Privacy Support for Sensitive Data Sharing in P2P Systems

Mohamed Jawad, Patricia Serrano-Alvarado, Patrick Valduriez, Stéphane Drapeau

Privacy Support for Sensitive Data Sharing in P2P Systems

Mohamed Jawad[1], Patricia Serrano-Alvarado[1], Patrick Valduriez[2] and Stéphane Drapeau[3]

[1]LINA, Université de Nantes
2, Rue de la Houssinière
44322 Nantes, France
Name.Lastname@univ-nantes.fr

[2]INRIA and LIRMM
161 rue Ada
34095 Montpellier, France
Patrick.Valduriez@inria.fr

[3]Obeo
7 Boulevard Ampère
44481 Carquefou, France
Stephane.Drapeau@Obeo.fr

Résumé

Les applications partageant des données sensibles peuvent bénéficier des avantages des systèmes P2P (Peer-to-Peer) mais uniquement si la confidentialité est préservée. Dans nos travaux antérieurs, nous avons proposé PriMod, un modèle de confidentialité pour partage de données P2P qui combine le contrôle d'accès basé sur les objectifs, la confiance et le chiffrement. Nous avons également proposé PriServ, un service basé sur PriMod, implémenté sur une table de hachage distribuée (DHT). Cette démonstration montre le prototype PriServ et souligne les bénéfices de notre approche en terme de préservation de la confidentialité de données au travers d'une application médicale de partage de données. Le scénario utilisé expose des aspects critiques comme la gestion de politiques de confidentialité, la publication de données, la recherche de données et la recherche de droits d'accès personnels.

Mots-clefs : data privacy, privacy preferences, P2P data sharing, DHT.

1 Introduction

We consider Privacy-Preserving Applications (PPA) those that manage sensitive or confidential data (e.g., medical records, private research results, financial information, legal documents, personal information). In general, PPAs are developed on trusted servers, nevertheless, such servers have the inherent disadvantages of client-server architectures. P2P (Peer-to-Peer) systems offer to professional online communities (e.g., medical or research communities) important advantages such as availability, scalability, distribution and autonomy. Building PPAs on top of P2P systems would facilitate data sharing, but to make this possible, it is necessary to provide some privacy guarantees since P2P systems can be considered as hostile because data can be accessed by potentially untrusted peers and used for unwanted uses (e.g., marketing, profiling, fraudulence or activities against the owner's preferences or ethics).

Data privacy is the right of individuals to determine for themselves *when*, *how* and *to what* extent information about them is communicated to others [1]. Organizations and legislations have defined well accepted privacy principles [1]. Data privacy laws have accentuated the relevance of the *purposes* of data access and the respect of user's privacy

1. http ://www.oecd.org

preferences in particular those of data owners. Several P2P systems propose mechanisms to ensure some kind of privacy such as OceanStore [2], Past [3], Freenet[4], OneSwarm [5] and SwarmScreen [6]. However, these works remain insufficient to cope with data owner's privacy. Questions like the following remain open : (a) how data owners can define dynamically their privacy preferences, (b) how they can control who access their data, (b) for what purposes their data can be used, (c) how they can know whom actually accessed their data, (d) how they decide who may know their privacy preferences, etc.

To give an answer to those questions, we proposed PriServ [2] [7, 8, 9], a privacy service implemented on top of a Distributed Hash Table (DHT). The key feature in PriServ is that owner peers (data publishers) keep control over their shared sensitive data. PriServ allows data owners to specify their privacy preferences and combines purpose based access control, trust and encryption.

This demonstration shows the PriServ prototype (Section 2) and the benefits of our approach through a medical PPA for data sharing (Section 3). The scenario we use exhibits critical aspects like privacy policy management, data publishing, data requesting and rights access searching.

2 PriServ, a privacy service for P2P data sharing

The goal of PriServ is to prevent unauthorized disclosure, data misuse, attacks to data integrity, and over all, to provide data owners a tool to share their sensitive data in P2P systems. Inspired from the Hippocratic databases [10], the mean challenge is to constraint data requesters to specify their intentions for data usage in terms of access purpose (e.g., a particular research, a particular project purpose, marketing, etc.) and operation (i.e., read, write, disclosure) while taking advantage of P2P architectures. The idea is to commit requesters to use data only for the intended purpose and operation. Legally, this compromise may be used against malicious requesters if it is turned out data have been used for purposes/operations non expressed in data requests.

PriServ is a middleware service implemented on top of DHTs and is independent of the type of DHT used. It can be based on whatever DHT as long as it implements the traditional *put* and *get* functions. Access purposes and operations are integrated in the creation of data hash keys. This way, publishing and requesting in PriServ are always done for specific purposes and operations. PriServ offers to the application layer two ways for publishing (publishData and publishReference).

Peers can have three roles : a *requester* to request data, a *server* to store and provide data and an *owner* to share data. Before publishing, data owners should define the privacy policies (PPs) they want to be preserved when accessing their data. PPs contain allowed access purpose, operation, conditions (e.g., retention time, obligations before or after using data, semantic conditions), necessary minimal trust levels and a ACL (Access Control List).

To summarize, the main functions of PriServ are the following :

publishData(data, PPId). Owners use this function to publish data content under a particular PP (PPId). To protect data privacy against potential untrusted servers, before distribution, data content is encrypted (symmetric cryptography), and digital signatures are used to protect data integrity from servers.

publishReference(data, PPId). Owners use this function to publish data references

2. http ://atlas.lina.univ-nantes.fr/gdd/appa/priserv

under a particular PP. Publishing only data references and storing data locally allow owners to provide themselves their data to right requesters.

request(dataRef, purpose, operation). Requesters use this function to request data for a specific purpose to perform a specific operation. This function compels requesters to specify the access purpose and the operation they will apply to requested data. For requesters, the way data have been published is transparent (publishData or publishReference), they will use always this request function.

dataRefSearch(purpose, operation). Requesters use this function to know the references of data they are authorized to request (their access rights) for a particular purpose and operation. This operation avoids the P2P system to publish a global schema. By using this function, a requester will know only the available data it is able to request, it will not know those information about other requesters. In this way, privacy of requesters and owners is preserved.

During the request process, the peer storing the data makes an access control based on the ACL sent by the owner during the publishing process. The data owner is always contacted by the requester either to request the data content (if only references have been published) or the decryption key (when encrypted data content have been published). Before satisfying the request, data owners check locally the trust level of requesters. If the trust level does not exists locally, they make a limited flooding among some friend peers (or known peers). If the data content has been published, before contacting the owner, requesters generate the digital signature of the encrypted data sent by the server (by using a known hash function for example). This signature is sent to the owner which can verify if the server has attacked the integrity of the data.

The publish cost of PriServ remains logarithmic. In general, in the literature it is considered that one data may be published at most for 10 different purposes and as the number of operations we consider is 3, the incurred overhead is minimal. The logarithmic requesting cost is affected by the potential limited flooding made to search the trust level of requesters. Nevertheless we have shown in [9] how this cost can be reduced and stabilized to a minimum cost. The dataRefSearch cost is logarithmic and the cost allowing this function possible is also logarithmic but depends on the pair (purpose, operation). To distribute the index of available data in the DHT, each owner periodically publish the list of dataRef linked to an ACL that is accessible under the intention pair (purpose, operation). Thus, in PriServ, the P2P storage system is used to store data and the index of stored data.

In the prototype, peers are implemented as Java objects using the Service Component Architecture (SCA [3]). They can be deployed over a single machine or several machines connected together via a network. Each object contains the code needed for calling the PriServ functions. To communicate between peers, we use Java RMI.

3 Medical PPA

In this demonstration, we consider a PPA for a collaborative medical research where medical records are shared. Participants of this application are scientists, doctors, students, nurses, etc. Any peer can be a requester and owners are doctors that share sensitive data (e.g., medical records of their patients).

Doctors, after defining their PPs (in accordance with their patients), associate them

3. http ://www.eclipse.org/stp/sca

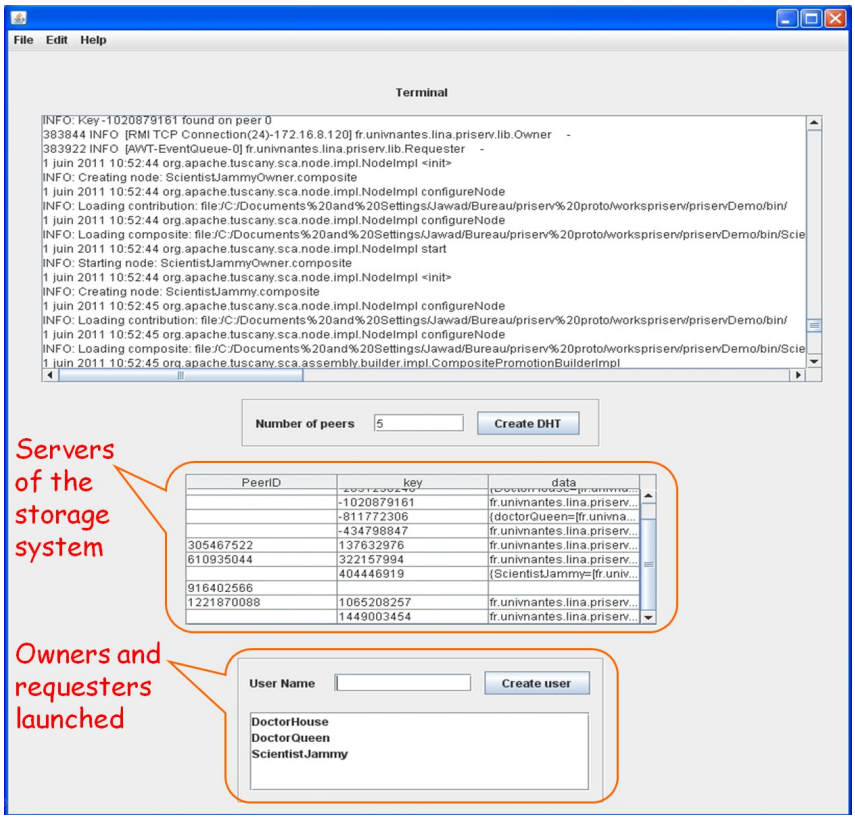

FIGURE 1 – Dashboard of the storage system (DHT)

to their medical records. For instance, a *doctor* may allow *reading* access on her medical records to a *nurse* for *consulting records* or may allow *writing* access on her information to another *doctor* for a *second diagnosis*.

The prototype provides a dashboard (Figure 1) to launch the P2P storage system and a user interface (Figure 2) for owners and requesters.

The demonstration starts by launching the dashboard of the P2P storage system (Figure 1 shows the peers acting as servers) and the peers of the medical PPA (DoctorQueen, DoctorHouse, etc.). Then, we will show the facilities provided to doctors to define their PPs and publish their medical records attached to their privacy preferences (see the owner zone in Figure 2). We will finish the demonstration by showing how requesters can ask the system their access rights (see allowed data in the requester zone of Figure 2) and how they are constrained to respect the privacy preferences of doctors during the data request.

4 Conclusion

With PriServ, data owners keep control of their shared data and their privacy preferences are preserved. It prevents malicious data access by compelling requesters to specify their usage intentions during the request process. PriServ makes possible to share sensitive data based on P2P systems not only in medical but in any collaborative community (e.g., banking, research, financial, familiar, etc.). It has integrated new privacy principles to data sharing in P2P systems, such as purpose specification, which make P2P systems a more trustworthy environment on which PPAs can be built.

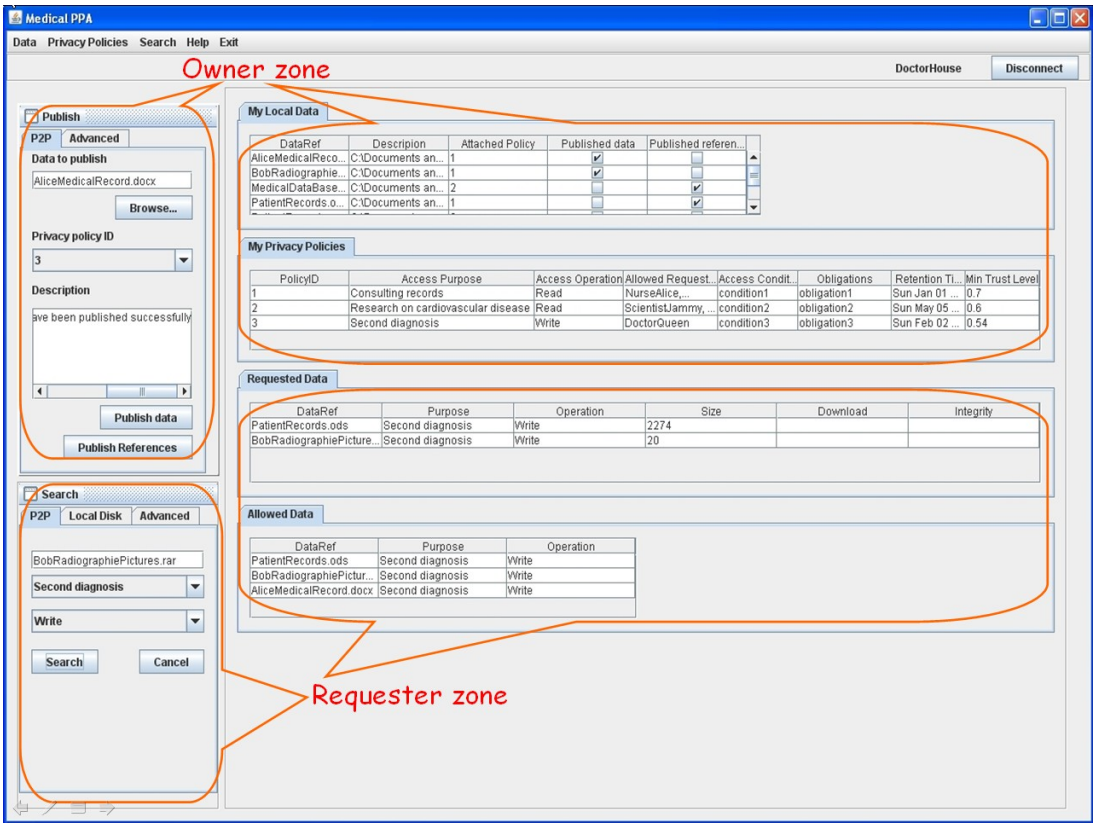

FIGURE 2 – GUI of the medical PPA that uses PriServ

Références

[1] Westin, A. : Privacy and Freedom. In : Atheneum, New York (1967)

[2] Kubiatowicz, J., et al. : OceanStore : An Architecture for Global-Scale Persistent Storage. In : Architectural Support for Programming Languages and Operating Systems (ASPLOS). (2000)

[3] Rowstron, A., House, G. : Storage Management and Caching in PAST, a Large-scale, Persistent Peer-to-Peer Storage Utility. In : Symposium on Operating Systems Principles (SOSP). (2001)

[4] Clarke, I., Miller, S.G., Hong, T.W., Sandberg, O., Wiley, B. : Protecting Free Expression Online with Freenet. IEEE Internet Computing **6** (2002) http ://www.aqualab.cs.northwestern.edu/projects/SwarmScreen.html.

[5] Isdal, T., Piatek, M., Krishnamurthy, A., Anderson, T. : Privacy-Preserving P2P Data Sharing with OneSwarm. Technical report, University of Washington (2009) http ://www.oneswarm.org/.

[6] Choffnes, D.R., Duch, J., Malmgren, D., Guiermà, R., Bustamante, F.E., Amaral, L. : SwarmScreen : Privacy Through Plausible Deniability in P2P Systems. Technical report, Northwestern EECS University (2009)

[7] Jawad, M., Serrano-Alvarado, P., Valduriez, P., Drapeau, S. : A Data Privacy Service for Structured P2P Systems. In : Mexican International Conference in Computer Science (ENC). (2009)

[8] Jawad, M., Serrano-Alvarado, P., Valduriez, P., Drapeau, S. : Data Privacy in Structured P2P Systems with PriServ. In : Bases de Données Avancées (BDA). (2009)

[9] Jawad, M., Serrano-Alvarado, P., Valduriez, P. : Design of PriServ, A Privacy Service for DHTs. In : Privacy and Anonymity in the Information Society (PAIS), collocated with EDBT. (2008)

[10] Agrawal, R., Kiernan, J., Srikant, R., Xu, Y. : Hippocratic Databases. In : Very Large Databases (VLDB). (2002)