A Framework for Transforming Abstract Privacy Models into Implementable UbiComp System Requirements

Ivan Gudymenko
Faculty of Computer Science
Dresden University of Technology
ivan.gudymenko@gmail.com

Katrin Borcea-Pfitzmann
Faculty of Computer Science
Dresden University of Technology
katrin.borcea@tu-dresden.de

ABSTRACT
During the development of UbiComp systems, privacy and security issues often come into play only after the design process is complete. The main development effort is typically concentrated on the direct functionality of the system, which too often results in immaturity of privacy compliance of the end product. This is one of the main burdens on the way to acceptance of such systems among potential users and to commercial success thereof as a consequence.

We claim that ensuring privacy and security in any UbiComp system should be taken into account already at the system design stage and should continue throughout all steps of the development of a UbiComp system. In this paper, we focus on privacy issues of UbiComp, namely we consider a framework which enables for consistent transformation of abstract privacy models into a set of implementable system requirements. A general approach to creating an abstract privacy model, which takes into account social, legal, and functional issues, is outlined. The further transformation of the model into a set of system-specific and platform-independent requirements is described.

Author Keywords
Ubiquitous Computing, Privacy, Privacy Model

ACM Classification Keywords
K.6.5 MANAGEMENT OF COMPUTING AND INFORMATION SYSTEMS: Security and Protection—*Privacy*

General Terms
Design

INTRODUCTION

Marc Weiser, one of the pioneers in the area of Ubiquitous Computing (UbiComp), outlined this concept as "The idea of integrating computers seamlessly into the world at large [...]" [21]. Frank Stajano in his book [18] described UbiComp as "[...] a scenario in which computing is *omnipresent*, and particularly in which devices that do not look like computers are endowed with computing capabilities." According to him, UbiComp does not imply "the computer on every desk" but rather embodying the computational power into different parts of the surrounding environment (clothes, household appliances, etc.), that normally are not considered to be equipped with it thus making them "smart".

Whereas UbiComp introduces a set of tangible benefits for the user[1], it also raises serious privacy concerns. The reason for this is that the advances of sensing technology and memory amplification have provided UbiComp systems with qualitatively new opportunities of covert surveillance. Marc Langheinrich claimed in [11] that "ubiquitous devices will per definition be ideally suited for covert operation and illegal surveillance, no matter how much disclosure protocols are being developed".

Privacy concerns of the users can impede the development and especially the deployment of UbiComp systems. To give an example, alongside its intended purpose, Smart Grid systems may pave the way to privacy violation scenarios (see [8, 13] for more details.)

That means, that deploying a UbiComp system in a privacy-preserving manner will increase the likelihood of its acceptance among potential users and broaden the target audience. Moreover, having created the infrastructure of a UbiComp system, it is relatively easy to deliver the end product to customers (to deploy the system, e.g. accompany individuals with respective sensors) since "individual investments pay off immediately" [14]. Due to this fact and because of the higher acceptance among customers, a system with decent privacy management mechanisms is more likely to be commercially successful.

In this paper, we outline how a UbiComp system can be designed in a privacy-preserving way. Namely, a concept, which describes how privacy requirements can be elaborated and how respective privacy mechanisms can be "woven" into UbiComp system's functionality, is considered. This approach enables for privacy to be *inherently* built into the UbiComp system under development, which should facilitate privacy management in the deployed system and make it more efficient.

USED TERMINOLOGY

Privacy is a broad notion and defining it is a difficult task due to the substantial difference of privacy perception among individuals. However, in order to avoid ambiguity and not to confuse the reader, we present our own understanding of this notion.

[1]For example, unobtrusiveness of the devices with respect to their size and operation mode, ability of the user to concentrate on the specific (business) tasks without having to pay much attention to the management of the underlying technical system, etc.

I. Gudymenko, K. Borcea-Pfitzmann: A Framework for Transforming Abstract Privacy Models into Implementable UbiComp System Requirements. Proc. of 1st International Workshop on Model-based Interactive Ubiquitous System 2011, Pisa, Italy, June 13, 2011, http://ceur-ws.org/Vol-787

The widespread ways of understanding privacy are ”the right to be let alone” [20] and also ”the right to be forgotten” [19, 5]. One of the common delusions is that the ”more” privacy the individual has, the better his identity is protected. However, privacy does not have a monotonic behavior. The optimum is situated in the vicinity of the ”golden middle” because individuals live in society and therefore experience the need for social interaction. This implies exchanging of certain pieces of private information between communicating entities. We claim, however, that individuals, without fully realizing it, need *adequate* and *appropriate* privacy. Managing privacy implies constant processes of negotiation between the parties involved and also with the person[2] concerned of which personal information of that individual is given out in which situation and enforcing that his/her privacy policy is being followed. Thus, in order to take the aforementioned issues into account, we adhere to the following definition of privacy, elaborated in [3]:

DEFINITION. *Privacy of an entity is the result of negotiating and enforcing when, how, to what extent, and in which context which data of this entity is disclosed to whom.*

This definition takes into account the communication partner, the context, in which the communication takes place, and the negotiation processes, which are needed to flexibly manage privacy. This is necessary to reason which personal information an individual is willing to disclose to get which kind of service and to solve possible conflicts, which might arise due to the contradiction of privacy goals of different individuals. The concept of *multilateral security* [1, 15] provides for a flexible and effective way of negotiating such conflicts in a privacy-respecting environment. Moreover, which personal data is disclosed, its granularity, and the enforcement of an individual's privacy requirements are also considered in the definition above.

MAKING PRIVACY INHERENTLY BUILT INTO THE UBICOMP SYSTEM'S FUNCTIONALITY

In order to provide for a privacy-respecting and secure UbiComp system, the process of ensuring privacy and security should begin already at the system design stage, the concept of which is known as ”privacy by design”, and it should continue throughout all the other steps of system development.

It is clearly impossible to predict the security and privacy requirements of all potential users and also their variations in response to future context changes during the system design stage. In order to provide for flexibility and extensibility, a concept of special extension/variation points (so-called hooks) for unforeseeable extensions/variations of privacy and security requirements can be utilized.

Thus, the process of ”weaving” privacy and security mechanisms into the UbiComp system's functionality can be divided into the following steps, depicted in Figure 1:

1. During the system design stage, generic (i.e. foreseeable) privacy and security requirements are considered. In order to provide for flexibility in future, a concept of extension/variation hooks with respect to privacy and security requirements is used.
2. At initialization time, an instantiation of generic requirements considered during the first step is carried out. Also, the so-called *binding*[3] of extension/variation hooks is performed.
3. At run-time, the previously implemented privacy and security management mechanisms are used. In order to provide for *dynamic* adaptation (e.g in response to context changes), the concept of dynamic extension/variation hooks may be exploited.

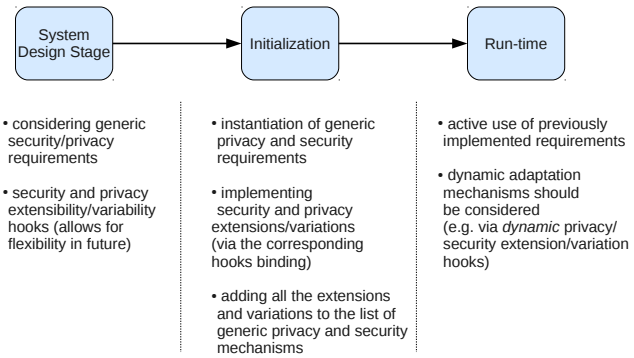

Figure 1. The process of making privacy requirements inherently built into the UbiComp system's functionality.

PRIVACY IN UBICOMP: PECULIARITIES

In order to provide for effective privacy management in UbiComp systems, it is sensible to explore which peculiarities privacy issues have in this domain.

The pervasive nature of UbiComp may impose certain constraints on the users of the system in that it might be hard for them to actually refuse to use it. This problem raises privacy concerns and is called ”the disability to opt-out” [7], where the following example was stated: it would be extremely difficult if not impossible to refuse to use the Ubiquitous RFID system in case ”such devices [RFID tags] get affixed to bank notes, ID cards, and every item that one can buy in a store”. If opt-out is nevertheless made possible, the following problems might arise:

- much inconvenience caused by opt-out (e.g. postal mail of a check instead of a credit card payment);
- opt-out can look suspicious (a denial to give away certain data in particular situations may look suspicious, e.g. switching the location sensor off during the time when a crime was committed, etc.)

Another privacy problem specific to UbiComp is a constantly rising likelihood that intimate conversations might become

[2]In each situation an individual is constantly performing reasoning about what he/she is willing to disclose to get which kind of service.

[3]The term is adopted from programing. It basically means that the corresponding hooks are being directly used, i.e. extension/variation has taken place via the hook.

publicly available. The authors of [7] call this problem ”the loss of ephemeral communication”. Similarly, Schneider states: ”The moral is clear: If you type it and send it, prepare to explain it in public later”. In this case, the problem of *violation of contextual integrity* arises. It was described in [4] as ”falsifying the context in which information has been communicated” by ”putting it into a wrong context”. For instance, consider an example of a debating club: a person receives a topic ”Should foreigners be allowed to work in Germany?” and should state arguments against it. If his speech is put into another context later on (e.g. shown at the TV) *without specifying the original context*, the speaker’s reputation might be dramatically spoiled (i.e. the ”decontextualization of communicated information” has turned ”innocuous” information into the ”mortifying” one [4]).

In [14], it was outlined that the privacy of an individual in UbiComp could be enhanced by changing the main direction of information flow to ”infrastructure → user” and applying filtering in order to avoid overload or annoyance of the user. This change of information flow ”enables a quantum leap in privacy by avoiding the possibility to gather huge amounts of personal data”. In this case, the infrastructure might also broadcast security and privacy advices (e.g. possible options, etc.) to the user if it appears to be of mutual interest to both, the provider(s) of the infrastructure and the users.

Thus, in order to provide for a privacy-respecting UbiComp system, the following issues have to be taken into account:

1. Provide for support of opt-in/opt-out according to the individual’s choice. At the same time, mechanisms against irresponsible behavior should be taken into account (i.e. non-repudiation of performed actions)[4].
2. Anonymization and encryption techniques for resource-constrained devices should be carefully considered in order to mitigate the problem of disclosure of the content of intimate conversations to public.
3. Mechanisms for protecting contextual integrity of data should be provided (especially in case of voice/video recording services, personal communication services, etc.) For example, attaching a special protected tag to data, which will specify the original context and protect the information from decontextualization, should be considered. The tag itself can be authenticated by the individual who owns the information or by the group of individuals to whom the data is relevant (using multi-party authentication, for instance).
4. It is also highly advisable to design a UbiComp system adhering to the concept of reverse information flow (”infrastructure → user”) where possible.

[4]Consider an example of an ”Ambient Coffee Machine” service in the organization, where users are able to drink coffee without being obliged to pay for it at the spot but required to do so at the end of the month. An irresponsible user might want to be using such a service for several weeks and then decide to opt-out ”due to privacy reasons” without paying. In this case, authentication and legal enforcement, for instance, can be used to prevent such case from happening.

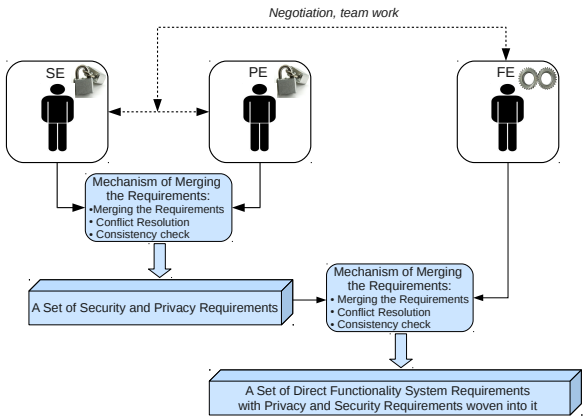

Figure 2. A process of joint development of privacy and security requirements for a UbiComp system.
SE = Security Engineer.
PE = Privacy Engineer.
FE = Direct Functionality System Engineer.

DESIGNING PRIVACY AND SECURITY REQUIREMENTS IN A JOINT FASHION

Privacy and security are closely connected to each other. Important is to understand that neither of them is a byproduct of the other one. Only if having considered both, privacy and security, can the developed UbiComp system be regarded as privacy-respecting and secure.

For this reason, we suggest that privacy and security requirements are elaborated in a joint fashion by two cooperative entities: the Privacy Engineer (PE) and the Security Engineer (SE) (see Figure 2). These entities are responsible for the whole design process of privacy and security policies respectively as well as for administrating and managing privacy and security in the deployed system. The process of designing policies for a privacy-respecting and secure UbiComp system should be carried out in the presence of collaboration between the PE and the SE. Further negotiation with the Functionality Engineer (FE), who is responsible for the design of the direct functionality of the system, should be considered as well. The reason for this is that it is expected that the requirements elaborated by the PE and the SE along with the ones of the FE may not be free of conflicts. That is why conflict resolution mechanisms should be considered during the process of merging the requirements. In order to ensure that the requirements are consolidated in a consistent way (i.e. specific requirements of each area after the merging conform to the ones before the merging), consistency checks should also be performed after the merging.

PRIVACY MODELING

In order to provide for privacy requirements, which are going to enable for efficient privacy management in the deployed system, we suggest that a corresponding model of privacy for the target domain of UbiComp is created. The respective requirements can be inferred from the model later on. Here, with the term ”abstract privacy model” we refer

to a high-level model, which takes into account social, legal, and functional issues and enables the developer to perform a combination of privacy issues from different fields in an interdisciplinary manner. Having an abstract privacy model in the first step will facilitate the process of taking various and often illusive privacy issues and considerations of the UbiComp area into account and make the approximation to the real world scenario more accurate.

Modeling privacy is not a trivial task. Existing privacy models are often abstract and difficult to transform into a set of system requirements. For instance, the model introduced in [12] deals with the concept of ”crossing personal borders”, i.e. privacy violation occurs when ”personal borders” of an individual are crossed. The author provides for a classification of privacy-violation scenarios, analyzes the privacy concerns of the individuals and also considers the impact of technological advance on privacy. However, the model is described in a loose and nontechnical way, which might impede its adoption for the process of inference of privacy requirements.

Another model was introduced in [17], which focuses on the activities that invade privacy: information collection, information processing, information dissemination, and invasion. The model consists of the data subject (the individual) and the data holders (who collect, process and disseminate private information). Similarly to the above mentioned model, it provides for a rather notional description of privacy issues and does not specify how the respective requirements can be inferred and further implemented.

Moreover, new approaches to modeling of privacy should also be considered because of the rapid evolution of technology. For instance, Shapiro in [16] gives an example of Fair Information Practices that have been commonly used for understanding informational privacy. However, he claims that ”As more things become digitized, informational privacy increasingly covers areas for which Fair Information Practices were never envisioned” (e.g. genetics, biometrics, etc.).

UbiComp definitely introduces a serious challenge regarding privacy modeling, translating a model into a set of system requirements and implementing it. It is of little help just having a good model of privacy if it can not be adopted into technical schemes of privacy regulation and thus be used within a UbiComp system. Provided that a decent and *implementable* model of privacy is available, respective privacy mechanisms should be woven into the UbiComp system functionality at the system design stage to allow for designing *inherently* privacy-respecting systems.

Therefore, it would be helpful to consider a framework which will enable for consistent transformation of an abstract privacy model into functional requirements of a UbiComp system, which in turn can be implemented.

Privacy Modeling Framework

The concept of the Privacy Modeling Framework is similar to the meta-modeling approach used in programming (e.g. meta-metamodel $\rightarrow$ metamodel $\rightarrow$ model, see [2] for more details). The task of providing for a consistent privacy model and transforming it into a set of implementable system requirements is within the competence of the Privacy Engineer (cf. Figure 2).

This approach implies several steps, which are depicted in Figure 3.

1. The Privacy Engineer entity (that might be a group of privacy experts in practice) creates an abstract privacy model. This implies the following steps:
   - investigating the privacy area of the future UbiComp system deployment, i.e. determining individuals' privacy concerns, possible privacy threats, taking into account various cultural differences in perception of privacy, etc.;
   - reviewing the current status of legal basis in the area of interest (i.e. finding out which privacy-related laws apply to the future UbiComp system deployment, how the situation is legally regulated and determining the weak sides of it);
   - creating the joint picture of privacy-related issues in the field;
   - on the aforementioned basis, an abstract privacy model is created (system- and platform-independent).
2. Next, a consistent transformation of the abstract privacy model created during the first step into a set of system-specific requirements is carried out. If some of the model preferences can not be transformed, a possible refinement of the abstract model should be considered. The result of the second step is a set of *implementable* system requirements.
3. The last step is the actual implementation (”weaving” of privacy mechanisms into the UbiComp system's functionality).

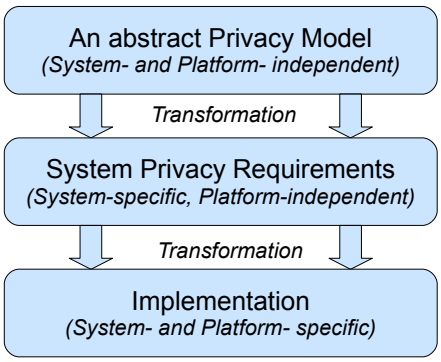

Figure 3. A general structure of a framework for transforming abstract privacy models into implementable requirements.

The above mentioned approach introduces a set of challenges:

1. Merging the individual privacy requirements with legal issues in the area of interest (step one) is a difficult task.

The reason for this is that the former is elusive and not easy to specify. The latter is well specified but coarse-grained and hence inflexible. For example, suppose it is written in the privacy law that location of the individual is private information and any exposure of this information to a third party is subject to law violation. The situation when the individual is *willing* for his location to be known to some of his friends at certain times, is not considered, however. Moreover, the legal part strongly depends on the region, which raises the question of international interoperability and aggravates the outsourcing problem, i.e. the privacy-sensitive data that is governed by law in one country, might be under threat of violation in the other one. This happens due to the absence of a unified international law protection system of privacy-sensitive data.

Having managed to specify privacy requirements of the individual and taken the legal prospective into account, the consistency of the *joint* abstract model should be considered.

2. The second step (transformation of abstract privacy model into a set of requirements) implies the existence (or creation) of a mature language that will enable to express the abstract model in a standardized, ready-to-implement format. To the best of our knowledge, only a few efforts have been made in this direction by now. The authors of [9] described their privacy model using a privacy control language that "includes user consent, obligations, and distributed authorization". In [10], a privacy-specific access control language was used to manage privacy in the environment of so-called "Platform for Enterprise Privacy Practices (E-P3P)", which defines technology for privacy-enabled management and exchange of customer data. The authors in [6] showed how a privacy policy can "be specified and implemented according to the Generalized Framework for Access Control (GFAC)-approach". In order to successfully complete the second step, it should be decided by which means the abstract model should be specified in the most comprehensive and consistent way (e.g. which language to choose or even to introduce a new one).

3. Along with privacy-specific questions, general framework-related issues arise:

   - The framework is described in an abstract way. That is why the ways of its implementation should be outlined. Moreover, it should also be considered, which degree of *automation* of the transformation process can be achieved.
   - Next, the *consistency* of the performed transformation should be carefully considered. Surely, certain trade-offs are going to arise. Their impact on the accuracy of the implemented privacy model should be assessed.

CONCLUSION AND FUTURE WORK

The paper has presented an approach to designing an *inherently* privacy-respecting UbiComp system. We claim that it is not possible to provide for a full-fledged support of privacy management, having considered this issue after designing the direct functionality of a UbiComp system, i.e. building privacy on top of the system. That is why the process of ensuring privacy and security has to begin at the *system design stage* and it should continue throughout all the other steps of system development.

Thus, an approach to making privacy inherently built into the UbiComp system's functionality was considered. In order to provide for *dynamic privacy management* (e.g. to enable the consideration of unforeseeable extensions towards privacy requirements), a concept of special extension/variation points can be utilized while designing a system.

Providing for *efficient privacy management* requires the exploration of the peculiarities of privacy in the target domain. Also, respective recommendations for developing appropriate privacy policies should be formulated. Having considered this issue, we presented our concept of designing privacy and security requirements in a joint fashion. The reason for this is that privacy and security are closely connected and mutually affect each other. According to this concept, privacy and security requirements should be considered by two cooperative entities: the Privacy Engineer (PE) and the Security Engineer (SE). Moreover, further negotiation with the designer of the direct functionality of the system (Functionality Engineer) is considered along with conflict resolution mechanisms.

The creation of an abstract privacy model was suggested to enable *effective development of privacy requirements*, which take various privacy issues and considerations of the UbiComp area into account and provide for a better approximation to the real world scenario. Respective privacy requirements can be further inferred from the model. This can be done within our Privacy Modeling Framework, which considers the creation of an abstract, domain-specific privacy model by the PE entity, further inferring respective requirements from it, and, lastly, implementing them into the UbiComp system's functionality.

Having described our conceptual view on ensuring privacy in a UbiComp system, more concrete ways of creating an abstract privacy model, means of specifying the requirements and necessary recommendations towards their implementation are to be elaborated. Finally, applying the concept to a particular real use case scenario is to be realized.

ACKNOWLEDGEMENT

The authors would like to express their gratitude to a great researcher and friend Andreas Pfitzmann who passed away in September 2010. He was not only a highly qualified professional but also a very kind and responsive person who inspired the people around him on their way to scientific excellence.

This paper is to a large extent influenced by discussions with Andreas and is written in commemoration of him.

REFERENCES

1. Andreas Pfitzmann. *Multilateral Security in Communications*. Addison-Wesley-Longman, 1999, ch. Technologies for Multilateral Security, 85–91.
2. Assmann, U., Zschaler, S., and Wagner, G. Ontologies, Meta-models, and the Model-Driven Paradigm. *Ontologies for Software Engineering and Software Technology* (2006), 249–273.
3. Berg, M., and Borcea-Pfitzmann, K. Implementability of the Identity Management Part in Pfitzmann/Hansen's Terminology for a Complex Digital World. In *Proceedings of PrimeLife / IFIP Summerschool on Privacy and Identity Management for Life*, S. Fischer-Hübner, M. Hansen, P. Duquenoy, and R. Leenes, Eds., IFIP Advances in Information and Communication Technology, Springer (2011).
4. Borcea-Pfitzmann, K., Pfitzmann, A., and Berg, M. Privacy 3.0 : = Data Minimization + User Control + Contextual Integrity (Privatheit 3.0 : = Datenminimierung + Nutzerkontrolle + Kontextuelle Integrität). *IT - Information Technology 53*, 1 (2011), 34–40.
5. Dou, E. EU proposes online right 'to be forgotten', Nov. 2010. Accessed online on 05.04.2011. Reuters. http://www.reuters.com/article/2011/03/17/us-eu-internet-privacy-idUSTRE72G48Z20110317.
6. Fischer-Hübner, S., and Ott, A. From a formal privacy model to its implementation. In *National Infromation Systems Security Conference* (Oct. 1998).
7. Henrici, D. *RFID Security and Privacy*. Springer, 2008.
8. Herold, R. SmartGrid Privacy Concerns, Sept. 2009. Accessed online on 03.04.2011. http://www.privacyguidance.com/files/SmartGridPrivacyConcernsTableHeroldSept_2009.pdf.
9. Karjoth, G., and Schunter, M. A privacy policy model for enterprises. In *Computer Security Foundations Workshop, 2002. Proceedings. 15th IEEE* (2002), 271–281.
10. Karjoth, G., Schunter, M., and Waidner, M. Platform for enterprise privacy practices: Privacy-enabled management of customer data. Springer (2002), 69–84.
11. Langheinrich, M. Privacy by Design – Principles of Privacy-Aware Ubiquitous Systems. In *Ubicomp 2001: Ubiquitous Computing*, G. Abowd, B. Brumitt, and S. Shafer, Eds., vol. 2201 of *Lecture Notes in Computer Science*. Springer Berlin / Heidelberg, 2001, 273–291.
12. Marx, G. T. Murky conceptual waters: The public and the private. *Ethics and Inf. Technol. 3* (Sept. 2001), 157–169.
13. Pentland, W. Why Smart People Are Suspicious of Smart Meters, Dec. 2010. Accessed online on 01.04.2011. Forbes. http://blogs.forbes.com/williampentland/2010/12/10/why-smart-people-are-suspicious-of-smart-meters.
14. Pfitzmann, A. Accompanying Ambient Intelligence (AAmI) – Why You should take your Sensors with You. A Sketch on the Future of privacy-aware, secure Ambient Intelligence., Apr. 2010.
15. Rannenberg, K. Multilateral security: A concept and examples for balanced security, 2000.
16. Shapiro, S. S. Privacy by design: moving from art to practice. *Commun. ACM 53* (June 2009), 27–29.
17. Solove, D. J. A taxonomy of privacy. *University of Pennsylvania Law Review 154*, 3 (Jan. 2006), 477 pp. GWU Law School Public Law Research Paper No. 129.
18. Stajano, F. *Security for Ubiquitous Computing*. John Wiley & Sons, LTD, 2002.
19. Warman, M. EU proposes online right 'to be forgotten', Nov. 2010. Accessed online on 05.04.2011. The Telegraph. http://www.telegraph.co.uk/technology/internet/8112702/EU-proposes-online-right-to-be-forgotten.html.
20. Warren, S. D., and Brandeis, L. D. The right to privacy. *Harward Law Review 4*, 5 (Dec. 1890), 193–220.
21. Weiser, M. The Computer for the 21st Century. *Scientific American* (Feb. 1991).