Privacy-Preservation for Publishing Sample Availability Data with Personal Identifiers

Ali Gholami and Erwin Laure
Swedish e-Science Research Center and Department of HPCViz and PDC, School of Computer Science and Communication, KTH Royal Institute of Technology
Email: {gholami, erwinl}@pdc.kth.se

Peter Somogyi and Ola Spjuth
Swedish e-Science Research Center and Department of Medical Epidemiology and Biostatistics, Karolinska Institute
Email: {peter.somogyi, ola.spjuth}@ki.se

Salman Niazi and Jim Dowling
Swedish e-Science Research Center and School of Information and Communication Technology, KTH Royal Institute of Technology
Email: {smkniazi, jdowling}@kth.se

***Abstract*—Medical organizations collect, store and process vast amounts of sensitive information about patients. Easy access to this information by researchers is crucial to improving medical research, but in many institutions, cumbersome security measures and walled-gardens have created a situation where even information about what medical data is out there is not available. One of the main security challenges in this area, is enabling researchers to cross-link different medical studies, while preserving the privacy of the patients involved. In this paper, we introduce a privacy-preserving system for publishing sample availability data that allows researchers to make queries that crosscut different studies. That is, researchers can ask questions such as how many patients have had both diabetes and prostate cancer, where the diabetes and prostate cancer information originates from different clinical registries. We realize our solution by having a two-level anonymiziation mechanism, where our toolkit for publishing availability data first pseudonymizes personal identifiers and then anonymizes sensitive attributes. Our toolkit also includes a web-based server that stores the encrypted pseudonymized sample data and allows researchers to execute cross-linked queries across different study data. We believe that our toolkit contributes a first step to support the privacy preserving publication of data containing personal identifiers.**

***Index Terms*—privacy protection, data encryption, distributed systems, database security**

I. Introduction

Over the last number of years, valuable data has been accumulated in many healthcare related databases throughout the developed world. By definition, these repositories contain sensitive medical data, which are fragmented depending on the type of clinical and research activities. Although it would be technically possible to merge the different data silos into a central database that could be queried by medical experts and researchers alike to allow for new insights previously thought impossible, the security risks of doing so are unacceptably high. Therefore, there is need to somehow combine the data from different databases, e.g., Bob's study DB and Alice's study DB in a way that minimizes the risk of exposing sensitive personal data.

In its simplest form, a use-case for the combined databases would be the following. Two databases are given: Bob's DB and Alice's DB. The Bob's DB contains entries of people who have some samples deposited related to their illness and are potentially eligible for research purposes. Alice DB is also a list of persons, along with their record of hospitalization and treatments they were subjected to. Both databases contain personal information, such as name, birth date and the Personal Identifier, which uniquely identifies them in the database. The proposed identifier can be used to link the different records in the different databases, but there is the obvious need to provide anonymity for the patients. While the combination of the information contained in different databases is extremely useful for research purposes, the actual information used to identify each individual isn't essential for the studies to be performed by the investigators themselves.

Existing sample availability systems, such as SAIL [1] provide individual level information on the availability of specific data types within a collection, not across foreign collections. That is, researchers are not able to cross-link (similar to an equality join in SQL) data from different outside studies, as the identity of the samples are completely anonymized. However, researchers would like to discover correlations between individuals in different studies, and this is not possible in the existing systems.

Manuscript received November 21st, 2013; revised May 7th, 2014.

doi: 10.12720/jomb.4.2.117-125

In this paper, we present a privacy-preserving system for publishing availability data about samples from patients to address the limitations of existing solutions, which allows researchers to cross-link sample availability data from different medical study databases, while preserving the privacy of the patients. To this end, we build an anonymization toolkit to anonymize and measure the re-identification risk of the sensitive data to be published, while cross-linking queries can be executed by the researcher.

The main contributions of this paper are:

- A graphical anonymization toolkit based on sdcMicro [2] and Java to anonymize sensitive data and to measure the re-identification risk;
- Combining encryption with pseudonymized personal identifiers (PIDs), for enabling cross-linking queries;
- Encryption of large anonymized data sets using X.509 public key certificates prior to publication via RESTful Web services into an integration server;
- Secure logging and auditing functionality for access to the datasets in the integration server.
- Our study shows that we are able to minimize the reidentification risk, while maintaining the ability to cross-link records even on different studies. We also learn that it is important for the issuers of personal identifiers to consider the use of ids that are not easily subjected to brute force attacks.

The structure of this paper is as follows. Section II describes the background and related work in the areas of privacy preservation and database federation. In Section III, we introduce the pseudonymization data model for personal identifiers. Section IV, discusses different anonymization methods to produce safe microdata. In Section V, we define the threat model including main threats to the privacy of the sensitive data. Section VI, presents an architectural overview of our solution and implementation details. Finally, we discuss the conclusions in Section VII.

II. Background and Related Work

A longtime confidentiality protection strategy is to dilute data by degrading the precision of given data records in a controlled process, so that the database can still satisfy the intended purpose, but is not specific enough to allow for easy re-identification. This challenge can be expressed as a task of managing re-identification risks, based on the identifying level of the attributes while taking into consideration the background knowledge available. The approach relies on an iterative optimization process without providing hard guarantees, mirroring risk management in other aspects of life such as being hit by an accident. A multilingual terminology for talking about privacy by data minimization including anonymity, unlinkability, undetectability, unobservability, pseudonymity, and identity management can be found in [3]. A recording or observational data set is called microdata. Every recording or observation has a set of variables. This set of variables needs to be categorized and may need to be modified in order to apply privacy-preserving data publishing measures. Microdata is expected to be safe, when its deliberate or accidental disclosure doesn't do any harm to the population involved.

To produce safe microdata, variables are categorized into at least three, not necessarily distinct groups: variables that are explicitly and directly identifying, such as personal numbers, social security numbers, serial numbers etc. Key variables (also called pseudo keys, quasi identifiers or non-sensitive attributes) are a group of variables that are identifying when used together. Linking based on key variables is applied when archived records are processed that have no explicit identifiers, e.g., [1] or when linking attacks are performed such as [4], [5] for the purpose of re-identification. Choosing is often based on mandatory items set forth by law (EU Data Protection Directive [6], HIPAA [7], Safe Harbor Agreement (SHA) [8] and such) or by managing the risk of being fined [9].

And last but not least, by using common sense: as a rule of thumb, non-sensitive attributes are the ones that are likely to appear in other databases, whether publicly accessible or not, therefore which can potentially be used for linking. A canonical example is the seemingly innocuous gender, birthdate and zip code triplet which is highly identifying to the majority of a population and can be found in a vast number of databases. Remaining variables (also called sensitive or non-confidential variables) that aren't in the two groups mentioned above. They are either not expected to appear in any other database and therefore cannot be used for linking or are not identifying by nature.

Preset software settings for acceptable risk levels may be set by legal requirements to be licensed as public use files (PUF) or microdata files under contract (MUC) for research purposes. As an indicator of the current state of affairs, data protected by HIPAA and Safe Harbor regulations result in a re-identification risk measure of approximately 0.04% (that is 4/10.000), ranging between 0.01% to 0.25% and being 10% to 60% in case of restricted data sets under nondisclosure agreements according to [10]. Further experimental measurements can be found in [11].

There are many existing toolkits to help produce safe microdata, providing commonly used anonymization algorithms such as k-anonymity [12] and *l*-diversity [13]. Argus [14], sdcMicro [2], and UTD Anonymization ToolBox [15], based on Incognito [16], are examples of open-source toolkits that provide workflow support for anonymizing sensitive data. Our toolkit leverages sdcMicro to anonymize our sensitive microdata. There have been similar attempts, several to ours, to provide support for federated queries over different data sources through database federation [17]-[19], where there are also two levels (local PID and global PID, collection PID and analysis PID hashes - using PGP keys). The identifier (global PID number, analysis PID etc.) is used to join different study data, not unlike in this case [19]. The

Clinical E-Science Framework (CLEF) [20] is another attempt to provide data privacy protection using pseudonymization. There have been also some efforts in the industry such as Custodix[1] to offer federated queries through implementing trusted third party (TTP) approach, however such TTP cannot be deployed in Sweden because of the existing restrictions.

III. Pseudonymity Model for Personal Identifiers

Data anonymization methods remove personally identifiable information (PII) of patients, helping to reduce the risk of patient re-identification when patient data is used for research purposes. However, researchers often want to crosslink different sample databases, for example, to discover correlations between different studies. For cross-linking, a PII could be used to link the individuals to their original records, however, using the original PII allows for patient re-identification. Pseudonymization is an alternative approach, where the PIIs are not stored in their original format, but crosslinking sample databases is still possible. There are two well-known pseudonymization schemes: to build a database to store mappings of the PIIs-to-pseudonyms or using cryptographic mechanisms applied to the pseudonyms [21].

Our proposed pseudonymity model to extract and convert the personal numbers is a two-level mechanism that maintains the possibility of joint queries over anonymous records in different collections. In Sweden, a personal number (Swedish civic registration number) is a 10-digit PIN issued by the National Tax Board for all residents in the country. The personal number or personal identifier (PID) is structured in three parts: date of birth, a three-digit birth number and a check digit. The date of birth construction contains two-digits each for the year, month, and date of birth, e.g., 610514. This is followed by a three-digit birth number (e.g., 323) and a check digit (e.g., 4), as shown in Fig. 1. The birth number value will be a number between 001 and 999, where the last digit is also used to indicate the gender, with men given an odd number and women an even number.

To de-identify the records, we use the secure hash algorithm (SHA-512) [22] to convert the PIDs to irreversible pseudonyms. To add another level of security, the de-identified records will be encrypted using the advanced encryption standard (AES) [23] with an embedded key in a slot of a Yubikey[2] device that is distributed off-line to the data providers. Fig. 1 illustrates the two-level mechanism, where SHA-512 de-identifies a Swedish PID and AES encrypts the pseudonymized PID (PPID).

We assume that our data model as a sample table (T) of a population, fragmented as PIDs, quasi-identifiers (QIDs), sensitive and non-sensitive attributes: T (PID, QID, Sensitive, Non-Sensitive). A QID can be considered as a combination of attributes that can be linked with external information to re-identify an individual e.g., zip code, birthdate and gender [4].

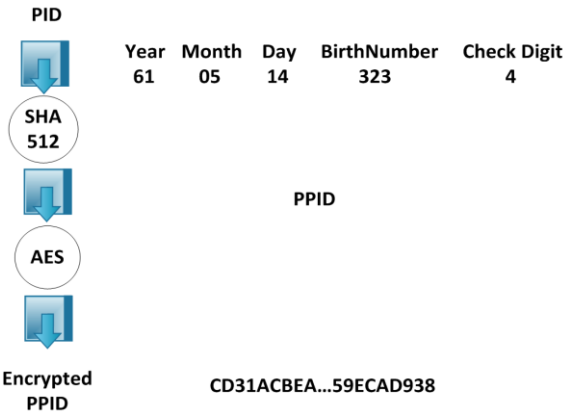

Figure 1. PID pseudonymization through a two-level hashing mechanism to provide the functionality for joint queries over different data sources.

IV. Anonymity Measures

To anonymize the data, several statistical disclosure control techniques can be applied to achieve anonymity measure goals, such as k-anonymity, *l*-diversity, or ε-differential privacy to reduce the re-identification risk of the participants as described in this section.

A. k-Anonymity

Sweeny, L. [4] proposed k-anonymity as a model for privacy preserving of QIDs. The k value is the minimum number of the records in a table that have similar QIDs. This notion of k records in a group reduces the risk of re-identification of a participant to the probability of *1/k*. However, *k*-anonymity is weak regarding the background knowledge of the adversary about a victim as described in [24].

B. l-Diversity

To overcome the limitations of *k*-anonymity, we use *l*-diversity [24] as an extra privacy measure to protect the anonymity of the individuals from re-identification through the adversary's background knowledge. The value of l defines at least *l* "well-represented" sensitive values in the table to reduce the confidence of inferring a sensitive attribute within a group. Entropy (E) of the entire table must hold E>log (*l*) to ensure every distinct QID block, at least has l distinct values for the sensitive attribute.

C. ε-Differential Privacy

ε-Differential privacy, a method proposed by Dwork [25] measures the privacy of an individual in a database through measuring the re-identification risk of that given participating individual by looking at the difference of the query results with and without inclusion to the database. Therefore, differential privacy helps to ensure that adding or removing a record to the database will not increase

[1] Custodix, https://www.custodix.com/
[2] Yubikey Website, http://www.yubico.com/

substantially the re-identification risk of a given individual included in the published data sets.

V. Threat Model

To ensure privacy protection through the proposed data model discussed in Section III, we define a threat model to declare the possible attacks and security breaches that cause loss of confidentiality and integrity of the data. The main threats to our data model are mainly due to cross-linking of the data sets with the de-identified PIDs that are not fully anonymized. Hence, we use an integration server that can be considered as a safe third-party server behind firewall that will be updated regularly with the patches and libraries, with restricted access to only administrative staff to ensure sufficient security and reliability.

A. Server Private Key Compromised

If server's private key will be compromised or stolen by an adversary, then the server's public key and associated private key should be revoked. In such a case, data providers should be notified about the incident and the anonymized datasets in the server's database should be re-encrypted with the server's new private key.

B. Inference Attacks

If a malicious adversary has access to both unencrypted anonymized data stored by the TTP and the key to pseudonymize the PIDs, then he/she will be able to make inference attacks through linking the victim's generated hashed personal identifier to the data acquired from the integration server e.g., through dictionary attacks. To mitigate the likelihood of such threats, all published data will be stored encrypted with the TTP's private key. Furthermore, when a researcher issues join queries over different databases, inference attacks become possible. For instance, issuing a query containing a small number of participants to find out, whether a specific person is available in any of the samples that are published in the integration server. As a countermeasure, the integration server will not accept queries less than N number of participants.

C. Malicious Sample Publication

We assume that sample data providers are trusted bodies, and that biobanks or other parties publish correct data sets to the integration server. However, a malicious or incorrect data provider could publish incorrect data sets either intentionally or by mistake. To reduce the effects of the publication data of incorrect, our system will keep track of the registered data and provide a flexible approach to remove the stored data.

D. Audit and Control

The TTP stores the audit trails and log files securely, to enable data providers to audit access to their data by the users. To ensure integrity of the audit trails, our system encrypts the audit logs and information using AES symmetric keys stored in the integration server key store.

E. Ethical Constraints

The usage of the system to issue and get results of the joint queries should be considered as a potential threat to the purpose of the collected data samples in the integration server. For instance, if a researcher uses the information for other studies that are out of scope of the agreed framework.

F. Query Reply Limitation

The final results of a query will be a set of records, combined within a table that is generated by the TTP. However, the combined table contains only the anonymized data but uniqueness of the entries in the intersection of different tables might raise the re-identification risk of the records through inference attacks. As a countermeasure, the TTP will check the anonymity of the combined table and will apply another level of minimization to ensure at least k-anonymity with the value of k=3.

G. Static Passwords

The Yubikey static passwords are vulnerable against a key logger that records keystrokes by a user. The information collected by a key logger usually saved as a file or sent directly to third parties. Because of the Yubikey function, as a USB keyboard, it will be possible for a key logger to intercept the text stream when in static mode. Therefore, users should be aware of underlying platforms and as a good recommendation use their local PCs to reduce the risk of password thefts.

VI. The ECPC Toolkit Overview

The e-Science for Cancer Prevention and Control (eCPC) is a flagship project within the Swedish e-Science Research Center (SeRC), aiming to develop a modular system for prediction of cancer initiation and progression using modeling and simulation. An important part of the project is to integrate data from different sources, such as biobanks containing data about samples, and clinical health registries (quality registries) containing information about patients and their diseases, treatments, and outcomes. This integrated data can then be used in subsequent modeling and simulation efforts. A big hurdle in medical data integration is the acceptance and participation of data providers. We present a first step in the data integration project operating on sample availability data, which lowers the barriers for data providers to participate. To this end, we developed a toolkit, shown in Fig. 2, which pseudonymizes sample availability data and then securely publishes the pseudonymized data to an integration server that can be queried by researchers (including support for crosslinking queries).

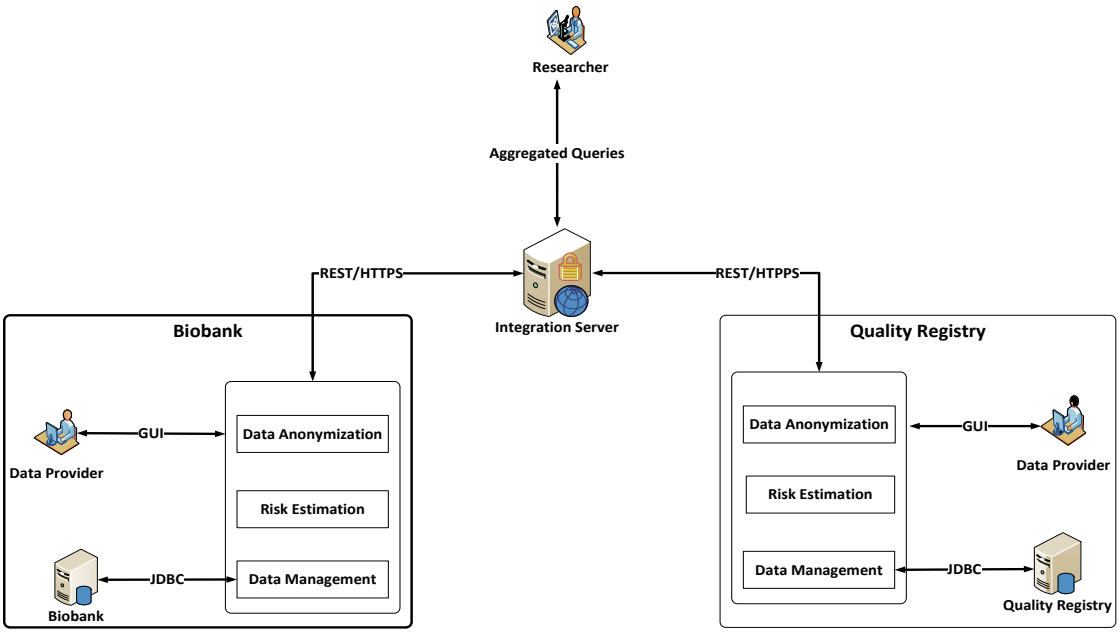

Figure 2. The eCPC toolkit design based on the privacy-preserving data publishing methods to upload the pseudonymized data to an external trusted third-party service.

In order to ensure the privacy of sensitive patient data, the eCPC toolkit applies the guidelines for safe microdata, outlined as follows:

- The eCPC toolkit removes all explicit identifiers, and it will extract and de-identify all the PIDs, as described in Section III;
- Then it categorizes the remaining attributes to determine the key variables according to both legal requirements and domain specific judgments, which may be subjective. Key variables might also be split into further categories according to the level that they are identifying. This distinction is useful when prioritizing which variables need to be modified to enhance safety: more identifying keys are modified first for observations that have a considerable high risk. This graded approach allows for better data quality preservation and therefore higher data utility;
- When key variables have been identified, the reidentification risk needs to be assessed. This is done by looking at the uniqueness of the observed entries through frequency counting and calculating probability estimates based on extrapolating models taking population frequencies into account;
- Entries that stand out from the rest and therefore have a considerable risk to be subject to re-identification are then modified. Numerous modification algorithms exist, namely generalization or global recoding and local suppression of outstanding values, recoding, swapping, rank swapping or perturbing with post randomization - not to be confused with randomized questionnaires when collecting data, hence the name post randomization. The methods to be applied depend on the nature of the variables, whether they are categorical or continuous, their structure such as significance order, hierarchy, geography, semantics and the size of the dataset in question. Nevertheless, each algorithm applied is recorded in a logbook for the analyst's documentation, e.g., the nature of the added noise, if any;
- The re-identification risk has to be measured again and the information loss has to be evaluated. If the risk is deemed acceptable and the quality of the data remains adequate, then the resulting microdata can be considered as safe. If not, the previous step needs to be repeated;
- Finally, the data provider e.g., biobank or quality registry, publishes the pseudonymized data sets to the integration server.

A. *Integration Server*

We implemented the eCPC integration server as a Java web application, as shown in Fig. 3. Researchers can visit the main page of the eCPC service and then they will be asked to authenticate to the system through SSL/TLS encrypted channels to protect their credentials from eavesdropping attacks. Users are authenticated using container-supported application level authentication provided by the web application server.

Furthermore, we setup a firewall in the eCPC integration server for security purposes that filters the traffic between the internal and external zones through the HTTPS connections. As Fig. 3 demonstrates, the integration server consists of three main components: security enforcement, integration engine and RESTful Web services API. The security enforcement component handles the security related tasks using encryption/decryption of the data sets, log files, user's authentication and auditing processes through the Java EE application server.

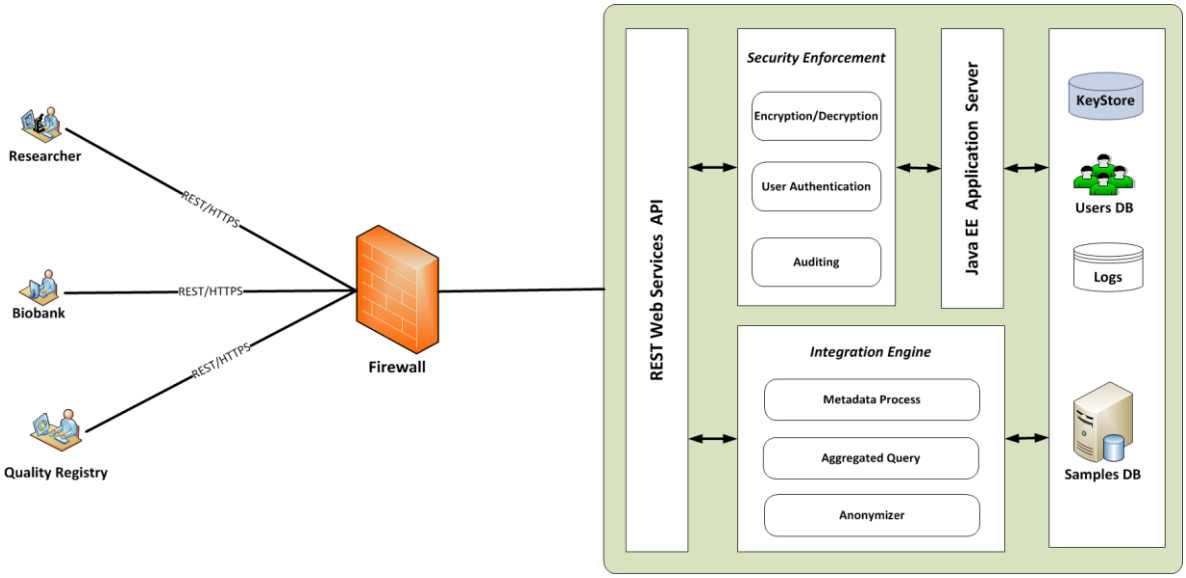

Figure 3. Overview of the eCPC integration server that is protected with firewall to filter the ingoing/outgoing traffic.

The integration engine deals with query processing and joins over different data sets. The metadata service, as a part of the integration engine, provides all available attributes from each data source that can be selected by the researcher when issuing a query. The RESTful Web services API receives incoming requests from the eCPC clients behind the firewall.

In a typical usage scenario, a researcher authenticates via a webpage with the web application server and after successful authentication he/she will be redirected to a webpage where he/she can issue queries. We store the data sets in MongoDB [26], see Fig. 3. Researchers can browse available data sets and select attributes of data sets for cross-linked queries. For example, a researcher might search for cross-linked samples in the prostate and diabetes quality registries by issuing a query like: 'how may samples are available for patients who have had both prostate cancer and diabetes and have a BMI greater than 30'. We used the built-in MapReduce API of MongoDB to implement joins over different data sets. As our queries are executed at application-level, we can still join across encrypted data sets by decrypting the contents of each data collection on the fly using the server's private key, stored in a secure key store.

The integration server also stores the logging events in a separate database by encrypting them using AES 256-bit symmetric keys, where keys are stored securely in the key store. When data providers wish to know about the access to their data in a specific period of time, the auditing component retrieves the logs associated with the data published by the owner and sends back the results to the data provider through RESTful API component.

B. Secure Data Management

In order to deploy a secure solution to store the pseudonymized published information on the integration server, we used public key certificates for encryption/decryption of the data sets. Although our data providers typically store their data sets in relational databases, the analysis of that data is typically carried out on comma separated values (CSV) files. As such, the main data format used for publishing data is a CSV file. The integration server provides a X.509 public key certificate to the client for data encryption, prior to data publishing. Fig. 4 shows the data encryption/decryption process, where a data provider encrypts its CSV files with the server's X.509 public key, and the integration server decrypts the uploaded data on-demand using its private key.

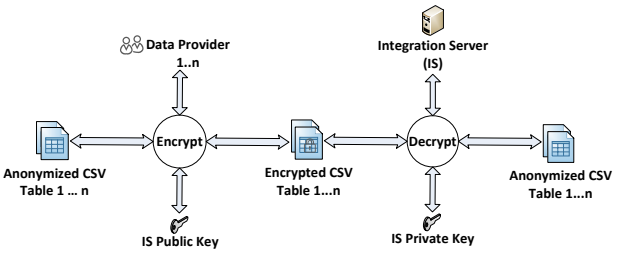

Figure 4. Public key encryption of the large sensitive data sets using the TTP's private key.

However, a limitation imposed on us by public key certificates is that the length of individual fields cannot exceed a size determined by the server's public key size, e.g., 512 bytes for a 4098-bit public key. However, this is not a problem for our toolkits as all fields are significantly smaller than 512 bytes.

The data management component (Fig. 2) uses Java database connectivity (JDBC) to export sensitive data sets from a relational data source to CSV files for pseudonymization. The data sets are then securely published to the integration server via a REST API running over HTTP on top of the SSL/TLS protocol, preventing eavesdropping attacks. The integration server also supports the removal of published data by a data provider.

C. Data Pseudonymization and Anonymization

The eCPC toolkit leverages microdata protection through anonymization algorithms implemented by *k*-

anonymity and *l*-diversity. Although the microdata as a whole is pseudonymized, we use existing data anonymization algorithms to anonymize sensitive attributes in the microdata. We now describe the anonymization phase in our pseudonymization process. A data provider selects a CSV data set that is generated from a relational database and defines the key and sensitive attributes to be anonymized. For this purpose, the eCPC toolkit visualizes the metadata of a specific CSV file, as shown in Fig. 5, to enable a data provider to tag attributes as sensitive, key or non-sensitive. We implemented our solution based on sdcMicro [2] that provides R-based API for both data anonymization and risk estimation. As our toolkit is implemented in Java, we ran sdcMicro in batch mode. We did not find any existing anonymization toolkits, e.g., μ-Argus [14] or UTD Anonymization Toolkit [15] that supports the calculation of re-identification risk and our anonymization algorithms, *k*-anonymity and *l*-diversity, in a platform-independent approach. Moreover, the R Java environments such as Rcaller [3] and Renjin [4] were not stable enough to run our sdcMicro tasks.

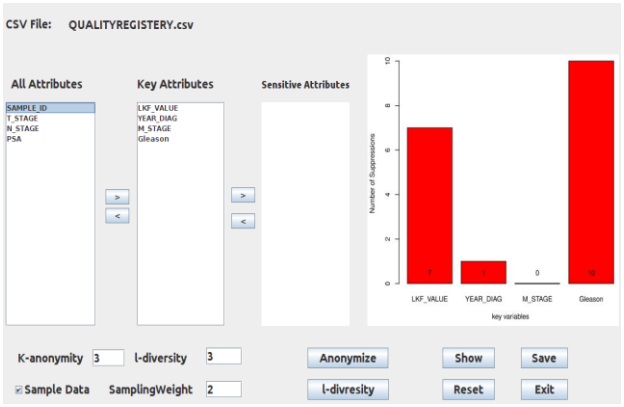

Figure 5. Public key encryption of the large sensitive data sets using the TTP's private key.

When a user presses the "Anonymize" button, two things happen: the PID is pseudonymized and the key attributes are anonymized. Afterwards, the user ensures that *l*-diversity is satisfied by setting the *l*-diversity value and pressing the relevant button The pseudonymization process converts PIDs using converted by the SHA-512 function, as described in Section III.

The pseudonymization process reads the embedded key in the Yubikey device to be used by the AES encryption function prior to data publishing

The anonymization phase provides the user with visual feedback in the form of a chart containing the number of suppressions for each sensitive attribute. An example of such a chart is given in Fig. 5. Anonymization will result in attribute values being suppressed when either *k*-anonymity or *l*-diversity constraints are not met. Our toolkit enables data providers to iteratively change the values of *k* and *l* to minimize the number of suppressions for a desired reidentification risk level.

D. Re-identification Risk

The purpose of the re-identification risk estimation process is to enable the data provider to measure the re-identification risk of anonymized sensitive attributes as a result of the data anonymization process, see Section VI-C. The eCPC toolkit allows the data provider to select the sensitive attributes for risk calculation. The risk measurement diagram of Fig. 6 demonstrates different levels of risk for different individual records. In this example, 43 out of 100 records will be reidentified with risk of r≤0.1.

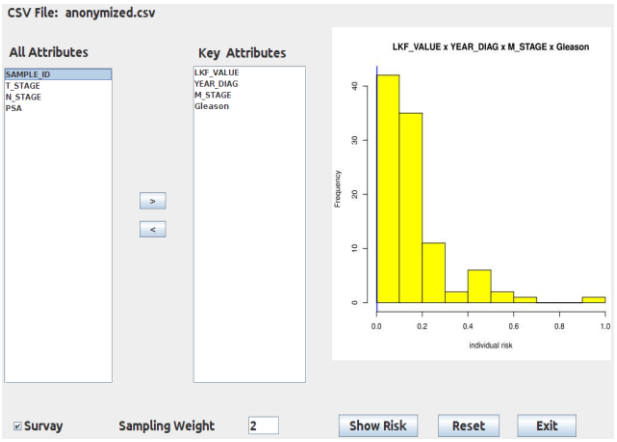

Figure 6. Public key encryption of the large sensitive data sets using the TTP's private key.

If the risk levels are deemed to be unsafe, the data provider will repeat the anonymization process with different values of k and l until the risk is considered to be acceptable, according to the safe microdata concept described in Section II.

E. Auditing Process

Our integration server supports researchers issuing queries for data availability from different data sources. In order to reduce the threat of malicious queries, we securely audit queries, identifying the queries that have been issued, by whom, and when. The eCPC toolkit client (a Java GUI) uses a REST API to allow data providers to download an audit trail for the queries that accessed their data source. For each query, the audit trail includes the name and institution of the requester, date of access, purpose of the study, IP address of the host that issued the query, and the actual query. Data providers can use this information to infer whether there has been a breach of privacy, and who was responsible for that breach. The audit trails can also be used to determine when a researcher is executing a brute force attack on the data sets.

VII. Conclusions

In recent times, patient clinical registries, medical studies, and biobanks have accumulated valuable clinical data that is currently not being fully exploited by medical researchers due to existing data being accumulated and

[3] RCaller, a library for calling R from Java, http://code.google.com/p/rcaller/

[4] Renjin, a JVM-based interpreter for the R language for statistical computing, http://code.google.com/p/renjin/

stored in off-line data stores, primarily for security, but sometimes also legal reasons. Although there are some existing systems that solve the problem of making sample availability information available online, such as SAIL [1], there are no systems we are aware of that support the cross-linking of data sources, as this requires storing pseudonymized references to the patient identifiers alongside the samples. Cross-linked data offers tremendous opportunities for researchers to identify inter-disease correlations in their modeling and simulation efforts. A big hurdle in medical data integration is the acceptance and participation of data providers.

To this end, we introduced a privacy-preservation publishing toolkit, called eCPC, that support the secure publishing of pseudonymized data sets to an integration server by data providers, and audited querying of the data sources by researchers. Our toolkit includes a secure de-identification mechanism for publishing pseudonymized patient identifiers through a two-level hashing mechanism, as well as tools to anonymize sensitive clinical data using *k*-anonymity and *l*-diversity algorithms. Our toolkit also estimates the re-identification risk for individual records, providing data providers with feedback for configuring the *k*-anonymity and *l*-diversity parameters. Furthermore, data providers can encrypt their large data sets using the public key certificate of the integration server for additional security. We also securely audit queries, helping to reduce the risk of misbehavior by researchers, and enabling subsequent identification of rogue users.

Our prototype demonstrated what we believe will be the first of many approaches to the privacy preserving publication of data containing personal identifiers. As data providers gain trust in the security of approaches such as ours, systems that support the cross-linking of pseudonymized data will appear that lead to new ways of utilizing sensitive data, in fields such as medical research.

ACKNOWLEDGMENT

The authors would like to thank Jan-Eric Litton from Karolinska Institutet for his support and Matthias Templ from Vienna University of Technology for helping with the sdcMicro package. This research work is funded by the Swedish e-Science Research Center (SeRC) as a part of the e-Science for Cancer Prevention and Control (eCPC) project.

REFERENCES

[1] D. Ford, K. Jones, J.-P. Verplancke, R. Lyons, *et al.*, "The sail databank: Building a national architecture for e-health research and evaluation," *BMC Health Services Research*, vol. 9, no. 1, pp. 157, 2009.

[2] M. Templ, "Statistical disclosure control for microdata using the rpackage sdcmicro," *Trans. Data Privacy*, vol. 1, pp. 67-85, Aug. 2008.

[3] A. Pfitzmann and M. Hansen, "Anonymity, unobservability, and pseudonymity: A consolidated proposal for terminology," July 2000.

[4] L. Sweeney, "Simple demographics often identify people uniquely," Carnegie Mellon University, Pittsburg, Working Paper 3, 2000.

[5] M. Gymrek, A. L. McGuire, D. Golan, E. Halperin, and Y. Erlich, "Identifying personal genomes by surname inference.," *Science (New York, N.Y.)*, vol. 339, pp. 321-324, Jan. 2013.

[6] E. U. Directive, "95/46/EC of the European Parliament and of the council of 24 October 1995 on the protection of individuals with regard to the processing of personal data and on the free movement of such data," *Official Journal of the EC*, vol. 23, 1995.

[7] U. States, Health Insurance Portability and Accountability Act of 1996 [microform], conference report (to accompany H.R. 3103). U.S. G.P.O [Washington, D.C.], 1996.

[8] US Department of Commerce, Safe harbour website. [Online]. Available: http://www.export.gov/safeharbor

[9] A. Pearlgood, "The impact of mandatory data infringement reporting," *Computer Fraud & Security*, vol. 2012, pp. 11-13, May 2012.

[10] K. Benitez and B. Malin, "Evaluating re-identification risks with respect to the hipaa privacy rule," *JAMIA*, vol. 17, no. 2, pp. 169-177, 2010.

[11] J. Qian and N. Qamar, "An experimental evaluation of de-identification tools for electronic health records," *CoRR*, vol. abs/1211.3836, 2012.

[12] L. Sweeney, "k-anonymity: A model for protecting privacy," *International Journal on Uncertainty, Fuzziness and Knowledge-based Systems*, vol. 10, no. 5, pp. 557-570, 2002.

[13] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "L-diversity: Privacy beyond k-anonymity," *ACM Trans. Knowl. Discov. Data*, vol. 1, Mar. 2007.

[14] A. J. Hundepool and L. C. R. J. Willenborg, "Mu-and tau-argus: Software for statistical disclosure control," 1996.

[15] Data and Privacy Lab. The University of Texas at Dallas. Manual for anonymization toolbox [Online]. Availabe: http://cs.utdallas.edu/dspl/cgibin/toolbox/anonManual.pdf

[16] K. LeFevre, D. J. DeWitt, and R. Ramakrishnan, "Incognito: efficient full-domain k-anonymity," in *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*, SIGMOD '05, (New York, NY, USA), 2005, pp. 49-60, ACM.

[17] J. Litton, J. Muilu, A. Bj ̈orklund, A. Leinonen, and N. Pedersen, "Data modeling and data communication in genomeutwin," *Twin Res*, vol. 6, no. 5, pp. 383-90, 2003.

[18] J. Muilu, L. Peltonen, and J. Litton, "The federated database-a basis for biobank-based post-genome studies, integrating phenome and genome data from 600,000 twin pairs in Europe.," *Eur J Hum Genet*, vol. 15, no. 7, pp. 718-723, 2007.

[19] G. Ölund, P. Lindqvist, and J.-E. Litton, "Bims: An information management system for biobanking in the 21st century," *IBM Syst. J.*, vol. 46, pp. 171-182, Jan. 2007.

[20] B. Riedl, V. Grascher, and T. Neubauer, "A secure e-health architecture based on the appliance of pseudonymization," *JSW*, vol. 3, no. 2, pp. 23-32, 2008.

[21] C. A. Shoniregun, K. Dube, and F. Mtenzi, "Electronic healthcare information security," *Advances in Information Security*, vol. 53, Springer, 2010.

[22] Secure Hash Standard, Washington: National Institute of Standards and Technology, 2002. Federal Information Processing Standard 180-2.

[23] N. I. of Standards and Technology, Advanced Encryption Standard, NIST FIPS PUB 197, 2001.

[24] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "l-diversity: Privacy beyond k-anonymity," *ACM Trans. Knowl. Discov. Data*, vol. 1, Mar. 2007.

[25] C. Dwork, "Differential privacy," *ICALP*, vol. 2, pp. 1-12, Springer, 2006.

[26] E. Plugge, T. Hawkins, and P. Membrey, *The Definitive Guide to MongoDB: The NoSQL Database for Cloud and Desktop Computing*, Berkely, CA, USA: Apress, 1st ed., 2010.

Ali Gholami is a PhD student at the Royal Institute of Technology. His research interests include the use of data structures and algorithms to build adaptive data management systems for grid and cloud computing. Another area of his research focuses on the security concerns associated with cloud computing. He is currently exploring strong and usable security factors to enable researchers to process sensitive data in the cloud.

Erwin Laure is a Professor in Computer Science and Director of PDC-Center for High Performance Computing Center at KTH, Stockholm. Prior to this position he was the Technical Director of the "Enabling Grids for E-Science in Europe (EGEE)" project working at CERN. He is the Coordinator of the EC-funded "ScalaLife" project and actively involved in major e-infrastructure projects (EGI, PRACE, EUDAT). His research interests include programming environments, languages, compilers and runtime systems for parallel and distributed computing, particularly exascale computing, as well as grid and cloud computing with a focus on data management.

Peter Somogyi is a Postdoctoral researcher at Karolinska Institutet. Prior to join KI he worked in CERN. His research interests are including privacy-preservation of sensitive data.

Ola Spjuth is Docent in Bioinformatics and works at Uppsala University and Karolinska Institutet. He obtained his PhD from Uppsala University in 2010 and was postdoctoral fellow at Karolinska Institutet and Institute for Molecular Medicine Finland (FIMM). His main research interests are in e-Science methods for improving predictive modeling in drug discovery and cancer research.

Salman Niazi is a PhD candidate the Royal Institute of Technology. He is interested in addressing the problems of security, scalability and strong consistency in distributed systems.

Jim Dowling is an Associate Professor at KTH – the Royal Institute of Technology. His research is in the area of distributed systems, where his main research interests are in applying mechanisms from complex systems and self-organization theory to build better computer systems. He is also active in the area of Big Data for genomics.