Securing Shareable Life-logs

Reza Rawassizadeh
Institute of Software Technology and Interactive Systems
Vienna University of Technology
Vienna, Austria
Email: rrawassizadeh@acm.org

A Min Tjoa
Institute of Software Technology and Interactive Systems
Vienna University of Technology
Vienna, Austria
Email: tjoa@ifs.tuwien.ac.at

***Abstract*—Sharing life-log information in a social community has many advantages, both for the user and society. But sharing any type of personal information is a threat to privacy. In particular, life-log information requires higher security considerations since it may contains very sensitive information about the user such as biological information, location, communication logs, etc. In this paper, first we discuss the risks of sharing life-log information in a social community. Then we will introduce a sharing model, which can eliminate the sharing capability of a life-log information object, based on the expiration time. Furthermore, general security approaches which might decrease security related risks for life-log systems will be described.**

I. Introduction

Life-Logs are used to sense and record users' daily life events. Sensors are the core components of the life-logs and they are used to sense contextual information of the users. Sensors can be desktop applications which record desktop activities of the user such as MyLifeBits [1] or they can be pervasive devices such as SenseCam [2] which is a body mounted camera to take pictures from the user's context. Life-logs may benefit users in many ways, for example: memory augmentation [3], behavior learning [4], psychological studies [5], personalization [6], etc.

Users benefit from using social networking technologies by sharing their personal information with other users. For instance a user can share content, find new friends based on the shared content (common interests) and stay in contact with old friends. These technologies allow users to enrich content using different mechanisms such as ranking, tagging, commenting, etc. In simple terms, sharing enriches the content.

The number of Social Networking Sites' (SNSs) users is growing fast [7]. Additionally, SNSs which provide more sharing features such as Facebook or Twitter have gained more success, because the number of their users in increasing in comparison to the traditional SNSs. Berslin and Decker [8] predicted that social networking will go beyond ego surfing in the future. They introduced a social networking stack to let users share information beyond the SNS domain, e.g. desktop environment. Based on the prediction of Berslin and Decker users are interested in sharing all their digital property and sharing life-log information which is a type of users' digital property.

In another ongoing research [9] we study that sharing some part of a user's life-log dataset with a social community can provide novel benefits for the users, e.g. learning the social behavior of a group [10], matchmaking [11], socially-aware recommendation systems, historical studies and sousveillance [12] (sousveillance is not surveillance). But life-log information is highly privacy sensitive, therefore developers need to pay tighter consideration to the security and the privacy of the life-log system, especially when they provide sharing capabilities for their life-log tool. Sharing any type of personal information with a social community might be harmful. Allen [13] described that using life-logs with the existing privacy laws and policies does not set appropriate limits on unwanted usage of information in using life-logs. She described that there is a high potential of incivility, emotional blackmail, exploitation, prosecution and social control by government as a result using life-log tools. Strahilevitz [14] stated that the most private information consists of sensitive personal matters such as sexual encounters and bodily functions, sensitive medical information and knowledge of the owner's fundamental weaknesses. A life-log tool can sense and record this information, therefore, from a privacy perspective, a life-log dataset is a very sensitive object. The sensitivity of the life-log information shows the importance of securing life logging process.

Moreover, life-logs produce datasets, which consist of continuous streams of the sensor data. A Life-log dataset is a kind of a data stream [15] and, because the information flows to the life-log from multiple sources (sensors) in a continuous manner, there is no guarantee for the sequence of the received data and the sequence may vary over time, the dataset size is unbound. These facts show that a life-log sharing model must be different than the traditional information sharing models, thus we need to introduce a sharing model that considers the privacy of users while they are sharing their life-log information.

It is notable that the current social networking sites are not appropriate for hosting life-log information. They enable users to share information, but we cannot use them as life-log tools. The main usage of life-logs is within the private scope and not within the social scope, and SNSs by their very nature, only provide social networking features and they do not provide enough capabilities to host private information. Users share information in social medias services manually, and despite the existence of automatic sharing capabilities, they are designed based on manual user interactions with

the system and do not work automatically. Life-logs sense the contextual information of the user automatically and it is not feasible to rely on manual user inputs. Their information visualization or reflection style is another salient difference, which makes them not appropriate for hosting life-log information. Because they can host a restricted amount of information in compare to a life-log. SNSs can be used as an output channel for few life-log data, but they cannot host the whole life-log data to share it. To our knowledge there is no operational life-log which lets users to share their information in a social community. We propose this research before implementing a social network that can host life-log information, because once a technology is deployed in an operational phase it is hard to make significant ethical or privacy related changes.

First we describe the risks associated with sharing life-log information. Then, based on the identified risks, we propose a data model and sharing policy that tries to reduce risks. Afterwards general approaches for securing life logging processes will be proposed. We emphasize being general because we intend to make it useful for similar life logging systems, and thus we do not focus on a specific model or a specific tool.

The remainder of this paper is organized as follows. First related works which are discussing security in pervasive devices will be introduced. Next section will describe risks of sharing life-log information in a social community. Then a data model which considers the sharing policy a life-log information will be introduced. The proposed model is a conceptual definition. Afterwards security issues which are related to the architecture of a life-log tool will be described. At the end we conclude this paper and describe the future work.

II. Related Works

Life-logs read the contextual information of the users and collect them, this indicates that they are a subset of the context-aware applications. In addition life-log tools are not restricted to desktop applications. They can be carried by the user and they can get more information in the ubiquitous environments rather than the desktop environments. In this paper the focus is on discussing security issues of life-log systems. Therefore in this section we describe related works on the security of pervasive devices which deliver contextual information from the environmental surrounding of users.

O'Hara et al. [16] stated that life-log information can be shared or enhanced by integration or cross-reference with information from others. Therefore they introduced having two logging scope, one is the public scope and the other one is the private one. They suggested to storing the private life-log data in personal knowledge bases, while public information that users intend to share can be stored separately. Total Recall[17] is a life-log system that uses a microphone and camera in a necklace to record the daily life events of users. In order to keep the privacy of the life-log information they suggested to use a privacy bit for each piece of data, which is masked, with an authenticity bit. There bits change when a modification is performed on data objects. CASA (Context-Aware Security Architecture) [18] proposed a security model for context-aware applications, which provides adaptive security services that ensure information access only to the authorized users. Al-Muhtadi et al [19] proposed a context-aware security scheme for smart spaces. They described a smart space as an environment containing ubiquitous sensors and surrounding users with information-rich atmosphere. This work tried to bring security services in the background and remove user distraction. They introduced Cerberus as a service which integrates identification, authentication, context-awareness and reasoning. Chaudhari et al. [20] proposed a life-log tool which uses audio and video sensors to capture and record users' visions and communications. It combined a real time audio distortion and visual blocking to protect the privacy of the individuals captured in a life-log video. A Pitch-shifting algorithm was used to distort the audio data. Face detection, tracking and blocking algorithms had been used to obfuscate subjects' faces. Their focus was on specific types of sensors, but our works is a holistic approach which can cover general aspects of privacy requirements.

III. Risks of Sharing Life-log Information

First we need to identify risks of sharing life-log information in a social community, then based on the identified risks we can describe security issues related to sharing life-log information. Sharing personal information has risks and benefits; Losing privacy is the most important risk of sharing personal information. Unfortunately life-logs have a controversial history, such as a DARPA's lifelog project [21] which was canceled in 2004 because of criticism of the system's privacy implications. Allen [13] identified two important potential hazards of life-logs: pernicious memory and pernicious surveillance. In order to be able to provide a data model and identify security requirements that reduce the potential hazards, we describe risks here in a different classification system. We have focused on listing risks explicitly and we cannot argue that we have listed all potential risks. More risks may be identified, when a life-log tool is used in a real operational and commercial environment.

- *Surveillance*: The sharing of life-log information with society might be interpreted as a form of surveillance. Any form of surveillance, which limits our behaviors without our desire, is not acceptable. Surveillance has some potential disadvantages such as increasing the number of suspects who are not guilty [22]. Security agencies, governmental organizations, business organizations, criminals and other types of organizations or industries, which benefit from monitoring social activities, can misuse surveillance data. Existing laws and policies do not provide an appropriate limit on the unwanted use of personal information. In the U.S. CAELA (Communications Assistance for Law Enforcement) enforces communication technologies to allow government access and surveillance

[23]. Also, in March 2006, the European Union adopted a directive, which mandates that the content of electronic communication services will not be deleted (remain for not less than six month and not more than two years). Hence they can be used for marketing and provisioning purposes [24].

- *Memory Hazards*: It has been proved that life-log tools can assist human memory [3]. The life-log can record all life events, disregarding the content. It means a life-log can prevent individuals from forgetting their errors [25]. Individuals naturally try to distance themselves from their errors and misfortunes because, psychologically, they need to forget misfortunes. Life-logs could be harmful in this case and they can also cause pathological rumination for unipolar and bipolar depression [26], [13]. Exposure of personal mistakes to society might be more harmful than keeping them private. For instance children by their nature are weaker at bearing misfortunes, they can go through their parents life-log and remember a misfortune which happened in their family. Another problem can appear by exposing the mental problems of individuals, which could have a negative impact on group mentality in a team or a group of individuals sharing a common interest. To handle the memory hazards of life-logs Dodge and Kitchen [25] suggested that the life-log should forget like real memory, based on the Scharter's six forms of forgetting [27].
- *Long term Availability of Personal Information*: The ideology and personality of an individual can change over time. Besides, an individual's life style may change over time. Sharing life-log information with society can be a permanent record of our mistakes. For instance, imagine a group of teenagers gathered at a party and posing for the camera in an embarrassing way. These pictures go online and every body can see them. At that time there is no problem. 30 years later one of those teenagers is now going to participate in an important political campaign, these pictures could harm his career. These problems are not only relevant to life-log information, other online traces of the individuals can cause him to suffer. Taking another example, consider a writer who has changed his mind over time and disagrees with his past opinions. In the era of the internet the chance of removing his old ideas from public view is very low.
- *Stealing Life-log Information*: Sharing life-log information increases the chance of loss or theft. This risk can be very harmful for the person (victim). Life-log can sense and record sensitive private information such as sexual encounters and bodily functions, therefore a life-log's dataset is a very sensitive object from a privacy point of view.

The described risks indicate that life-log could be harmful as well as being useful. With the exception of memory hazards all other risks are related to the sphere of privacy. Therefore, any associated data models for life-log tools must carefully manage privacy related issues.

IV. Data Model and Sharing Policy

We are living in a spatio-temporal world. This means that all of our life events except dreams happen in a specific location and at a specific date-time. Based on the current available technologies it is not always possible to sense the location, because location sensors such as GPS do not function in every environment. For instance GPS can not work indoors. There are other approaches such as A-GPS (Assisted GPS) to solve this problem, but they are not always able to sense location and they are imprecise. On the other hand most operating systems have date-time which is accessible as long as the target device has not been turned off. This means most devices with computing capabilities can provide timestamps. Therefore we conclude that date-time is a necessary field for any life-log record and all life-log information objects will be stored with the timestamp.

Life-log data types vary based on the sensor output. Data can be a text or a binary object such as a movie or a picture. We assume that all records of a life-log dataset have a date-time without considering the data type of the record. Life events can be recorded in a continuous-time manner or a discrete-time manner. Audio files and video files are examples of continuous data. In simple terms, if there is a start and end time for an event, then it is a continuous-time event but when there is a unique time, it is then a discrete-time event. Our proposed data model is independent from types of life event (discrete or continuous).

In order to have a flexible security model, a security policies is been defined based on the sensor, and group of users who can access the specific user's sensor information. An access scope will be defined for each life event. As previously described, each life event is a record in the life-log dataset. Atomicity is the main property of a life event. The dataset composed from a set of near infinite" number of life event data objects, which come from heterogeneous sensors. The dataset size is near infinite because the user's life is ongoing and during his life the dataset size increases. In addition, there are a wide variety of sensors that can be used by the life-log which provide different data types with different structures. The proposed model does not have any dependency on a sensor's data type or data structure. In other words, there should not be any boundaries between the access scope definition and the sensor structure. We provide a conceptual definition of the model, which enables users to share their life-log information. It is not dependent on technology and a life-log dataset can be in any data format such as XML, JSON , or RDBMS. Life-log data is date-time (timestamp) dependent. Therefore, we use timestamp data as a mandatory field for each life event record. It is notable that in the life-log domain nobody other than the owner of the dataset is able to manipulate the data and the only access right is "read", nor any other data manipulation rights.

We assumed that each life event is a data entity or record in the dataset. The Life-log dataset of a person P is represented

by *L(P)* and $L(P) = \{E_1, E_2, ...E_n\}$. The life-log data entity E is a 3-tuple $< D, T, S >$, where *T* is the timestamp, which can be continuous or discrete; if continuous, it will be the start timestamp and the end timestamp. D is the information object and it can be binary data e.g. image, audio or textual data e.g. GPS location, microblog content, etc. S is a 2-tuple $< A, E >$, where A is the access scope and E is the access expiration date. A can be *Private*, *Public* and *Friend*. *Private* means that the data entity is not shared and nobody other than the owner can access this information object and E is null. *Public* means that everybody in this social domain can access this information object and no access limitation has been defined for it but E can not be null and there an expiration date should be defined for that data object. *Friend* defines users who can access this information object. The user can define which user(s) or group of users from the social domain can access this object and E must be defined for access expiration date. In this case $A = \{P_1, P_2, ...P_n\}$, which is a finite set of users or group of users who can access the subject's information objects. How to extract the friend list is not within the scope of this paper. Developers or end users can define it based on their interpretations of the system.

Figure 1 shows that User A created an data object at time

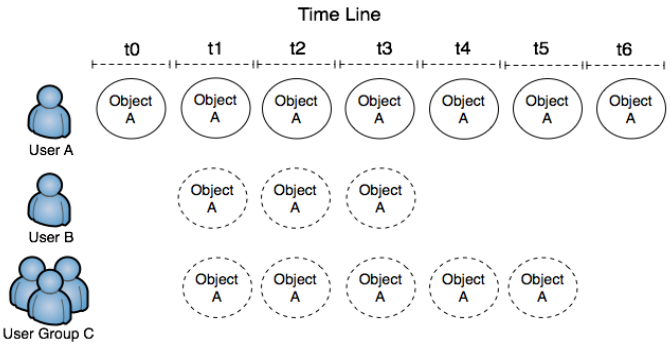

Fig. 1. Access Expiration Definition for a Shareable Information Object.

t0, then shares this data object at time t1 with User B and a group of Users (User Group C). User B's access to data object A will be expired at time t3 and Users in Group C can not access to this object after time t5.

Please note that only one access right that will be granted to other users is the "read" right. Unlike other access control systems we do not have other permissions such as delete, update, etc. It is the authorization process and not the authentication process, because only Read can be granted to the user. It is tedious for the user to specify the scope for each data object, therefore in the implementation model he can grant access to a user or a group of users for a group of objects, based on time. For instance all videos from the beginning of March up to the end of April can be marked to be shared in a family scope which is a group of users.

This model is a conceptual approach to enable the user to share his life-log's information in a social community. It is designed to be independent of implementation technologies and it is flexible enough to allow developers to manipulate data entities, but we think these three fields are necessary components for each data entity. We argue that this model tries to reduce risks based on identifying access scope for each information object in the life-log dataset and defining access expiration policy. Implementation of the model is not within the scope of this paper.

V. Architecture Related Security Recommendations

During the implementation phase, architecture based security issues of the system must also be taken into account. This section discusses the security considerations of the system in the architecture of life-log tools. First we describe the life logging process in general and introduce life-log components. Then, we identify and describe general security considerations based on the life-log components. We suggest that developers should address the security issues during the design phase of a life-log tool.

It is notable that our focus is first on securing the life-logging process, and then we discuss the security issues of sharing life-log information. Therefore these security recommendations can be used for a life-log tool without sharing capabilities as well.

Usually, a life logging process has three stages and each stage requires specific security considerations. The first stage is sensing the information from the user environment with sensors; the second stage is collecting the sensed information; and the third stage enables users to browse and retrieve information from their life-log dataset. Figure 2 shows the generic life-log component architecture. There we have highlighted the parts of a life-log system that need to be secure.

Users should be able to define what information object they intend to collect. Users might need to configure sensors in order to set their configuration parameters such as sensing interval, etc. The first stage connects the life-log system to sensors and reads sensor data. At this stage two things might need to be secure. First some sensors might require authentication, second if the sensor data contains sensitive information data transmission from the sensor to the life-log tool should be secure too e.g. encrypted data. We suggest providing dynamic security modules for the sensing stage since sensors might be added or removed dynamically to the life-log tool, therefore a non-centric dynamic security module for each sensor increases the flexibility and scalability of the life-log tool.

The second stage is to collect the sensed information in the life-log device. This stage creates a dataset of the life-log information. The dataset contains a set of life-log records. Data that comes from the sensors is mostly raw data and in order to enable users to browse and access them, some changes have to be made to the raw data. Changes might include annotation, aggregating sensors' data, migrating data from one format to another format, etc. During the collection phase developers should consider the third party tools that they are using to

change data. For instance a security threat might be the use of an annotation engine from a third party which sends users' information to that third party.

The third stage is storing the life-log data. Here storages (storing devices) which host life-log information should be secure. We suggest maintaining data in encrypted format if are intended to be stored as files, or if they will be stored in a database, designers should consider to define appropriate access rights on the database.

Life-log devices are pervasive devices such as mobile phones, but they might be desktop applications too. Unlike desktop computers, pervasive devices are prone to loss or damage [28], hence they are not capable of hosting personal information. Therefore, life-logs should maintain their data on reliable storage media such as the personal computer of the user. If a life-log tool does not use a pervasive device then there is no need to have a local storage and data can be stored directly on a reliable storage media. But life-logs usually contain at least a pervasive device. Transferring data from a pervasive device to a reliable storage media might not be a standard stage for a life-logging process, but it is normally required. Communications and data transfers are very sensitive from a security point of view and sharing information in a social community requires communication. This demonstrates that during the communication, data should be transfer encrypted. Securing the connection can be done by the transport layer security (TLS). However additional to TLS, using a digital signature has been suggested as an additional way to secure data transfer.

Current SNSs are based on client-server architecture and, because they own servers, they have access to users' information and they probably own the users information. We suggest sharing the life-log information with peer-to-peer (P2P) SNS such as a PeerSoN [29] social network model. We do not delve into the details of securing P2P social networks because it is not within the scope of this paper, but P2P social networks have a major advantage over traditional social networks. In a traditional SNS users upload their personal information on the service provider site, but in a P2P SNS users keep their information local. Hence, the chance of misusing users information is reduced and there will be less privacy risks.

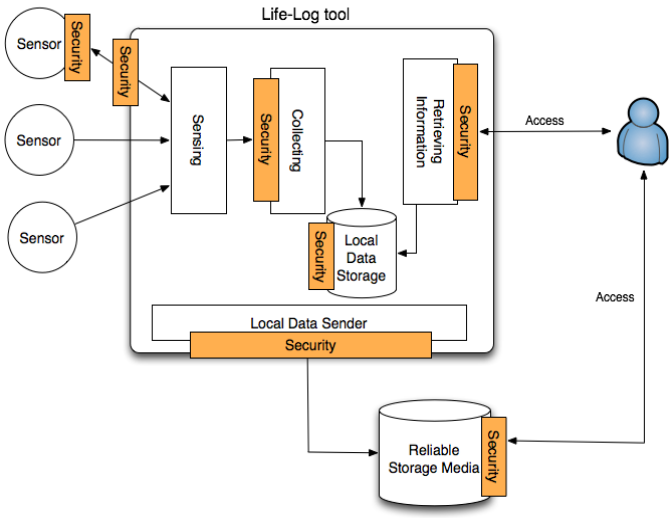

Fig. 2. Security Requirements in a Generic Life-log Architecture.

A P2P SNS requires the provision of secure access to the shared life-log information. Here secure means to perform the authentication and authorization on the users who intend to access to a life-log information. On the other hand communication between nodes in a P2P SNS should be secure. As it has been explained before, communication and data transfer are very sensitive.

During the design phase of secure modules, other design considerations should be taken into account e.g. designers should consider that, unlike desktop applications, pervasive applications authentication and authorization processes should not to be intrusive [19], [18]. Using the standard RBAC (Role Based Access Control) model for accessing the information is an appropriate method of limiting unwanted access, because a user can chose a user or group of users and grant them the appropriate access. Each access will be granted with an expiration timestamp as it has been explained in the "Data Model and Sharing Policy" section of this paper.

VI. Conclusion and Future Work

In this research we have introduced the risks of sharing life-log information. Based on the identified risks, we proposed a sharing model and security recommendations in order to reduce risks. Our suggestions tended to be general in order to make them useful for any life-log tools with sharing capabilities. On the other hand in our sharing model we have introduced a sharing expiration policy, which might help users benefit from sharing their information by removing old shared objects to reduce sharing related risks. As a future are of research, we might consider implementing this model on a real life-log tool or using current social networks as an output channel for sharing life-log information. Although they are not designed to host life-log information, they can assist us to study consequences of sharing life-log information in a real operational environment.

References

[1] J. Gemmell, G. Bell, and R. Lueder, "MyLifeBits: a personal database for everything," *Communications of the ACM*, vol. 49, no. 1, p. 95, 2006.

[2] S. Hodges, L. Williams, E. Berry, S. Izadi, J. Srinivasan, A. Butler, G. Smyth, N. Kapur, and K. Wood, "SenseCam: A retrospective memory aid," *UbiComp 2006: Ubiquitous Computing*, pp. 177–193.

[3] A. J. Sellen, A. Fogg, M. Aitken, S. Hodges, C. Rother, and K. Wood, "Do Life-Logging Technologies Support Memory for the Past?: An Experimental Study Using Sensecam," in *CHI '07: Proceedings of the SIGCHI conference on Human factors in computing systems*, 2007, pp. 81–90.

[4] B. P. Clarkson, "Life Patterns: Structures of wearable sensors," Ph.D. dissertation, Massachusetts Institute of Technology. Dept. of Electrical Engineering and Computer Science, 2002.

[5] H. T. R. Ladd Wheeler, "Self-Recording of Everyday Life Events: Origins, Types, and Uses," *Journal of Personality*, pp. 339–354, 1991.

[6] A. Fitzgibbon and E. Reiter, "Memories for life: Managing information over a human lifetime." *UK Computing Research Committee Grand Challenge proposal*, vol. 22, pp. 13–16, 2003.

[7] "Social Network Sites: Definition, History, and Scholarship," *Journal of Computer-Mediated Communication*, vol. 13, no. 1-2, November 2007.

[8] J. Breslin and S. Decker, "The Future of Social Networks on the Internet: The Need for Semantics," *IEEE Internet Computing*, vol. 11, pp. 86–90, 2007.

[9] R. Rawassizadeh, "Toward Sharing Life-log Information with Society," 2010, in Submission Process.

[10] N. Eagle and A. S. Pentland, "Reality mining: sensing complex social systems," *Personal Ubiquitous Computing*, vol. 10, no. 4, pp. 255–268, May 2006.

[11] N. Eagle and A. Pentland, "Social Serendipity: Mobilizing Social Software," *IEEE Pervasive Computing*, vol. 4, pp. 28–34, 2005.

[12] S. Carlson, "On The Record, All the Time," *Chronicle of Higher Education*, 09 2007.

[13] A. Allen, "Dredging up the Past: Lifelogging, Memory, and Surveillance," *The University of Chicago Law Review*, vol. 75, no. 1, pp. 47–74, 2008.

[14] L. Strahilevitz, "A Social Networks Theory of Privacy," *The University of Chicago Law Review*, vol. 72, no. 3, pp. 919–988, 2005.

[15] B. Babcock, S. Babu, M. Datar, R. Motwani, and J. Widom, "Models and Issues in Data Stream Systems," in *PODS '02: Proceedings of the twenty-first ACM SIGMOD-SIGACT-SIGART symposium on Principles of database systems*, 2002, pp. 1–16.

[16] K. OHara, M. Tuffield, and N. Shadbolt, "Lifelogging: Privacy and Empowerment with Memories for life," *Identity in the Information Society*, vol. 1, no. 1, pp. 155–172, 2008.

[17] W. Cheng, L. Golubchik, and D. Kay, "Total Recall: Are Privacy Changes Inevitable?" in *CARPE'04: Proceedings of the the 1st ACM workshop on Continuous archival and retrieval of personal experiences*, 2004, pp. 86–92.

[18] M. J. Covington, P. Fogla, Z. Zhan, and M. Ahamad, "A Context-Aware Security Architecture for Emerging Applications," in *ACSAC '02: Proceedings of the 18th Annual Computer Security Applications Conference*, 2002, pp. 249–260.

[19] J. Al-Muhtadi, A. Ranganathan, R. Campbell, and M. D. Mickunas, "Cerberus: A Context-Aware Security Scheme for Smart Spaces," in *Pervasive Computing and Communications, IEEE International Conference on*, 2003, pp. 489–496.

[20] J. Chaudhari, S. Cheung, and M. Venkatesh, "Privacy Protection for Life-log Video," in *SAFE'07: IEEE Workshop on Signal Processing Applications for Public Security and Forensics*, 2007, pp. 1–5.

[21] N. Shachtman, "A Spy Machine of DARPAs Dreams," http://www.wired.com/print/techbiz/media/news/2003/05/58909, 2003, last Accessed =[1-Apr-2010].

[22] D. Lyon, *Surveillance after September 11*. John Wiley Sons, 2003.

[23] "CALEA," http://www.fcc.gov/calea, September 2005, [Accessed 10 October 2009].

[24] "DIRECTIVE 2006/24/EC OF THE EUROPEAN PARLIAMENT AND OF THE COUNCIL," *Official Journal of European Union*, March 2006.

[25] M. Dodge and R. Kitchin, "'Outlines of a world coming into existence': pervasive computing and the ethics of forgetting," *Environment and Planning B: Planning and Design*, vol. 34, no. 3, pp. 431–445, 2007.

[26] K. Addis, M.E. andCarpenter, "Why, why, why ?:Reason-Giving and Rumination as Predictors of Response to Activation and Insight-Oriented Treatment Rationales," *Journal of Clinical Psychology*, vol. 55, no. 7, pp. 881–894, 1999.

[27] D. Schacter, *The Seven Sins of Memory. How the Mind Forgets and Remembers*. New York: Houghton Mifflin, 2001.

[28] M. Satyanarayanan, "Fundamental challenges in mobile computing," in *PODC '96: Proceedings of the fifteenth annual ACM symposium on principles of distributed computing*, 1996, pp. 1–7.

[29] S. Buchegger, D. Schiöberg, L.-H. Vu, and A. Datta, "PeerSoN: P2P Social Networking: Early Experiences and Insights," in *Proceedings of the Second ACM EuroSys Workshop on Social Network Systems*, 2009, pp. 46–52.