Architecting Pervasive Computing Systems for Privacy: A Survey

Roberto Speicys Cardoso, Valérie Issarny

HAL Id: inria-00415925

https://inria.hal.science/inria-00415925

Submitted on 11 Sep 2009

Architecting Pervasive Computing Systems for Privacy: A Survey

Roberto Speicys Cardoso   Valérie Issarny

INRIA Rocquencourt

78153 Le Chesnay, France

roberto.speicys_cardoso@inria.fr

valerie.issarny@inria.fr

Abstract

In pervasive computing systems, a higher number of interactions will be mediated by computers, amplifying the menace to privacy. Privacy protection in pervasive environments is still a big issue, despite the growing number of works on this subject as evidenced by this survey. In this paper, we propose a taxonomy for privacy invasion attacks, classify existing privacy enhancing technologies according to the protection provided for those attacks, and introduce a service-oriented privacy-enhanced architecture for pervasive computing.

1 Introduction

Each culture deals with privacy in a different way. Those differences may appear to be subtle when comparing civilizations with similar cultures, but they become bigger when we take into account more isolated cultures that were less influenced by the western perspective on privacy. In some tribes, for instance, the concept of a private space does not exist and it is common to find whole families living in the same house without walls. However, while one can question how much privacy is necessary on a society (or even if it is needed at all), it is clear that citizens have the right of a private space and this right must be protected, so that they can *choose* to reveal personal data instead of being *forced* to disclose it.

Controversy exists on the meaning of privacy, as this word is used in different scenarios and contexts whenever people want to justify the confidentiality of an information. In this paper we use a pragmatic definition: *privacy is control over information disclosure*. As a consequence, a privacy-aware system allows for conscious disclosure decisions. A privacy invasion occurs when information regarding an entity is disclosed without its explicit consent.

Even this simplified definition presents many challenges, since it is yet unclear which is the best way to control information disclosure, how to identify and manage it, and when an information is sensitive. At times, a seemingly innocent transaction may be highly invasive when correlated with other data. Future pervasive environments have the potential of collecting and correlating a greater quantity of data, thus increasing the menace to privacy posed by computer systems. This risk is closely related to the system architecture, as it defines how data flows from users to applications. In our vision, pervasive computing systems and applications will follow the service-oriented architecture (SOA) paradigm as it is well suited to handle the heterogeneity, dynamism and mobility inherent to those environments.

A generic pervasive service-oriented application can be described as consisting of a *mobile client* using some kind of *contextual information* to access a *service*. Client and service provider roles are not fixed, and contextual information is used to customize the service to the user's needs. Every other entity not directly related to the service access is considered as a *third party*. According to our definition of privacy, client and service providers can be victims of privacy invasions performed by each other, or by a non-related third party. In this scenario, information regarding a service access, including the service existence, should only be available to parties involved with the transaction. In addition, they should disclose only the minimal required information to complete a given task. Whenever these conditions do not hold, privacy invasions may happen according to the strategies described on Table 1. Categorization of privacy protection techniques according to the attacks they intend to avoid can help system designers to understand their goals and their common properties.

2 Privacy Protection Technologies

We classify privacy technologies according to the protection they provide against the attacks we identified, namely: protecting the service access content, protecting the existence of a service access, protecting information disclosure and protecting information usage.

IEEE
COMPUTER
SOCIETY

Attacker	Privacy Invasion Strategy
Third Party	Learn about the occurrence of a service access
	Learn about the contents of a service access
Client	Obtain more information than required for the service access
	Use obtained information for a different purpose
Service Provider	Obtain more information than required for the service access
	Use obtained information for a different purpose

Table 1. Privacy invasion strategies

Protecting the content of a service access   This problem is the same as avoiding leakage of data on a *legitimate* communication channel [19]. This is usually obtained by using cryptographic techniques such as public-key encryption or secure communication protocols. Although some issues still exist (key management notably being one of them), available cryptosystems provide an acceptable protection against this type of privacy invasion strategy.

Protecting the existence of a service access   Covert channels [19] can be used to reveal information about the occurrence of a service access. Those channels can either be avoided or have their quality degraded to impede practical information leakage. Design guidelines [18, 20] and patterns [8] can help software designers to avoid covert channel creation at design time, but are not sufficient to eradicate covert channels since some of them are unavoidable. For instance, IP network packets must have a source address and a destination address to allow for packet routing. This information, however, was never meant to be used to reveal information about the user's activity. Many systems tried to reduce the bandwidth of this channel by mixing messages during routing [7, 26, 10] or introducing noise by generating fake packets to hide real communication [12, 28].

Protecting information disclosure   Clients and service providers must disclose only data that is strictly necessary during a service access. Decisions about which data has to be disclosed, for what purposes and how it should be disclosed can be agreed upon beforehand, through the use of labeling protocols such as P3P [25] and PawS [21]. Personal and contextual data must also be modified to reveal only data required by the transaction and nothing else. Three techniques can be used to adjust personal data resolution: modification, multiplication and generalization. Modification substitutes the requested information for a different information, multiplication replaces a requested information with a set of data where the requested information is contained and generalization substitutes the requested information with another less specific data that could correspond to different values of the requested information. Many techniques were developed to change the resolution of location data, modifying [13, 2], generalizing [1] or multiplying them [11, 16]. Images are also sensitive data, and there are techniques that enables modification [9] and generalization [15, 3, 29, 22] of specific kinds of images (such as the face of a person recorded by a webcam). Finally, identity information can also be generalized to avoid connections between data and individuals [6, 27, 4].

Protecting information usage   This is one of the biggest challenges to privacy in pervasive computing. Even though labeling protocols such as P3P and PawS can be used to specify usage restrictions, they can do little to enforce those constraints. Privacy Rights Management (PRM) [17] was proposed as an approach inspired by Digital Rights Management (DRM) to protect personal data usage.

The main shortcoming of those technologies is that they are totally disconnected. The privacy benefits provided by one solution disappear when another technology is used concurrently. For instance, location-privacy strategies can reveal the source of a message even if communication anonymization protocols are in place. Most of the time it is impossible to combine two techniques due to conflicting requirements. Software architectures for privacy are necessary to provide a common framework for privacy technologies, such that they can be combined to effectively protect personal data at all system levels. However, it is only recently that some architectures were proposed to control use of personal data [23], access and storage [14] and identity [5].

3   Towards a Pervasive Privacy Architecture

We envision future pervasive computing environments as open systems, where users will be able to spontaneously collaborate by creating ad-hoc networks and discovering and composing services dynamically. In such environments with heterogeneous devices and multiple administrative domains, usage control is hard to obtain and might be even unfeasible. We thus propose an architecture that provides

mechanisms to control disclosure of personal information and enable users to reveal only the information strictly necessary to perform a service access. This can reduce the effects of usage violations of personal data executed by malicious third parties.

The use of SOAs in environments as the one described above creates new threats to privacy. Service information such access, discovery and published descriptions are sensitive and must be protected, since they can provide information about an user's activity to an external observer. Moreover, contextual information may be used in service publishing and location to improve service discovery performance [24] increasing the sensitiveness of service information. To protect it, it is necessary to enforce access control rules not only during service access, but also over service publishing, discovery and composition.

Contextual data disclosure must also be controlled, since it is rich and may reveal personal information. Clients and service providers must agree on the resolution required for the context data, as well as other details such as periodicity. Before disclosure, context data resolution must be automatically adjusted to the level previously negotiated to avoid transmission of sensitive data irrelevant for the transaction. Access to this data must also be controlled through a scalable mechanism capable of managing a high number of objects, enforceable whenever data is accessed or exchanged, and that does not rely on a fixed infrastructure that may not be available on pervasive environments.
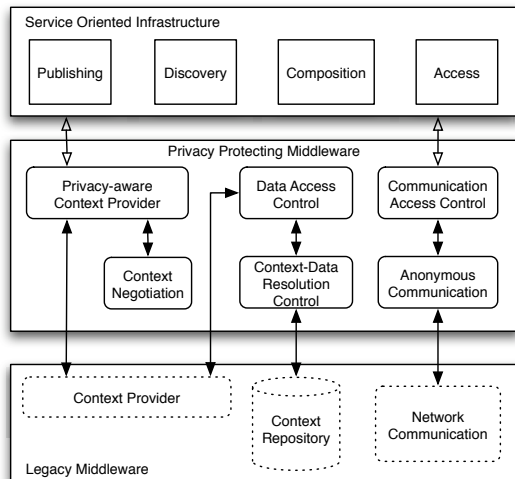

Figure 1. A Privacy Protecting Middleware

Our architecture is designed as a complement for existing service-oriented context-aware pervasive architectures. It leverages existing components of such architectures and introduces privacy-related modules that interact to provide multi-level protection of personal data. Figure 1 shows the architecture design. The legacy middleware layer contains modules that already exist on traditional context-aware pervasive architectures. The service-oriented infrastructure layer contains modules to publish, discover, compose and access services. Our middleware sits in between those two layers, mediating requests for contextual data and communication. It enforces access control whenever data is accessed and transmitted, and adapts contextual information to protect the user's personal data.

Before any request for contextual data, the Context Negotiation module defines its characteristics so that only information strictly required for the transaction is disclosed. After that, whenever context data is requested, the Context-Data Resolution Control module modifies it to comply with the requirements previously negotiated. A Data Access Control module verifies if access to data is allowed taking into account contextual information such as location or presence of other users.

All network communication must be allowed by the Communication Access Control component. While the Data Access Control module performs decisions based on the content of requests, its communication counterpart deals mostly with network-related issues. Before finally reaching the network, messages can be anonymized by the Anonymous Communication component. The communication anonymity level can be adapted according to the message content: data that is highly sensitive may require stronger anonymization than public data, for instance.

4 Conclusion

This work is a first step on integrating privacy protection technologies on a service-oriented architecture for pervasive computing. Right now we are working on detailing the components of the architecture, their functions and interactions. Privacy architectures are crucial for the definition of a software design where technologies do not overlap, do not have conflicting requirements and cooperate to provide multi-level privacy protection and effectively protect the user's personal data.

References

[1] J. Al-Muhtadi, R. Campbell, A. Kapadia, M. Dennis Mickunas, and S. Yi. Routing through the mist: Privacy preserving communication in ubiquitous computing environments. In *ICDCS'02 - 22nd IEEE International Conference*, pages 74–83, 2002.

[2] A. R. Beresford and F. Stajano. Location privacy in pervasive computing. *IEEE Pervasive Computing*, 2(1):46–55, 2003.

[3] M. Boyle, C. Edwards, and S. Greenberg. The effects of filtered video on awareness and privacy. In *Proceedings of ACM CSCW '00*, pages 1–10, 2000.

[4] S. A. Brands. *Rethinking Public Key Infrastructures and Digital Certificates: building in privacy*. The MIT Press, 2000.

[5] J. Camenisch, a. shelat, D. Sommer, S. Fischer-Hübner, M. Hansen, H. Krasemann, G. Lacoste, R. Leenes, and J. Tseng. Privacy and identity management for everyone. In *Proceedings of DIM '05*, pages 20–27, 2005.

[6] D. Chaum and E. Van Heyst. Group signatures. In *Eurocrypt '91*, pages 257–265, 1991.

[7] D. L. Chaum. Untraceable electronic mail, return addresses, and digital pseudonyms. *Communications of the ACM*, 24(2):84–90, 1981.

[8] E. S. Chung, J. I. Hong, J. Lin, M. K. Prabaker, J. A. Landay, and A. L. Liu. Development and evaluation of emerging design patterns for ubiquitous computing. In *Proceedings of DIS '04*, pages 233–242, 2004.

[9] J. L. Crowley, J. Coutaz, and F. Bérard. Perceptual user interfaces: things that see. *Communications of the ACM*, 43(3):54–64, 2000.

[10] R. Dingledine, N. Mathewson, and P. Syverson. Tor: The second-generation onion router. In *Proceedings of the 13th USENIX*, pages 303–320, 2004.

[11] M. Duckham and L. Kulik. A formal model of obfuscation and negotiation for location privacy. In *Proceedings of PERVASIVE 2005*, pages 152–170, 2005.

[12] M. J. Freedman and R. Morris. Tarzan: a peer-to-peer anonymizing network layer. In *Proceedings of ACM CCS '02*, pages 193–206, 2002.

[13] M. Gruteser and D. Grunwald. Anonymous usage of location-based services through spatial and temporal cloaking. In *ACM MobiSys2003*, pages 31–42, 2003.

[14] J. I. Hong and J. A. Landay. An architecture for privacy-sensitive ubiquitous computing. In *ACM MobiSys '04*, pages 177–189, 2004.

[15] S. E. Hudson and I. Smith. Techniques for addressing fundamental privacy and disruption tradeoffs in awareness support systems. In *Proceedings of ACM CSCW '96*, pages 248–257, 1996.

[16] H. Kido, Y. Yanagisawa, and T. Satoh. An anonymous communication technique using dummies for location-based services. In *Proceedings of the IEEE ICPS'05*, pages 88–97, 2005.

[17] L. Korba and S. Kenny. Towards meeting the privacy challenge: Adapting DRM. In *ACM CCS-9 Workshop*, pages 118–136, 2003.

[18] S. Lahlou and F. Jegou. European disappearing computer privacy design guidelines v1.1. Technical Report EDC-PG 2003, Ambient Agoras, 2004.

[19] B. W. Lampson. A note on the confinement problem. *Communications of the ACM*, 16(10):613–615, 1973.

[20] M. Langheinrich. Privacy by design - principles of privacy-aware ubiquitous systems. In *UbiComp 2001*, pages 273–291, 2001.

[21] M. Langheinrich. A privacy awareness system for ubiquitous computing environments. In *UbiComp 2002*, pages 237–245, 2002.

[22] C. Neustaedter and S. Greenberg. The design of a context-aware home media space for balancing privacy and awareness. In *UbiComp 2003*, pages 297–314, 2003.

[23] A. Pretschner, M. Hilty, and D. Basin. Distributed usage control. *Communications of the ACM*, 49(9):39–44, 2006.

[24] P.-G. Raverdy, O. Riva, A. de La Chapelle, R. Chibout, and V. Issarny. Efficient context-aware service discovery in multi-protocol pervasive environments. In *Proceedings of MDM'06*, page 3. IEEE Computer Society, May 2006.

[25] J. Reagle and L. F. Cranor. The platform for privacy preferences. *Commununications of the ACM*, 42(2):48–55, 1999.

[26] M. K. Reiter and A. D. Rubin. Crowds: anonymity for Web transactions. *ACM Transactions on Information and System Security*, 1(1):66–92, 1998.

[27] R. L. Rivest, A. S., and Y. Tauman. How to leak a secret. In *Proceedings of ASIACRYPT '01*, pages 552–565, 2001.

[28] R. Sherwood, B. Bhattacharjee, and A. Srinivasan. P5: A protocol for scalable anonymous communication. In *Proceedings of IEEE SP '02*, page 58, 2002.

[29] Q. A. Zhao and J. T. Stasko. Evaluating image filtering based techniques in media space applications. In *Proceedings of ACM CSCW '98*, pages 11–18, 1998.