Lifelogging Protection Scheme for Internet-Based Personal Assistants

David Pàmies-Estrems[1,2](✉), Nesrine Kaaniche[3,4], Maryline Laurent[3,4], Jordi Castellà-Roca[1,2], and Joaquin Garcia-Alfaro[3]

[1] Department of Computer Science and Mathematics, CYBERCAT-Center for Cybersecurity Research of Catalonia, Universitat Rovira i Virgili, Tarragona, Spain
david.pamies@mail.com

[2] UNESCO Chair in Data Privacy, Tarragona, Spain

[3] Télécom SudParis & CNRS/SAMOVAR, Université Paris-Saclay, Paris, France

[4] Chair Values and Policies of Personal Information, Institut Mines-Télécom, Paris, France

Abstract. Internet-based personal assistants are promising devices combining voice control and search technologies to pull out relevant information to domestic users. They are expected to assist in a smart way to household activities, such as scheduling meetings, finding locations, reporting of cultural events, sending of messages and a lot more. The information collected by these devices, including personalized lifelogs about their corresponding users, is likely to be stored by well-established Internet players related to web search engines and social media. This can lead to serious privacy risks. The issue of protecting the identity of domestic users and their sensitive data must be tackled at design time, to promptly mitigate privacy threats. Towards this end, this paper proposes a protection scheme that jointly handles the aforementioned issues by combining log anonymization and sanitizable signatures.

1 Introduction

Most of the time, we use tools created by third parties to access the information we need from the Internet. Traditionally, people have been using web search engines, as the main gateway to the Internet. As time goes by, we can find other alternatives. New proposals are trying to reduce the barriers to access information even more, and to make it accessible to everyone. As a consequence of these innovations, today we can find a multitude of technological tools that have been developed precisely for this reason, leading towards Internet-based personal assistants, consolidated by technologies such as smartphones, smartwatches and smartgateways.

For reasons of economics of scale, the development of this type of devices is only available to a few technological organizations [8]. These few organizations can have access to all the data generated using their devices, such as user queries and usage statistics [17]. It is often easy to forget that all of our usage data is

J. Garcia-Alfaro et al. (Eds.): DPM 2018/CBT 2018, LNCS 11025, pp. 1–10, 2018.
https://doi.org/10.1007/978-3-030-00305-0_31

stored on Internet servers. In this case, the situation is even more accentuated, since the user is not in front of a computer. Users tend to establish more relaxed relationship with the device, sometimes without even knowing if it is working or sending information to another site, and mostly seeing it as a friend or an extension of its person.

The reality is that these devices are usually interconnected to other services. When we make a request to them, the organization that created them performs an information request on their servers. Apart from fulfilling the request, several other data gets registered in the form of a log. Anyone who uses these services is constantly generating logs and providing personal information to the organization. Additionally, searches made on most modern devices often send the user location and the local time as two additional parameters when it comes to finding the most convenient information in each situation. Therefore, user, location and local time are also registered in the logs of the servers.

Lifelogging, i.e., the recording of information about our everyday lives using smart devices, involves the collection of a huge volume of sensitive information [20]. It can lead to very serious privacy risks of personal data disclosure, as these data can be exploited in isolation, as well as combining the information generated between several of these devices. In addition, the widespread development of technologies such as Artificial Intelligence and Big Data, make the task of extracting information or relevant relationships easier every day [6]. To protect the identity and sensitive users' data, there are some techniques that allow to eliminate direct users' identifiers. However, a specialized type of attack, called Record Linkage attack, allows to link different user records, which contain seemingly harmless information, but when all the data can be mapped, it can end up revealing sensitive information from the users [15].

In this paper, we address the issue of transforming raw user's data from lifelogging data streams generated by Internet-based personal devices like Google Home and Amazon Echo [13]. We study the relation of such devices with other data information actors in terms of EU data protection directives and propose a protection solution via anonymity transformation and malleable signatures. Our proposal takes into account the role of the organizations and their needs to monetize generated data. Our protection scheme aims at limiting the risk of privacy disclosure, while maintaining an adequate level of data utility. The paper is organized as follows. Section 2 reports related work. Section 3 provides the background. Section 4 presents of our proposal. Section 5 closes the paper.

2 Related Work

Early methods to transform raw user's information to a set of privacy protecting data started with batch processing methods. Batch processing methods rely on executing match processing techniques (e.g., via statistic or semantic matching techniques) to remove the interactions that disclose user's identity from a series of stored user logs. Some methods would simply remove old sets of interactions assuming that the logs will not be large enough to enable identity disclosure [5].

This lead to flawed techniques given the likelihood of highly identifying interactions. Even the removal of highly identifying data, such as credit cards or addresses [4], are prone to record linkage attacks [2].

The use of *statistical disclosure control* methods can help to reduce the number of deleted records [12]. They group together sets of similar logs. Then, they use prototypes of interactions, instead of the original interactions, making them indistinguishable from each other. Users are still conserved and the interactions are transformed to minimize the risk of information disclosure. Such methods can be improved to include real-time processing, to minimize and avoid the storage of large sets of data requiring *a posteriori* treatment. Open problems using *statistical disclosure control* methods include data mining processing of large network data streams [11].

The work presented in this paper extends an anonymization scheme for web search logs using *statistical de-identification* [14]. The original scheme allows to web search engine providers to share user's raw data with third party organizations with a high degree of privacy and a relatively low decrease of data utility. The extension allows more complex data structures based on lifelogging logs, resulting on an increase of data attributes, such as spatial location of the queries and the processing of user commands. It combines *sanitizable signatures* [1] with probabilistic *k-anonymity* privacy preservation [14,19].

Sanitizable signatures are malleable mathematical schemes that allow a designated party, the *sanitizer*, to modify given parts of a ciphertext c, created by the *signer*. The sanitizer can modify parts of c in a controlled way. The signer divides $c \in \{0,1\}^*$ into N blocks $m_1, \cdots, m_N$, and provides a subset $\textsc{Adm} \subseteq \{1, N\}$ to the sanitizer. The subset $\textsc{Adm}$ represents the description of the admissible modifications. In the end, the signer signs c using a key related to the sanitizer. Using the aforementioned key, the sanitizer is able to modify the admissible parts of c defined in $\textsc{Adm}$, in a way that keeps the resulting signature valid, under the public key of the signer. The scheme can satisfy *unlinkability*, to guarantees that it is unfeasible to distinguish between sanitized signatures that have been produced from the same ciphertext or by the same sanitizer. It is also possible to limit the set of all possible modifications on one single block and to enforce the same modifications on different messages blocks [3].

The combination of sanitizable signatures and probabilistic k-anonymization in our approach satisfies indistinguishability and real-time (e.g., streaming) data processing [10]. Indistinguishability in k-anonymity methods guarantees that each record in the dataset that has been k-anonymized is indistinguishable from at least $k - 1$ other records. Probabilistic k-anonymity relaxes the indistinguishability requirement of k-anonymity and only requires that the probability of re-identification is the same as in k-anonymity, i.e., users cannot be re-identified by record linkage attacks with a probability greater than $1/k$. In addition, anonymized logs are generated using real user queries, i.e., they are not modified, but distributed among other users with similar interests, leading towards quasi-identifiers that get dispersed between several users and thus preventing record linkage attacks, while maintaining data utility as well [14].

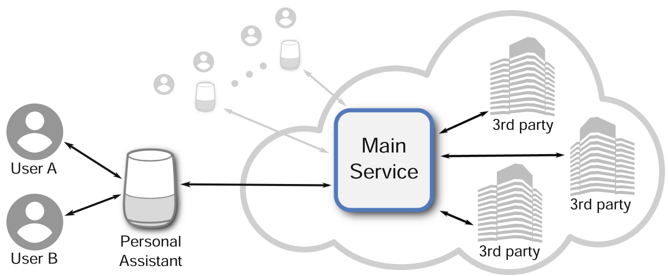


Fig. 1. Architecture for existing Internet-based personal assistants. *Users* represent the data subject, authorized to interact with the *Personal Assistant* devices, by submitting queries and commands. *Personal Assistant* devices send those commands to the *Main Service* that take the role of data collectors. Finally, *Third Parties* are the entities acting as data processors. They represent the parties with interest on legitimately accessing the anonymized logs.

3 Problem Statement

3.1 EU Data Protection Actors

EU Directive 95/46/EC, nowadays superseded by the new General Data Protection Regulation (GDPR) [16], to which we will refer during the rest of the paper, defines different roles that are relevant to the protection of general-case lifelogging environments. First, it defines the *Data Controller* as "*the natural or legal person, public authority, agency or any other body which alone or jointly with others determines the purposes and means of the processing of personal data*" [7]. Lifelogging environments need to clearly identify who is the *Data Controller*, since it determines which national law is applied. The data controller is responsible to determine which data must be processed, which third parties can access this data and when this data must be deleted.

Moreover, the figure of the *Data Processor* has the responsibility to ensure the security in the processing of personal data. The directive states that it is the "*natural or legal person, public authority, agency or any other body that processes personal data on behalf of the controller*". It is also necessary to determine the *Data Processor*, as it also sets the national law to be applied. It is also necessary to consider the *Data Subject*, as the person who is generating the data and from which we need the consent. The directive also requires to guarantee a set of basic rights to the *Data Subject*, such as the right to access their information or to oppose to the data processing.

Figure 1 depicts a lifelogging environment which involves several actors, namely: *Users*, *Personal Assistant devices*, *remote Main Services* and *Third Parties*. Users represent the actors related to data subject, i.e., they represent the entities that are authorized to interact with the Personal Assistant devices, by submitting queries and commands. The Personal Assistant devices receive both queries and commands from associated users. Queries and commands are sent and processed by the Main Services for customized results. The remote Main Services take the role of data collectors. They have direct access to the original queries and command and control logs, sent by the Personal Assistant devices. Third Parties are the entities acting as data processors. They represent the parties that express interest on legitimately accessing the anonymized query and command logs, to eventually process and use them.

3.2 Data Structure

Personal Assistant devices may receive three different types of queries: (1) general search queries, (2) location based queries and (3) commands. They are transferred to the Main Servers for processing. Hence, the Main Service stores all the original logs for each Personal Assistant device with respect to its different associated Users. Queries and commands are defined as follows:

- **General search queries** — Traditional web search-like queries. These queries help users to find what they are looking for, from Internet websites. Users just have to ask a question and the system returns the main result they are looking for.
- **Location based queries** — Use of spatial and temporal data. They can be classified on two main categories: elementary queries and derivative queries. *Navigation and search for Point of Interests* are typical elementary location based queries. Derivative queries are mainly processed for *guiding* or *tracking* to provide customized results to users.
- **Commands** — Allow users to request direct actions that affect their own environment. Actions are usually related to home automation, multimedia control and alarms. Although these actions usually only have a local repercussion, all the data they generate is also stored together with the rest of the logs.

3.3 Privacy and Utility Trade-off

The proposed scheme aims at fulfilling two main requirements (scalability and performance requirements will be addressed in future versions of the work). First, privacy requirements, in terms of user data protection. Second, data utility requirements, in terms of log monetization. These two requirements together allow that non-sensitive user information can be sold to Third Parties, allowing Third Parties to extract user characteristics from the data they acquire. Since query and command logs together can reveal sensitive information, a trade-off between anonymizing logs and keeping them useful to extract information through data mining processes must be guaranteed. Therefore, the main challenge related to data utility is to anonymize sensitive user data removing as few information as possible in order to have enough interesting information to be analyzed. To do so, the proposed scheme aims to build fake logs and user profiles, which should maintain users' interests and break quasi-identifiers that could allow to identify a user. Queries should be anonymized to not relate sensitive information to a user identity. It should be as difficult as possible to relocate queries in order to build original user's profile. In the end, the proposed system should generate those fake logs and profiles with other users' queries.

4 Our Proposal

We extend the initial architecture presented in Sect. 3 to handle the aforementioned goals in terms of privacy regulation, security and functional requirements. Figure 2 depicts the extended architecture. An entity named the Identity Screener ensures the compliance with the legal constraints and requirements to settle, e.g., privacy prevention algorithms, based on criteria set by EU regulation directives [7,16]. It acts as a container of privacy filters to enforce data protection and control any misuse between any other parties. A second entity, the Auditor, acts as a dedicated agent which is responsible of auditing the Identity Screener and the Main Service activities, with respect to accountability and users' consent requirements. In the sequel, we describe more in depth the working properties of our extended architecture and its idealized Identity Screener conducting sanitizable signatures and pre-anonymization of logs.

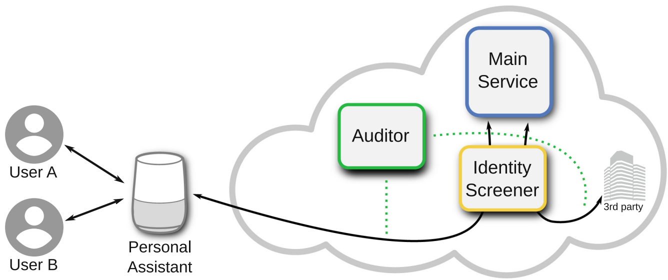
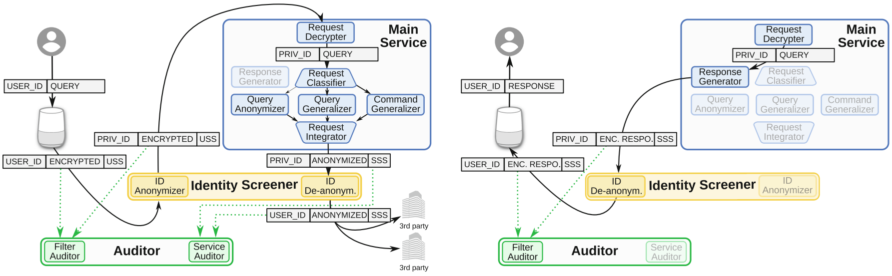


Fig. 2. Extended architecture. It includes an Identity Screener ensuring the compliance of privacy; and an Auditor, responsible of auditing accountability and users' consent requirements.

4.1 Working Properties of the Extended Architecture

To elaborate on the operations of the extended architecture, we refine and examine more in depth the internal components that the full system requires to handle requests and responses. Figure 3 depicts the proposed system. It shows the interactions of a User and its Personal Assistant, and the eventual generation of queries. The queries are sent through the network for treatment. Once treated, the resulting logs become properly anonymized. Then, it becomes possible to provide the anonymized logs to third parties, e.g., to monetize them. Next, we describe the main steps performed at each stage.



(a) (b)

Fig. 3. (a) Request architecture: a User interacts with its Personal Assistant, generating a series of queries that are sent through the Identity Screener (which sanitizes the user identity) to a Main Service that anonymizes the queries, and redirect them back through the Identity Screener to the Third Parties. (b) Response architecture: Main Service creates and signs the response for the User, via the Identity Screener, which restores the User identity and redirects the response to the Personal Assistant (i.e., decrypts and provides the result to the User).

1. **System initialization** — As a prior step to the start of the system execution, it shall be ensured the distribution of the key pairs to create and check the User Sanitizable Signatures and the Service Sanitizable Signatures, as well as the public key of the Main Service to all the Personal Assistants.
2. **Query pre-processing** — Two steps are conducted. **First**, in a local step, the User sends a question to its Personal Assistant, which recognizes who has formulated the question and transforms it into text. Once transformed, the query is encrypted using the public key of the Main Service and gets signed using the User Sanitizable Signature. The signature allows the Identity Screener to modify some data about the user (e.g., its real identity), but keeps the remainder elements of the query. **Second**, the query is sent to the Identity Screener (e.g., a distinct administrative entity than the Main Service). A specific module replaces the original User identifier (cf. *USER_ID* in Fig. 3) with a pseudonymous (cf. *PRIV_ID*), preventing the Main Service from knowing the real identity of the user that generated the original query (the Identity Screener does not have access to the original query, which remains encrypted).
3. **Anonymization** — Procedures conducted at the Main Service:
   - **Request Decrypter:** Verifies the signature of the query and decrypts the body of the query with the Main Service private key.
   - **Request Classifier:** Determines the log class (w.r.t. the three classes in Sect. 3.2) and decides how the log shall be treated. General search queries are redirected to the *Query Anonymizer* procedure [14], location-based queries to the *Query Generalizer* procedure [18] and command-based queries to the *Command Generalizer* procedure [9] (conducting k-anonymity and data perturbation treatment tailored for each class).

   - **Request Integrator:** Unifies the anonymization results, adds a Service Sanitizable Signature (to allow the Identity Screener to modify the User field, but not the rest) and releases the logs.
4. **Query post-processing** — The Main Service releases the anonymized logs to the Identity Screener, which checks the Sanitizable Signature Service. If the check is validated, it restores the original USER_ID, through the *ID De-anonymizer* procedure. This way, the Third Parties can extract the interests of users, while protecting the logs from record linkage attacks (since the text of the query remains conveniently anonymized).
5. **Audit** — The auditing process is performed by a dedicated authority, mainly relying on the verification process of Service Sanitizable Signature. That is, the auditor has to verify the consistency of signed queries and responses, generated by the User, the Identity Screener and the Main Service, such as:
   - **Identity Screener activities auditing** — Verification of Identity Screener signed queries consistency. Honestly generated signatures (using *signing correctness*) and sanitized signatures (using *sanitizing correctness*) have to be accepted by the verifier. Honestly generated proofs on valid signatures (*proof correctness*) have to be accepted by the Service Sanitizable Signature algorithms [1].
   - **Main Service activities** — Verification of the consistency of signed original queries' responses and anonymized query logs, generated by the Main Service. Each anonymized query has to be sent through the Identity Screener in order to retrieve the USER_ID query identifier, before transmitting to Third Parties. Hence, the auditor may check the signatures after the Main Service and the Identity Screener processing, as well as to verify if transfer actions are allowed with regard to each user authorization vector.

4.2 Discussion

Some limitations in our approach remain open. First, w.r.t. Users's communication, it must be ensured that the Personal Assistant does not send information to the Main Service directly, therefore escaping the treatment of the Identity Screener. On the contrary, the communication with the Third Parties does not have this problem. If they want to recover the original USER_ID, all messages must go through the Identity Screener. In this case, the possible privacy problem would appear if any of the Third Parties send the data back to the Main Service once it has been processed by the Identity Screener. If this situation arises, the Main Service would have access to the anonymized query and the original USER_ID. If the Main Service stores the correspondence between the original query and the anonymized query, it could fetch the original Query and User pair. Solutions to handle these limitations are under investigation.

5 Conclusion

Internet-based personal assistants can lead to serious privacy risks. They may release sensitive information about the identity of domestic users and their sensitive data. The issue must be tackled by jointly addressing anonymization by organizational roles in terms of *Data Controller*, *Data Processor* and *Data Subject*. Towards this end, we have proposed an architecture that combines lifelogging anonymization and sanitizable signatures, to promptly mitigate privacy threats. Future work includes a more exhaustive analysis about the cooperation of the different elements of our architecture, as well as to provide further investigations about the current techniques included in the architecture with a specific brand of Internet-based personal assistants. Ongoing code for the implementation of our proposal is available at github (cf. http://j.mp/lps-ipa).

References

1. Ateniese, G., Chou, D.H., de Medeiros, B., Tsudik, G.: Sanitizable signatures. In: 10th European Conference on Research in Computer Security, ESORICS 2005 (2005)
2. Barbaro, M., Zeller, T.: A Face Is Exposed for AOL Searcher No. 4417749 (2006). https://www.nytimes.com/2006/08/09/technology/09aol.html
3. Canard, S., Jambert, A.: On extended sanitizable signature schemes. In: Pieprzyk, J. (ed.) CT-RSA 2010. LNCS, vol. 5985, pp. 179–194. Springer, Heidelberg (2010). https://doi.org/10.1007/978-3-642-11925-5_13
4. Center for Democracy and Technology: Search privacy practices: a work in progress (2007). http://www.cdt.org/privacy/20070808searchprivacy.pdf
5. Cooper, A.: A survey of query log privacy-enhancing techniques from a policy perspective. ACM Trans. Web (TWEB) **2**(4), 19 (2008)
6. Danezis, G., Domingo-Ferrer, J., Hansen, M., Hoepman, J.-H., Metayer, D.L., Tirtea, R., Schiffner, S.: Privacy and data protection by design-from policy to engineering. arXiv preprint arXiv:1501.03726 (2015)
7. European Parliament and Council of the European Union: Directive 95/46/EC of the European parliament and of the council (1995)
8. Grimes, J.M., Jaeger, P.T., Lin, J.: Weathering the storm: the policy implications of cloud computing (2009)
9. Gruteser, M., Grunwald, D.: Anonymous usage of location-based services through spatial and temporal cloaking. In: International conference on Mobile Systems, Applications and Services, pp. 31–42. ACM (2003)
10. Guo, K., Zhang, Q.: Fast clustering-based anonymization approaches with time constraints for data streams. Know. Based Syst. **46**, 95–108 (2013)
11. Krempl, G., et al.: Open challenges for data stream mining research. ACM SIGKDD Explor. Newsl. **16**(1), 1–10 (2014)
12. Navarro-Arribas, G., Torra, V.: Tree-based microaggregation for the anonymization of search logs. In: 2009 International Joint Conference on Web Intelligence and Intelligent Agent Technology, Washington, DC, USA, pp. 155–158 (2009)
13. Nijholt, A.: Google Home: experience, support and re-experience of social home activities. Inf. Sci. **178**(3), 612–630 (2008)

14. Pàmies-Estrems, D., Castellà-Roca, J., Viejo, A.: Working at the web search engine side to generate privacy-preserving user profiles. Expert Syst. Appl. **64**, 523–535 (2016)
15. Poblete, B., Spiliopoulou, M., Baeza-Yates, R.: Website privacy preservation for query log publishing. In: Bonchi, F., Ferrari, E., Malin, B., Saygin, Y. (eds.) PInKDD 2007. LNCS, vol. 4890, pp. 80–96. Springer, Heidelberg (2008). https://doi.org/10.1007/978-3-540-78478-4_5
16. Regulation (EU): 2016/679 of the European parliament and of the council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data, OJEU l 119/1 of 4.05.2016. In: Elementary Particle Theory (2016)
17. Sarconi, P., Calore, M.: OK, house: get smart. How to make the most of Amazon Echo and Google Home. Wired **25**(6), 39–41 (2017)
18. Shankar, P., Ganapathy, V., Iftode, L.: Privately querying location-based services with SybilQuery. In: 11th International Conference on Ubiquitous Computing. ACM (2009)
19. Soria-Comas, J., Domingo-Ferrer, J.: Probabilistic k-anonymity through microaggregation and data swapping. In: 2012 IEEE International Conference on Fuzzy Systems, pp. 1–8. IEEE (2012)
20. Wang, P., Smeaton, A.: Using visual lifelogs to automatically characterize everyday activities. Inf. Sci. **230**, 147–161 (2013)