Security requirements for environmental sensing technology

Giovanni Iachello, Gregory D. Abowd
College of Computing & GVU Center
Georgia Institute of Technology
801 Atlantic Dr.
Atlanta, GA 30332 USA
{giac, abowd}@cc.gatech.edu

ABSTRACT
In this paper, we identify security objectives and requirements for a class of ubiquitous computing technology, namely environmental sensing infrastructure. The method used starts with analyzing scenarios to define user-driven security objectives for specific applications and identify hints to potential misuse in a realistic setting. The described applications are examined to understand how they could be built using existing sensing technology (TinyOS, InCA and the Context Toolkit are considered here). The resulting security requirements, related to technical, social and legal issues, drive the discussion about the features and shortcomings of the considered sensing technologies and aid in proposing suggestions for enhancing their security properties.

Keywords
Security, ubicomp, legislation, privacy, data protection, environmental sensing, capture and access

INTRODUCTION

Environmental sensing technology, which supports the collection, digitalization, storage and the retrieval of information from physical or virtual environments, is an essential component of many ubicomp systems. Besides recording for future use, applications use environmental information to support interaction, to build logical models of the physical world and for personalization and classification purposes.

The peculiar nature of ubicomp technology, where seamless interaction occurs in integrated and invisible systems, requires a more comprehensive approach to security than what has previously characterized ICT applications. The protection of users' privacy is one obvious concern, but inextricable from it is the issue of securing these systems against abuse, disruption and exploitation in harmful activities. This entails concerns about the technology's usefulness and fairness, its consequences on social safety and well-being and consideration of its economic aspects.

Two considerations prompt a comprehensive evaluation of security for environmental sensing technology today. First, relatively simple and closed ubicomp applications are now evolving in structured, interconnected systems, encompassing mobile telecommunication platforms such as cell phones and a variety of service providers and brokers. This shift stresses the ability of current social safeguards (including customs and legislation) to preserve a baseline protection regime for individuals.

The second observation recognizes that much of the past research has concentrated on body and spatial privacy, especially in awareness and memory-enhancing systems [3, 16, 12]. In fact, privacy issues have been present to ubicomp researchers from the early beginnings of the field [24], in connection with location-aware and environmental sensing technologies, as these formed the basis of many ubiquitous computing applications. As early as 1993, Bellotti introduced a thorough discussion on how the design of environmental video capture devices should be directed in order to comply with specific spatial privacy requirements [3]. Given the research-oriented nature of early ubicomp applications and their rapid evolution, limiting inquiry to privacy was adequate.

However, a functioning service market for ubicomp applications also needs efficient and secure mechanisms supporting service delivery, property protection, payments and arbitration. Research to date only partially addressed these new problems, most notably with work on context-based identification and access control [4, 20]. Much research has gone recently into mobile systems security, and noteworthy developments include brokerage and mediation systems for mobile telecommunications and commerce [17, 18, 19]; however, their conclusions, based on assumptions on the type of hardware available (more or less advanced personal mobile terminal systems) have been difficult to transfer to the vast array of technologies used in ubicomp.

Besides the general work mentioned above, specific technologies have been examined, such as radio frequency identification (RFID) [6] in order to understand how this

technology can threaten personal privacy, and proposals have been made in order to adapt the Fair Information Practices (FIPS) guidelines [23] to that technology. Other authors have taken inspiration from the FIPS and the OECD (Organization for Economic Cooperation and Development) guidelines for the management of personal information [15] to define high-level privacy objectives that ubiquitous computing applications should meet [12].

About Environmental Sensing Technology

Environmental sensing technology comprises all systems which collect and store information deriving from a physical or virtual environment. Although technology as simple as video surveillance cameras can be included in this definition, here we concentrate on digital, interconnected sensing systems able to automatically process and make information available for further use. As opposed to integrated applications, sensing systems are designed to be independent from specific applications, and instead provide services to any entity interested in particular types of data gathered in some environment.

Information can be sensed either from physical sources, such as sensors collecting humidity readings and recordings of conversations in a room, or from computerized sources, such as the position of a cell phone within a network or security badge access logs. After capture, which includes the quantization of measures and their codification in some digital format, sensing systems generally transfer the information from its origin to a processing center, where it is either stored or delivered to interested applications.

Stored information can be subsequently accessed based on a variety of mechanisms, which generally include query and search systems. In many cases, information is stored within the sensing infrastructure, which also provides access to it. Storage architectures vary greatly, and range from blackboard systems to linear recordings, from relational databases to data warehouses.

The sensing infrastructure can also be responsible for coordinating various other activities, among which data aggregation (i.e. the synthesis of higher-level information from raw readings, such as the location of an individual in a building based on RFID scans on doors). Finally, some systems also include subscription mechanisms to automatically notify applications when some environmental condition is verified or information of interest becomes available.

METHOD

While our work shares with many of the above-cited efforts the goal of defining security properties and requirements for ubicomp, the method is somewhat different, as we analyze user experience in an integrated manner together with existing software and hardware infrastructures. Our aim is to anticipate specific problematic issues before the actual deployment of new ubicomp services, and to define targeted requirements and avenues for improving existing applications and infrastructures in order to better support users' security goals and concerns.

Our focus on infrastructure technology is driven by the recognition of its crucial role in system security. While robust and secure infrastructure can successfully control insecure applications, secure applications cannot be build upon untrustworthy infrastructure.

The development of several technologies for advanced environmental sensing and their deployment in research projects has provided a wealth of useful experience upon we build. Specifically, we have drawn input from the experience gained from:

- the development of novel applications and services;
- their testing in experimental and real-world settings;
- the engineering solutions to recurrent complex implementation problems embodied in these systems.

Security issues are analyzed by writing "use scenarios", which describe the use of specific applications. Application requirements which are identified through the scenarios are applied to the infrastructure systems, to understand if, and how, the infrastructure supports the implementation of such requirement. This approach allows us to define significant, realistic and clear security objectives directly from real-world examples of current research or near-to-come applications. Scenarios provide not only a "story" to recount an application to the reader, but include also implicit information about the context in which the application is used, in order to understand side-conditions needed for a thorough discussion of the single case.

As part of this ongoing research effort, several scenarios have been developed and analyzed, yielding a systematic set of requirements and guidelines for ubicomp systems not limited to sensing technology, which is being synthesized in form of a technical report. One of these scenarios is reported below in a reduced form, along with a discussion of salient security and privacy requirements and some legal implications. In particular, experience with data protection legislation has been taken into account to highlight some areas where the peculiarities of sensing technology challenge the regulatory frameworks already in place.

SENSING TECHNOLOGIES

Three different infrastructures for building ubiquitous computing applications have been taken into consideration for the preparation of this paper, in order to provide solid and realistic technical roots to the discussion. We are aware of the risk of fixing on of a system's peculiar features; therefore, special care was taken to concentrate on general design and to avoid discussing aspects which would relate to the specific system.

TinyOS embedded network operating system [10]

TinyOS is an embedded operating system which runs on 8-bit microcontrollers. It provides a number of services for building wireless sensing applications, including radio

networking, interfaces to ADC ports, power management and scheduling. The operating system is designed to work on various types of hardware devices commonly referred to as "Berkeley motes" which provide sensors for acceleration, magnetic field, luminosity, sound pressure, temperature and mechanical pressure. Thanks to their small size and low cost, these devices are widely used, especially for remote measurement and simple on-board aggregation and transmission. TinyOS has been developed at UC Berkeley and Intel Research.

Infrastructure for Capture and Access (InCA)

InCA is an experimental infrastructure which supports the capture, storage, delivery and query-based access to stored multi-media information, including video, audio, digital ink strokes and text. It has been used in several projects including classroom capture systems, meeting capture and applications in the home. The infrastructure is network-based, and is composed of Java modules connected by a TCP network. Information can be tagged based on various attributes, and queried at later time based on these tags. Moreover, permanent queries can be issued (subscriptions), which allow applications to be notified when any relevant data is generated subsequently. InCA has been developed at Georgia Institute of Technology.

Context Toolkit [5]

The Context Toolkit provides a set of abstractions which can be used to sense, store and process environmental data of various kinds. The main abstraction in the toolkit is the "context widget", which contains environmental information in trees of *attribute, value* pairs. It is possible to register with a widget in order to be notified when a specific value changes. Widgets also provide services, which can be executed either synchronously or asynchronously. The toolkit is written in Java and is network-based. The Context Toolkit has been developed at Georgia Institute of Technology and later at Intel Research and UC Berkeley.

The three systems mentioned above target different aspects of environmental data sensing. As opposed to TinyOS, InCA and the Context Toolkit do not address the actual sensing and first-degree processing of information. InCA mainly provides support for storing streaming information, while the Context Toolkit was not designed for high bandwidth data, but supports sophisticated aggregation functions. On the other hand, TinyOS covers a wide span of the environmental information chain, although data are usually transferred to a separate host system for use. Only simple applications can be embedded directly on the wireless sensor platform due to low performance and stringent power requirements. Complete ubicomp systems could thus be built which sport some combination of all three systems: wireless sensing, high-bandwidth streaming data collection and event-based "contextual" information.

SCENARIO

The following scenario helps in framing user expectations and system requirements with regard to a specific application. The application described in the scenario (the "Digital Family Portrait") is loosely inspired by an ongoing research project at Georgia Tech[1] [14]. It was chosen as a paradigmatic example of a "peripheral" communication system used in a home setting, which nevertheless collects vast amounts of environmental data to function. Further, it also exposes manifold implications related to the social assumptions associated with private dwellings.

Scenario: The Digital Family Portrait

A digital picture frame, used to display photographs in a home, allows relatives to maintain contact, by projecting subliminal cues of a distant person's life in the domestic environment.

This is done by observing the distant person's activity through non-obtrusive sensors. Environmental information is gathered using wireless infrared sensors and video cameras and the application computes synthetic metrics for "activity" and "deviation from usual behavior", relative to stored information and pre-defined patterns. These metrics are then transmitted to the receiving picture frame which translates them into visible cues by employing peripheral display techniques. Far-away family members can thus be aware of the other's whereabouts, and notice if something unusual happens.

While the monitored user is usually happy that someone else is aware of her well being, one day she decides to leave home for some days without informing her relatives, and feels limited by the portrait. On the one hand, deactivating the application would represent an extraordinary event and would signal that the relatives watchful eye is unwelcome. On the other, simply leaving the application on would alarm the relative, as no activity at all would be reported at the other end.

DISCUSSION

The following discussion of requirements is structured around the environmental information lifecycle from collection to access. In parallel, the considered technical frameworks are assessed in light of these requirements.

Collection

Users ought to be able to choose whether personal information can be collected and stored by the systems they interact with. Choice not only is required by legislation, but is also considered precondition for acceptance. In most current applications, users can opt not to use some application (e.g. automatic toll payment systems) if they deem the benefit-risk tradeoff unacceptable, but not all choices are "all-or-nothing". For example, in the previous scenario the user might only want to **reduce** the spatial and temporal reach of the application, e.g. by limiting the sensing functions to a subset of the home environments, or by temporarily deactivating select reporting functions. This should not

[1] The authors have not collaborated with this research project.

entail an immediate notification to the other side, to avoid signaling that the application has been disabled.

In general, the choice of stopping the collection of information should not carry a disproportionate **cost** for the user, whether direct or indirect (in terms of provided services). In this case, where users might feel uncomfortable to explicitly hide their whereabouts from their relatives, the system should continue working, with reduced performance, even if the user does not consent to data collection.

How do these objectives impact the sensing infrastructure? A set of high-level requirements can be derived by this example, which influence not only the protection of user privacy but also application reliability and availability.

- **Systems should provide baseline functionality when environment information is not available** (e.g. in the case of the portrait, the application might allow the visualization to slowly decay, so that relatively short intermissions in the capture of information are not noticeable from the other side).
- **Disabling capture should be easy** or even automatic for the user and possible during execution of the application.
- Users should be able to **choose among different clearly defined capture "intensity levels"** (or profiles) to find the best compromise between captured information accuracy and provided services.
- Enabling and disabling collection of data should be **verifiable and enforceable**, in order to increase user trust and control.

Most of the mentioned requirements impact applications as well as sensing systems; for example, the first requirement implies that the capture system should be able to inform applications that data of interest is not available or is not being collected, so that applications can react accordingly. The second requirement implies that the infrastructure can turn capture on and off without disrupting applications, and vice versa. The last two requirements have strong implications on infrastructure: according to the third requirement, capture infrastructure should support modulating the amount and quality of captured information on user request. The fourth requirement suggests the need for implementing choice-enabling mechanisms directly in the capture framework to ease verification and to centralize enforcement of user policies. In fact, centralized enforcement is more effective in preventing malicious applications from disregarding user preferences and simplifies management – it is reasonable to assume that the sensing infrastructure is managed under the same authority as the surrounding physical environment – thus impacting also the second requirement.

The notification mechanisms built into InCA and the Context Toolkit would well support the first requirement, given their ability to register subscriptions to information of interest on behalf of applications and the possibility of issuing asynchronous queries on the data. However, neither infrastructure can reliably enforce the definition of capture profiles (third requirement), because the infrastructure lacks support for varying degrees of information "fidelity" or quality and, being open, any application can request the infrastructure to record full-blown environmental data.

The second and fourth requirement are also difficult to implement, because:

- access to the infrastructure functions is open and allowed to any application;
- the network traffic is not encrypted;
- the infrastructure cannot independently activate or deactivate capture.

TinyOS provides a low-level sensing infrastructure. While it could be possible to disable networked sensors from collecting data, more complex operating modes would be best implemented at a higher architectural level. The small size and unobtrusive nature of this hardware platform poses however a great challenge for complying with requirement 4, as the sensors are unable to signal their activation, except by lighting the three LEDs mounted on the boards or by sounding the buzzer, which is not an option in most deployments.

Incidentally, influential studies on the subject of video surveillance (e.g. [1]) have pointed out that in principle all data gathering should be commensurate to their purpose and the advantages for users and society. This has clearly a strong influence on environmental sensing design: does for example a remote monitoring application like the one described justify continuous observation of people in residential areas?

Communication

The sensing infrastructure which supports the described application, if not adequately secured, might provide easy leeway for misuse. Eavesdropping is clearly a security concern, as recurrent cases involving 802.11 and X10 networks suggest. Setting aside confidentiality issues, there are other ways malicious users could exploit wireless domestic networks, which are increasingly popular due to low installation costs and setup ease. For example, by hooking up unauthorized "sensors" which insert false or misleading sensed information into the system.

Securing such an environment will require not only users to authenticate themselves, but also to verify the identity and credentials of network components, both fixed and mobile, which cannot be guaranteed by securing physical wired connections.

- **Sensors should be able to perform authentication with each other and with the network**, to block unauthorized sensing devices to be covertly introduced in the network.

None of the analyzed infrastructures provides protection against such threats: building jammers or sniffers of envi-

ronmental information within the Context Toolkit or InCA is relatively easy, especially if the underlying TCP network is not secure (e.g. is on wireless medium). Any component can register with both infrastructures and provide fake sensor data to applications and for permanent storage. Neither infrastructure provides support for authenticating sensors and information providers and consumers.

TinyOS is currently being enhanced with symmetric encryption modules, although the open nature of the technology and the lack of asymmetric encryption required by PKI-like systems (due to performance problems) make strong authentication and trusted operation complicated.

The option of using trusted computing architectures for environmental sensors has been proposed, but is at the present stage difficult to carry out due to its high implementation costs and other concerns [9].

Storage

In order to provide the service described in the scenario, the system needs to store (even though temporarily) a certain amount of sensed environmental information. The type of information conveyed by the system through the subliminal communication channel requires to compute and compare current activity and movement patterns to previously captured data, and to keep historical records (due to exogenous requirements which we will not cite here).

Consider information collected using video cameras; anybody entering the observed dwelling would be subject to data collection. Respecting the basic privacy concerns of all individuals with relation to their "presence" in captured data might prove to be at least impractical. At a very basic level, the visitor might request that the information collected during his or her stay in the environment be deleted or amended.

Without considering the technical challenges (manual intervention to remove segments of collected video feeds based on temporal and spatial criteria), this requires applications also to adapt to a rather counter-intuitive "variable" recorded "reality". Information sensed in the environment turns from hard fact into a fluctuating construction. A potentially changing dataset requires the infrastructure and the applications to implement the following requirements.

- The sensing framework should **provide mechanisms to remove environment information from storage**, to comply with the transparency principles of data protection regulation. In any case, it should be possible to reset the state of the application when stored information is removed.
- This feature, in turn, requires applications to **not depend on the invariability of stored information**. On the contrary, applications should be able to cope with changing stored sensor information.

Both InCA and the Context Toolkit provide very limited modification and update functions, where at all. This is understandable due to the complexity of database consistency problems coupled with the experimental nature of these frameworks.

InCA stores separate data objects independently from each other, being only associated to attributes. This makes it relatively easy to edit and replace data objects within the store without breaking the coherency of the complete data store at the infrastructure level, granted that the specific format can be edited. At the *application* level, it is much harder to eliminate all traces from the stores for a specific data object, and it is possible to damage the captured data through editing (e.g. removing subsets of captured information), as the application which removes or edits data needs to understand all data formats involved in a specific sensing setup and the relationships between individual data objects.

The Context Toolkit only allows to modify information while this is in working memory (i.e. until shortly after collection). Once stored, sensed information cannot be altered within the database through the toolkit (although separate applications might be able to do this by circumventing the infrastructure). Both infrastructures' notification systems could be extended to support alerting applications that some part of the sensed information has changed.

TinyOS systems do not store vast amounts of information and are generally left unaffected by the issue.

One of the main issues with storage is that of managing the expiration (and disposal) of stored data. The expiration of collected environmental information (and thus its removal from storage) is a social concern even if specific privacy legislation is not in force [16]. It should be noted however, that even in jurisdictions where data protection legislation is in effect, the problems, technical and organizational, associated with data expiration have not yet been dealt with and researched in depth (notwithstanding promising work, e.g. [2]) as very few databases have reached their aging or contractual time limits. The "deterioration" of information [11] is an appealing option, but might be very complex to apply, and even conflict with existing legislation, which requires on the contrary to protect data integrity as long as the data exists.

Access

How large amounts of sensed information can be exploited is strongly dependent on a variety of details, which collectively define the "search properties" of the data, i.e. what techniques and what cost is associated to extracting from the dataset the answer to a specific query.

The above scenario exposes the substantial difference between two kinds of sensed information: high-volume, semantically sparse information such as streaming video and audio feeds, and low-volume, but highly synthetic information like the "activity metrics" which can be computed from the streaming data. The former is harder to search for specific occurrences (e.g. an individual entering a room). The latter is on the contrary a rich, compact representation

of the user activity and life, and lends itself to extensive search functions and complex queries (e.g. it would provide good information about the user's living patterns).

One might object that evolving technology will make searching vast A/V datasets increasingly easy, but, after all, the protection of privacy has always been a balancing act of the plausible with the desirable. While compact, easily searchable representations can be computed from high-volume data, the associated cost (not only in terms of computing power required, but also of time) represents an effective barrier against misuse. Based on the encoding, queries can simply be too expensive, or the response delay to long, to be useful. Infrastructure systems provide environmental information to a host of different applications, and are in a good position to enforce constraints on what data, and in what format, may be accessed.

Moreover, search properties on data also influence how data protection regulation can be complied with. In case a user legitimately requests that all data relating to her be deleted from a system, the degree to which it is possible to comply with the request depends on how expensive searching for that specific data is.

Designers should take into account these variables during development, by considering among others, the following design choices.

- **Define how datasets are to be accessed (accessibility analysis)** (e.g. time-based access vs. query-based search). This includes enforcing access control when applications try to circumvent the safeguards set by accessibility analysis.
- **Define what information is to be stored permanently** and what should be discarded.

In InCA, specific A/V media objects are accessed by evaluating related attributes. Since media objects cover an intrinsic time span, the attributes, implicitly, relate to the entire time span of the object.[2] Such time span is determined by the application and not by the infrastructure itself. Once a media object has been identified, it can be extracted from the system and searched in its content.

The Context Toolkit focuses on event-based information, and any item in the database can be used to perform search queries. Thus, all information stored in the database is potentially reachable by external applications. The infrastructure does not provide, on the other hand, methods to limit the kind of queries applications can perform on the data once it is stored.

TinyOS-based systems allow to store relatively small amounts of sensed data on the single sensor and data must be transmitted off the device by specific applications to be available to external uses. However, recent work [13] has proposed network query systems which allow to efficiently extract semantically rich data from a collection of devices using a powerful query language. This will require designers to analyze, similarly to the other infrastructures above, the accessibility to information in the network.

The scenario also hints at the need for user authentication. For example, many individuals living in the same environment (e.g. an assisted living facility) would require access control on a per user-basis to the stored data. Recognizing that explicit authentication mechanisms impact the seamless use of ubicomp applications, researchers have tried to build automatic authentication systems often based on sensing technology itself (e.g. through user identification based on a video recording).

- **Authentication functions be provided and enforced at the infrastructure level**. This is required to provide strong guarantees against uncontrolled access.

Authentication is only one precondition for secure operation; the home infrastructure of the scenario would also have to enforce much more complex policies (e.g. preventing users from using the collected video feeds to spy on each other). Again, none of the infrastructure systems analyzed provides user-level access control.

Finally, underlying the previous discussion about access control is a more substantial point. The described sensing system subverts our assumptions about space developed and codified over the centuries in the architecture of homes, public spaces, and working environments. The problem lies partially in the fracture between our daily experience and expectations and the capabilities of the technology. Users should be able to judge the risks and merits of specific applications intuitively; one method of achieving this would be to leverage our common understanding of the physical properties of space and architecture. This leads to the following, far-reaching requirement.

- **Sensing systems should enforce access rights based on physical attributes**, such as collection location and user presence.

As bathroom walls serve the purpose of avoiding others to look inside while the bathroom is in use, similar assumptions should be incorporated in technology. None of the sensing systems considered supports security policies of this kind at the infrastructure level.

CONCLUSIONS

The result of this work is twofold. A method for deriving security requirements has been proposed and tested; and a collection of security requirements has been derived for environmental sensing technology.

The method utilizes usage scenarios to identify security objectives for specific applications and compares the resulting requirements with existing technology, to understand how these systems could be enhanced to address them. The discussion of one such scenario has exposed a

[2] "Timed attributes" which contain temporal span information have been added to InCA to address this problem but are not fully implemented.

number of open issues with three sensing technologies and the relationship between user experience and the technical aspects of information security. The identified requirements, divided by information lifecycle stage are synthesized below.

Collection

- Applications should provide baseline functionality when environment information is not available.
- Disabling capture should be easy or even automatic.
- Systems should allow choice among different clearly defined capture "intensity levels".
- Enabling and disabling collection of data should be verifiable and enforceable.

Communication

- Sensors should perform authentication.

Storage

- Infrastructure should support removal of information from storage.
- Infrastructure support changes to stored information.

Access

- Designers should perform analysis of access to information based on the "search properties" of information.
- Designers should identify whether information is to be stored or discarded.
- Infrastructure should provide user authentication functions.
- Access policies based on physical attributes.

These security requirements are not exhaustive, and thus are intended to be used along with requirements derived by other means (iterating the same method over other scenarios or through other forms of requirements analysis). However, they provide information and insight which could be difficult to obtain through other means, and thus suggest that the method used can provide good results for rapidly assessing experimental infrastructure deficiencies and paths for enhancement, without the cost and delay of actual deployment.

Ongoing and future work will expand this avenue of research both by examining more scenarios and different infrastructures and by feeding the results back to the examined technologies in the form of enhancements to their design.

ACKNOWLEDGEMENTS

We would like to thank Khai Truong, Michael Covington, Anind Dey, and especially Colin Potts for the most interesting discussions and their support to this research project. This work was partially funded by the National Science Foundation through the Graduate Research Fellowship Program and by the Sam Nunn School of International Affairs at Georgia Institute of Technology.

REFERENCES

1. Article 29 Data Protection Working Party of the European Commission, Working Document on the Processing of Personal Data by means of Video Surveillance, 11750/02/EN WP 67, http://www.europa.eu.int, November 25 2002
2. Ashley, P., Schunter, M., Powers, C., From Privacy Promises to Privacy Management – A New Approach for Enforcing Privacy Throughout an Enterprise, ACM New Security Paradigms Workshop (NSPW), Virginia Beach VA, 2002
3. Bellotti, V., Sellen, A., Design for Privacy in Ubiquitous Computing Environments, in Proceedings of CSCW 93, Milan, Italy
4. Covington, M., et. al., A Context-Aware Security Architecture for Emerging Applications, in Proceedings of ACSAC 18, ACM Press
5. Dey, A.K., Salber, D. and Abowd, G.D., A conceptual Framework and a Toolkit for Supporting the Rapid Prototyping of Context-Aware Applications in the Human-Computer Interaction (HCI) Journal, Volume 16 (2-4), 2001, pp. 97-166
6. Garfinkel, S., Adopting Fair Information Practices to Low Cost RFID Systems, in Ubiquitous Computing 2002 Privacy Workshop
7. EU, Directive 97/66/EC of the European Parliament and of the Council of 15 December 1997 concerning the processing of personal data and the protection of privacy in the telecommunications sector
8. EU, Directive 2002/58/EC of the European Parliament and of the Council of 12 July 2002 concerning the processing of personal data and the protection of privacy in the electronic communications sector (Directive on privacy and electronic communications)
9. Fordahl, M., Critics: 'Trusted computing' threatens consumer freedom, Securityfocus, Nov 4 2002, http://www.securityfocus.com/news/1564
10. Hill, J., Szewczyk, R., Woo, A., Hollar, S., Culler, D.E., Pister, K.S.J., System Architecture Directions for Networked Sensors, in Proceedings of ASPLOS 2000
11. Jiang, X., Landay, J., Modeling Privacy Control in Context-Aware Systems, IEEE Pervasive Computing July-September 2002
12. Langheinrich, M., Privacy by Design - Principles of Privacy-Aware Ubiquitous Systems, in Proc. Ubicomp 2001, Springer-Verlag LNCS 2201, pp. 273-291, 2001
13. Madden, S., Franklin, M., Hellerstein, J., Hong, W., TAG: a Tiny AGgregation Service for Ad-Hoc Sensor Networks, 5th Annual Symposium on Operating Systems Design and Implementation (OSDI), December, 2002

14. Mynatt, E.D., Rowan, J., Craighill, S. and Jacobs, A., Digital family portraits: Providing peace of mind for extended family members, in Proceedings of the ACM Conference on Human Factors in Computing Systems (CHI 2001). Seattle, WA, ACM Press, pp. 333-340
15. OECD, Guidelines on the Protection of Privacy and Transborder Flows of Personal Data, 1980, http://www.oecd.org
16. Palen, L., Dourish, P., Unpacking "Privacy" for a Networked World, in Proceedings of CHI2003, ACM Press
17. SEMPER: Secure Electronic Marketplace for Europe, final report, LNCS 1854, Springer Verlag, Berlin, 2000, ISBN 3-540-67825-5
18. Rannenberg, K., Multilateral Security: A Concept and Examples for Balanced Security, ACM New Security Paradigms Workshop, Ballycotton, Ireland, 2001
19. Reichenbach M., Damker, H., Federrath, H., Rannenberg, K., Individual Management of Personal Reachability in Mobile Communication, Information Security in Research and Business; Proceedings of the IFIP TC11 13th International Information Security Conference (SEC97): May 1997, Copenhagen, Denmark, pp.163–74
20. Seigneur, J-M., Farrell, S., Jensen, C.D., Secure ubiquitous computing based on entity recognition, in Workshop on Security in Ubiquitous Computing, Ubicomp 2002, Göteborg, Sweden, 29th September 2002, http://www.teco.edu/~philip/ubicomp2002ws/
21. Supreme Court of the United States, Kyllo v. United States
22. United States, Health Insurance Portability and Accountability Act of 1997, http://www4.law.cornell.edu/uscode/
23. United States Department of Health, Education and Welfare, Records, Computers and the Rights of Citizens, Report of the Secretary's Advisory Committee on Automated Personal Data Systems, July, 1973, http://aspe.hhs.gov/datacncl/1973privacy/
24. Weiser, M., Some Computer Science Problems in Ubiquitous Computing, Communications of the ACM, July 1993.