High-level View of a TSF

In order to take this decision, two sub-components are used:

- a trust engine that can dynamically assess the trustworthiness of the requesting entity based on pieces of evidence (e.g., observation or recommendation [41])
- a risk engine that can dynamically evaluate the risk involved in the interaction and choose the action that would maintain the appropriate cost/benefit

In the background, another component is in charge of gathering evidence (e.g., recommendations, comparisons between expected outcomes of the chosen actions and real outcomes…) This evidence is used to update risk and trust information. Thus, trust and risk follow a managed life-cycle. The Entity Recognition (ER [34]) module deals with digital identities and is in charge of recognising them. We especially put emphasis on ER in the remainder of the paper.

The next section contrasts digital and real-world trust, with an emphasis on a key element which is identity. A discussion on the advantages and disadvantages of alternative formats for trust values of entities is given in Section 3. The engineering of identity in SECURE and feedback on the design and implementation choices made is presented in Section 4. Section 5 surveys related work and we draw conclusions.

2 Contrasting Digital and Real-World Trust

In the real-world, rich context is available for trust-mediated decisions. For social scientists [27], there are three types of trust: interpersonal trust, system trust and dispositional trust. Dispositional trust is said to be independent of any party or context. Interpersonal trust is requesting entity and context specific. So, trust partly

depends on context. In computing systems, sources of context are fewer and less certain due to more or less trustworthy technical infrastructure between interacting parties. Dey defines context as "any information that can be used to characterize situation" [6] and emphasizes that not all types of context are equally important. The most important types are: location, identity, time and activity. Time is supposedly the easiest type to get (if there is no misconfiguration or timing-attack). Location is rather new but pervasive computing will provide it. Even though the notion of identity is part of legacy security mechanisms, identification is more or less certain depending on resources spent for security. Capturing the real external activity of the user is still challenging for pervasive context-aware computing [6]. Since we argue for Dey's view on context (i.e., identity is part of context, indeed an important part), we say that the level of trust is computed based on context. This is slightly different from the alternative of computing trust based on identities and then context.

More has to be said about the notion of identities in computing systems. Traditionally, users to be enrolled in the administered computing infrastructure are known and what they do electronically is bound to their real-world identity. This allows for the possibility of bringing the faulty user to court. In an open environment (with no unique authority) like the Internet, it is not uncommon to be able to create as many virtual identities as wanted (e.g., email addresses) with weak links to the real-world identity. Public Key Infrastructures (PKI) with central authorities have not shown their feasibility to legally bind any human with a cryptographic key yet (mainly due to management issues). On one hand, initiatives are needed to solve the problems of managing these multiple and dependable identities [7]. One of the main issues for TSFs in pervasive computing, where no central authority is legitimate, is the fact that it is hard to verify that a sole person has created many identities who blindly recommend one of these entities in order to fool the TSF. The level of trust in the latter entity eventually increases and passes above a threshold which grants the asset. This type of attack is called the Sybil attack [10]. On the other hand, these different virtual identities can be used as pseudonyms, which are privacy enhancing techniques due to their level of indirection between the real-world identity and the electronic data.

Trust, as with privacy, is dynamic and evolving interaction after interaction. The intrinsic property of trust to evolve autonomously improves the capability to auto-configure [33]. Privacy is a constant interaction where information flows between parties [16]. Privacy expectations vary [1, 16] and depend on context [19]. So, privacy policies based on context [11, 15, 21, 24] and trust [36] can be made closer to the real-world privacy expectations. However, recalling the process of trust formation makes apparent the fact that privacy is at stake in trust-based systems. In order to be able to trust another entity, the first step is to establish the level of trust in that entity[1], which is the result of an analysis of the existing knowledge and evidence. Thus, trust relies on profiling, where more information is better, because it allows the likely behaviour of the other entity to be more accurately predicted. Any link with the real-world identity of the user changes this information into sensitive personally identifiable information (PII). This is aggravated than in real-life because information

[1] In this paper, we use the following terms as synonyms: *level of trust* and *trustworthiness*. In a TSF, they are represented as a trust value. This is different than *trust*, which is the concept.

is easily stored and retrieved for a long period of time. In Section 4, we present how we engineered identities for pervasive computational trust in order to mitigate these issues. The next section discusses some of the advantages and disadvantages of trust values format, which is a significant difference between real-world trust and computational trust. In the real-world, there is no such well-defined format, which is essential for computing systems to communicate.

3 Trust Values Format: Interoperability, Privacy, Scalability

In the literature, there is no real consensus regarding the digital representation of trust, e.g., the format of trust values of an entity, and pieces of evidence exchanged between interacting parties. At the beginning of research on computational trust, a trust value was a value on a scale [0,1] or was composed of discrete levels: full distrust, distrust, neutral, middle trust, blind trust. Then, more complex structures appeared: Jøsang's (belief, uncertainty, disbelief) (b,u,d)-triple [17] or the Trust Information Structure [40, 41] of SECURE. In this section, we look at the format from the point of view of privacy protection, interoperability, ease and accuracy of trust calculation, performance and scalability. Trust values can be more or less expressive (i.e., they contain more or less information): the level of expressiveness seems to depend on the application. However, due to the broad range of applications that can be found in pervasive computing, there should be a context mapping mechanism to adjust trust values calculated in one application to different applications or more generally different contexts. Such a mechanism increases interoperability. More expressive representation is likely to help this mapping. A trust value may be the aggregation of trust values in specific contexts: this helps to exclude trust irrelevant to the context of interest. For example, if there are two applications: one for allowing the requesting entity to drive a car and another one to ride a motorcycle, the trust value is the aggregation of a trust value for cars and a trust value for motorcycles. It makes sense that the trust value for motorcycles can be extrapolated from the trust value for cars, because the same traffic laws apply and the ability to position yourself in traffic is similar for cars and motorcycles. A trust value may simply consist of the inexpressive result of trust calculation due to privacy reasons but it is harder for mapping. In our example, knowing that the trust value for car is 0.6 (which can be the result of many pieces of evidence) is less useful that knowing that the trust value contains the success rate of the driving exam questions also found in the motorcycle exam. A trust value may include these pieces of evidence to facilitate mapping but this may violate privacy (see the latter example). At the other extreme, a trust value may only consist of the pieces of evidence without the trust calculation result, because the trust value calculation can reveal more than the value (e.g., how trust is calculated). Another reason may be that the observers or recommenders are willing to provide objective evidence without wanting to disclose the subjective feeling represented by some trust value calculation. From a performance and scalability point of view, the more trust contexts are aggregated and pieces of evidence are stored in the trust value, the larger the trust value becomes. In fact, performance and scalability are of great concern in pervasive computing where severely resource constrained devices may be found. Large trust values mean that fewer can be stored. Past history of one specific entity may be longer with trust values with more evidence though. Following (in Table 1) is a qualitative example trade-off summary between trust value format choices. The top

half of the table describes the trust values and the bottom half describes the impact of each type of trust value on each of the criteria examined, e.g., the trust value format in column 4 contains an inexpressive result of trust calculation and additional pieces of evidence and has a medium negative effect on privacy risk, a medium positive effect on interoperability and a small negative effect on scalability.

Trust Value format	contains					
Inexpressive result of trust calculation	x		x	x		x
Pieces of Evidence		x	x		x	x
Agreggation according to Context				x	x	x
	estimation					
Privacy risk	=	=	--	-	-	---
Context Mapping / interoperability	-	+	++	+	++	+++
Scalability (based on Trust Value size)	+	-	-	-	--	--

x: the Trust Value contains …
Negative effect compared to other formats: -: small; --: medium; ---: strong
Positive effect compared to other formats: =: similar;+: small; ++:medium; +++:strong

Table 1. Qualitative Assessment of Different Trust Value Formats

4 SECURE: Feedback on Choices Made Regarding Identity

Our initial investigations on a range of applications scenarios and early-prototypes (see D5.1[2] [38], D1.2 [28] and D5.2 [32], which provides an instantiation of the SECURE framework in Java) advocate that the choice of a representation of a trust value is possible as long as trust and information orderings (see D1.1 [14] and D1.2 [9]) are defined on trust values. The SECURE deliverable D1.2 [29] explains how to come up with trust value representations compliant with the formal trust model. In SECURE, the finalised version of a trust value has yet to be finalised but a (s,i,c)-triple (where s is the number of events that supports a proposition f, i is the number of events that have no information or are inconclusive about f and c is the number of events that contradict f) seems a promising format, which brings the required properties and can be used in diverse applications (D1.3 [30] develops further the use of (s,i,c)-triples). For example, a (b,u,d)-triple [17] can be computed from (s,i,c)-triples. The current official format of trust values of an entity in SECURE is used within the Trust Information Structure [40, 41]. There are three layers: the bottom layer with the list of pieces of evidence; a middle layer with two types of trust values (trust value due to observations and trust value due to recommendations) to avoid issues related to the use of second-hand evidence; the top layer with combined trust values which are used as the local trust values for the requesting entities.
An outstanding choice related to the format of trust values has still to be made. In SECURE, it is possible to query another entity to obtain the trust value of a third requesting entity. This trust value is used as it is provided. This process is called a *reference*. If the trust value contains an aggregation of trust values related to different contexts/applications, the requesting entity doing the reference can choose to ask either for the full trust value or the part of the trust value of interest. For example, if the request for driving a car is made, the part of the trust value related to driving a motorcycle is not sent in the reference trust value. Again, there is a privacy issue.

[2] Throughout the remainder of the paper, Dx.y means the SECURE deliverable Dx.y. The deliverables cited in this document are publicly available from http://secure.dsg.cs.tcd.ie.

Requesting for the full trust value is a bigger privacy threat for the entity sending the reference than sending a specific part of the trust value. It is less privacy risky for the entity asking for the reference because it discloses less about what the requesting entity has asked for than if only a specific part of the trust value is requested.
An advantage of getting the full trust value is to allow for the best context mapping possible without several exchanges between entities involved in the reference. Due to the notion of reciprocity in privacy concerns, the final choice seems to be in favour of asking for parts of trust values. In doing so, the sending entity knows more about what the requesting entity asked the requested entity for (to compensate the disclosure of its part of trust value) and the requested entity is still able to carry out the decision making. The most appropriate way of referencing may depend on the type of application though.
Since the beginning of the SECURE project, the viability of suing any real-world identity has been considered marginal. Our expectation is that entities are in general virtually anonymous to the extent that identity conveys little information about likely behaviour. What is important as a prerequisite is not really "Who exactly does this entity represent?" but "Do I recognize this entity as a trustworthy collaborator?" As there is no a priori information concerning likely behaviour; identity therefore does not imply privilege. Before retrieving trust from the TSF, interacting entities must be recognized. It has been observed that authentication in pervasive computing systems is not necessarily enough to ensure security, because identity conveys no *a priori* information about the likely behaviour of the other entity [5, 34]. We have proposed Entity Recognition (ER) [34] as a more general replacement for authentication that does not necessarily bind an identity to the recognised entity (i.e., authentication is a special case of recognition that binds an externally visible identity to the recognised entity). We conjecture that the ability to recognise another entity, possibly using any of its observable attributes, is sufficient to establish trust in that entity based on past experience. Our end-to-end trust model [34] starts with recognition, which is a more general concept than authentication, i.e., entity recognition encompasses authentication. To allow for dynamic enrollment of strangers and unknown entities, we have proposed the entity recognition (ER) process, which consists of four steps:

1. Triggering of the recognition mechanism
2. Detective Work to recognize the entity using the available recognition scheme(s)
3. Discriminative Retention of information relevant for possible recall or recognition
4. Upper-level Action based on the outcome of recognition, which includes a level of confidence in recognition

From a privacy point of view, this use of virtual identities – pseudonyms (mapping to principals in SECURE) – is a first technological line of defence. In a TSF, the minimum requirement is a local reference for the formation of trust, which is in turn managed by other components in the TSF. According to the privacy protection principle of "collection limitation" [22], data collection should be strictly restricted to mandatory required data for the purpose of the collection.
Our requirement is to establish the trustworthiness of entities and not their real-world identity. This is why pseudonymity, the level of indirection between trust and the real-world entity, is necessary. Transaction pseudonyms [19] (i.e., a pseudonym used for

only one transaction) and anonymity cannot be effectively used because they do not allow linkability between transactions as required when building trust. There is an inherent conflict between trust and privacy because both depend on knowledge about an entity but in the opposite ways. Although trust allows us to accept risk and engage in actions with a potential harmful outcome, a computational TSF must take into account that humans need (or have the right to) privacy.

However, depending on what benefits can be reaped through trustworthiness, people may be willing to trade part of their privacy for increased trustworthiness: hence, contextual privacy/trust trade is needed. We have proposed [35] a model for privacy/trust trade based on linkability of pieces of evidence. If insufficient evidence is available under the chosen pseudonym, more evidence may be linked to this pseudonym in order to improve trustworthiness and grant the request. Some thresholds should be set concerning the acceptable evidence that should be disclosed. This is why we have introduced the link selection engagement (*liseng*) algorithm to ensure that the Minimal Linkability[3] principle [35] is taken into account. During a trade process, the following three levels must be balanced: the level of privacy asset of the evidence envisaged to be disclosed; the trustworthiness assessment impact of the evidence to be disclosed; and the utility of the requested action.

We have emphasized that care should be taken when linked evidence on multiple virtual identities is assessed. The most important requirement is to avoid counting the same evidence twice when it is presented as part of two different pseudonyms or overcounting overlapping evidence. We found [35] that in some cases, passing recommendations in the form of a simple trust value, instead of all supporting information, does not fulfil the later requirement. Assessing evidence may require analysis and comparison of each piece of evidence to other pieces of evidence. This is in favour of a trust value format including as fine-grained pieces of evidence as possible.

Our initial investigations have shown [35] that combining levels of trust in entities is not uncommon. For example, the outcome of ER [34] can be a set of n principals p (i.e., virtual entity or pseudonym) associated with a level of confidence in recognition lcr:

$$\sum_{i=1}^{n} lcr_i \, p_i$$

When we apply the APER [34] scheme (message-based recognition using cryptographic keys, hashes of previous messages and challenge/responses) to recognise the sender of an email, we combine the level of trust of principals who were using emails with a text email address and upgrade to emails as APER messages. The second scheme, called VER [37], we have been implementing is based on vision recognition: once again principals recognised with different recognition techniques must have their pieces of evidence linked and assessed.

To cope with scalability, we have proposed to forget about entities, that the entity has not collaborated with after a certain time or more generally based on context [34, 36].

[3] "No more evidence than needed should be linked."

5 Related Work

One of the main issues for the management of multiple dependable identities is the support of trust levels [7]. We indeed demonstrate in this paper that the SECURE project addresses this issue.

Wagella et al. [42] use trustworthiness of an information receiver to make the decision on whether private information should be disclosed or not, which is another way to envisage the relation between trust and privacy.

Kosba and Schreck [19] highlighted the fact that reputation systems do not mandatory require explicit link with real world identities. We added that too much evidence can lead to the disclosure of the implicit link [35].

Different SECURE deliverables discuss trust values format [29, 40] and context integration in a TSF [9, 40].

Others [12, 13, 19] have presented how pseudonyms can be used for privacy protection and shown that different levels of pseudonymity and configurations exist. Their work is valuable to choose the right type of configuration and pseudonymity.

Previous work on identity management in ubicomp environments [15, 24] demonstrated that the model of switching identities according to context is appealing and meaningful for users. Our own prototype [36], where pseudonyms are disclosed based on location, confirms the usefulness of context. Different TSFs have been used for sharing personal information in ubicomp environments [11, 39]. However, these TSFs do not use pseudonyms and their focus is not on identity matters. Another related work, although this one only focuses on recommendation, is the OpenPrivacy platform [20]. The user can create many pseudonyms linked with specific information.

Robinson and Beigl [31] investigate one of the first real trust/context-aware spaces based on the Smart-Its context sensing, computation and communication platform. We think that the combination of an advanced TSF (such as SECURE) and their sensing platform would bring interesting results. A first step would be the creation of an ER scheme based on context sensed by Smart-Its, whose name may be the CONTEXTER scheme.

Langheinrich's work [21] is valuable to understand privacy in context-aware pervasive computing. His recent work on the issue of TSF validation [23] shows that the outstanding validation of the SECURE framework is challenging.

6 Conclusion

Identity is a central element of computational trust. In pervasive computing, where there is no central authority legitimate for all entities, more or less trustworthy technical infrastructure between parties facilitates attacks (e.g., the Sybil attack) on trust/risk-based security frameworks. However, this weakness can be used for privacy protection.

Different alternatives are possible for the implementation of identity in a TSF. There is a trade-off between the aimed level of trust, privacy, interoperability and scalability.

We argue for a solution that explicitly takes into account these different levels and so can be used in a diversity of applications (as it can be expected in pervasive computing). We propose the following generic mechanisms to engineer this solution.

The potential weakness of the technical infrastructure is taken into account in our ER process thanks to levels of confidence in recognition. Our privacy/trust trade model includes means to link pieces of evidence of different pseudonymous virtual identities whilst respecting the Minimal Linkability principle.
In addition to the fact that identity is a part of context, context-awareness is promising for auto-configuration, privacy protection, interoperability and scalability. The validation of the SECURE framework (due by the end of 2004) will bring more results on these matters.

7 Acknowledgments

This work is sponsored by the European Union through the Information Society Technologies (IST) programme, which funds the IST-2001-32486 SECURE project and the IST-2001-34910 iTrust Working Group.

8 References

[1] B. D. Brunk, "Understanding the Privacy Space", in *First Monday*, vol. 7, no. 10, Library of the University of Illinois, Chicago, 2002.
[2] V. Cahill, et al., "Using Trust for Secure Collaboration in Uncertain Environments", in *Pervasive Computing*, vol. 2(3), IEEE, 2003.
[3] R. Campbell, J. Al-Muhtadi, P. Naldurg, G. Sampermane, and M. D. Mickunas, "Towards Security and Privacy for Pervasive Computing", in *Proceedings of the International Symposium on Software Security*,2002.
[4] T. M. Cooley, "A Treatise on the Law of Torts", Callaghan, Chicago, 1888.
[5] S. Creese, M. Goldsmith, B. Roscoe, and I. Zakiuddin, "Authentication for Pervasive Computing", in *Proceedings of the First International Conference on Security in Pervasive Computing*, 2003.
[6] J. L. Crowley, J. Coutaz, G. Rey, and P. Reignier, "Perceptual Components for Context Aware Computing", in *Proceedings of Ubicomp'02*, 2002.
[7] E. Damiani, S. D. C. d. Vimercati, and P. Samarati, "Managing Multiple and Dependable Identities", in *7(6)*, pp. 29-37, IEEE Internet Computing, 2003.
[8] A. K. Dey, "Understanding and Using Context", in *Personal and Ubiquitous Computing Journal*, vol. 5 (1), pp. 4-7, 2001.
[9] N. Dimmock, J. Bacon, A. Belokosztolszki, D. Eyers, D. Ingram, and K. Moody, "Preliminary Definition of a Trust-based Access Control Model", SECURE Deliverable 3.2, 2004, http://secure.dsg.cs.tcd.ie.
[10] J. R. Douceur, "The Sybil Attack", in *Proceedings of the 1st International Workshop on Peer-to-Peer Systems*, 2002, http://research.microsoft.com/sn/farsite/IPTPS2002.pdf.
[11] J. Goecks and E. Mynatt, "Enabling Privacy Management in Ubiquitous Computing Environments through Trust and Reputation Systems", in *Proceedings of the Conference on Computer Supported Cooperative Work*, ACM, 2002.
[12] I. Goldberg, "A Pseudonymous Communications Infrastructure for the Internet", PhD Thesis, University of California at Berkeley, 2000.
[13] R. Hes and J. Borking, "Privacy Enhancing Technologies: The Path to Anonymity", ISBN 90 74087 12 4, 2000.

[14] D. Ingram, J. Bacon, N. Dimmock, K. Moody, B. Shand, and A. Twigg, "Definition of Risk Model", SECURE Deliverable 3.1, 2003.
[15] U. Jendricke, M. Kreutzer, and A. Zugenmaier, "Pervasive Privacy with Identity Management", in *Proceedings of the Workshop on Security in Ubiquitous Computing, Ubicomp 2002*, 2002.
[16] X. Jiang, J. I. Hong, and J. A. Landay, "Approximate Information Flows: Socially Based Modeling of Privacy in Ubiquitous Computing", in *Proceedings of the 4th International Conference on Ubiquitous Computing (Ubicomp 2002), LNCS 2498*, pp. 176-193, Springer-Verlag, 2002.
[17] A. Jøsang, "A Logic for Uncertain Probabilities", in *Fuzziness and Knowledge-Based Systems*, vol. 9(3), 2001.
[18] A. Jøsang, "The right type of trust for distributed systems", in *Proceedings of the 1996 New Security Paradigms Workshop*, ACM, 1996.
[19] A. Kobsa and J. Schreck, "Privacy through Pseudonymity in User-Adaptive Systems", in *ACM Transactions on Internet Technology*, vol. 3 (2), 2003.
[20] F. Labalme and K. Burton, "Enhancing the Internet with Reputations", 2001, www.openprivacy.org/papers/200103-white.html.
[21] M. Langheinrich, "A Privacy Awareness System for Ubiquitous Computing Environments", in *Proceedings of Ubicomp*, 2002.
[22] M. Langheinrich, "Privacy by Design - Principles of Privacy-Aware Ubiquitous Systems", in *Proceedings of Ubicomp*, Springer Verlag, 2001.
[23] M. Langheinrich, "When Trust Does Not Compute – The Role of Trust in Ubiquitous Computing", in *Proceedings of the Privacy Workshop of Ubicomp*, 2003.
[24] S. Lederer, C. Beckmann, A. K. Dey, and J. Mankoff, "Managing Personal Information Disclosure in Ubiquitous Computing Environments", Intel Research, IRB-TR-03-015, 2003.
[25] S. Marsh, "Formalising Trust as a Computational Concept", PhD Thesis, University of Stirling, 1994.
[26] A. H. Maslow, "Motivation and Personality", Harper, 1954.
[27] D. McKnight and N. L. Chervany, "The Meanings of Trust", MISRC 96-04, University of Minnesota, Management Informations Systems Research Center, 1996.
[28] N. Mogens, M. Carbone, O. Danvy, I. Damgaard, K. Krukow, A. Møller, and J. B. Nielsen, "A Model for Trust", SECURE Deliverable 1.1.
[29] N. Mogens, M. Carbone, and K. Krukow, "An Operational Model", SECURE Deliverable 1.2, 2004, http://secure.dsg.cs.tcd.ie.
[30] N. Mogens, M. Carbone, and K. Krukow, "Revised Computational Trust Model", SECURE Deliverable 1.3, 2004, http://secure.dsg.cs.tcd.ie.
[31] P. Robinson and M. Beigl, "Trust Context Spaces", in *Proceedings of Security in Pervasive Computing*, LNCS 2802, Springer-Verlag, 2003.
[32] SECURE, "Secure Environments for Collaboration among Ubiquitous Roaming Entities", Website, http://secure.dsg.cs.tcd.ie.
[33] J.-M. Seigneur, C. Damsgaard Jensen, S. Farrell, E. Gray, and Y. Chen, "Towards Security Auto-configuration for Smart Appliances", in *Proceedings of the Smart Objects Conference*, 2003, http://www.grenoble-soc.com/proceedings03/Pdf/45-Seigneur.pdf.

[34] J.-M. Seigneur, S. Farrell, C. D. Jensen, E. Gray, and Y. Chen, "End-to-end Trust Starts with Recognition", in *Proceedings of the First International Conference on Security in Pervasive Computing*, Springer-Verlag, 2003.

[35] J.-M. Seigneur and C. D. Jensen, "Trading Privacy for Trust", in *Proceedings of iTrust'04 the Second International Conference on Trust Management*, LNCS 2995, Springer-Verlag, 2004.

[36] J.-M. Seigneur and C. D. Jensen, "Trust Enhanced Ubiquitous Payment without Too Much Privacy Loss", in *Proceedings of SAC 2004*, ACM, 2004.

[37] J.-M. Seigneur, D. Solis, and F. Shevlin, "Ambient Intelligence through Image Retrieval", in *Proceedings of the 3rd International Conference on Image and Video Retrieval*, LNCS, Springer-Verlag, 2004.

[38] G. d. M. Serugendo, C. Bryce, et al., "Application Scenarios", SECURE Deliverable 5.1, 2003, http://secure.dsg.cs.tcd.ie.

[39] B. Shand, N. Dimmock, and J. Bacon, "Trust for Ubiquitous, Transparent Collaboration", in *Proceedings of PerCom*, IEEE, 2003.

[40] S. Terzis, W. Wagealla, C. English, A. McGettrick, and P. Nixon, "The SECURE Collaboration Model", SECURE Deliverables D2.1, D.2.2 and D2.3, 2004, http://secure.dsg.cs.tcd.ie.

[41] W. Wagealla, M. Carbone, C. English, S. Terzis, and P. Nixon, "A Formal Model of Trust Lifecycle Management", in *Proceedings of the Workshop on Formal Aspects of Security and Trust*, 2003, http://www.iit.cnr.it/FAST2003/fast-proc-final.pdf.

[42] W. Wagealla, S. Terzis, and C. English, "Trust-Based Model for Privacy Control in Context-Aware Systems", in *Proceedings of the Workshop on Security in Ubiquitous Computing*, 2003, http://www.vs.inf.ethz.ch/events/ubicomp2003sec/papers/secubi03_p03.pdf.

[43] M. Weiser, "The Computer for the 21st Century", Scientific American, 1991, http://www.ubiq.com/hypertext/weiser/SciAmDraft3.html.