Grande Region Security and Reliability Day 2015

March 11, 2015, Trier, Germany

Table of Contents

Michael Backes, Praveen Manoharan, and Pascal Berrang.
How well do you blend into the crowd? – d-convergence: A novel paradigm for quantifying privacy in the age of Big-Data

Huu-Hiep Nguyen, Abdessamad Imine, and Michael Rusinowitch.
Anonymizing Social Graphs via Uncertainty Semantics

Marcos Cramer, Jun Pang, and Yang Zhang.
A Logical Approach to Restricting Access in Online Social Networks

Michael Backes, Pascal Berrang and Praveen Manoharan.
Assessing the Effectiveness of Countermeasures Against Authorship Recognition

Walid Belkhir, Maxime Bride, Yannick Chevalier, and Michael Rusinowitch.
Secure Service Mediator Synthesis with Parametrized Automata

Daniel Fett, Ralf Kuesters, and Guido Schmitz.
Analyzing the BrowserID SSO System with Primary Identity Providers Using an Expressive Model of the Web

Tim Ruffing, Aniket Kate, and Dominique Schrder.
One Bitcoin at the Price of Two – Preventing Double-Spending and Equivocation using Bitcoin

Tim Ruffing, Pedro Moreno-Sanchez, and Aniket Kate.
CoinShuffle: Practical Decentralized Coin Mixing for Bitcoin

Pedro Moreno-Sanchez, Aniket Kate, Matteo Maffei, and Kim Pecina.
Privacy Preserving Payments in Credit Networks. Enabling trust with privacy in online marketplaces

Andreas Zeller.
Mining Apps for Abnormal Usage of Sensitive Data

Li Li, Tegawend F. Bissyand, Jacques Klein, and Yves Le Traon.
Using an Instrumentation based Approach to Detect Inter-Component Leaks in Android Apps

Matteo Maffei, Giulio Malavolta, Manuel Reinert, and Dominique Schroeder.
Privacy and Access Control for Outsourced Personal Records

Marcos Cramer and Agustin Ambrossio.
A logic of trust for reasoning about delegation and revocation

Ralf Kuesters and Tomasz Truderung.
Saving Re-Encryption Randomized Partial Checking Mix Nets for Risk-Avoiding Adversaries

Michael Backes, Aniket Kate, Praveen Manoharan, Sebastian Meiser, and Esfandiar Mohammadi.
MATor: Towards Measuring the Degree of Anonymity in Tor

Jasmin Christian Blanchette and Andrei Popescu.
Isabelle and Security

Michael Backes and Praveen Manoharan.
Sensitivity Assessment of Personal Information in Heterogeneous Data

Cesare Bartolini, Gabriela Gheorghe, Andra Giurgiu, Mehrdad Sabetzadeh and Nicolas Sannier.
Assessing IT Security Standards Against the Upcoming GDPR for Cloud Systems

Dayana Spagnuelo and Gabriele Lenzini.
Security on medical data sharing: a literature review

Phu H. Nguyen.
A System of Model-Driven Security Design Patterns

How well do you blend into the crowd?
-
d-convergence: A novel paradigm for quantifying privacy in the age of Big-Data

Michael Backes
CISPA, Saarland University
backes@cs.uni-saarland.de

Pascal Berrang
CISPA, Saarland University
berrang@cs.uni-saarland.de

Praveen Manoharan
CISPA, Saarland University
manoharan@cs.uni-saarland.de

***Abstract*—The advent of the Big-Data paradigm and the thereby emerging personalized tracking and monetization of personal information have amplified the privacy concerns of Internet users. Privacy-enhancing technologies struggle to keep pace with this Internet-scale development, and currently *lack even the basic methodology* to assess privacy in this world of rapid dissemination of unstructured, heterogeneous data with many involved actors. We refer to this problem as Big-Data privacy. Existing privacy models (k-anonymity, t-closeness, or the currently most popular notion of Differential Privacy) are inherently inadequate to reason about Big-Data privacy: they require an a-priori structure and classification of the data under consideration, and they disregard adversaries that utilize ubiquitously available background knowledge to infer further privacy-sensitive information.**

In this paper, we develop a user-centric privacy model for reasoning about Big-Data privacy. Our model constitutes a reinterpretation of statistical language models that are predominantly used in the information retrieval (IR) community to characterize documents with regard to their information content, and it explicitly leverages ideas from IR to cope with arbitrary (unstructured, heterogeneous) data in dynamically changing contexts. At the core of the model is our new notion of d-convergence, which measures the similarity of entities in a given setting, and hence allows us to derive bounds on the probability for a given entity to be singled out from its peers. This in particular entails a novel definition of Big-Data privacy based on indistinguishability of entities. We demonstrate the applicability of our privacy model on a collection of 40 million comments collected from the Online Social Network Reddit, which we stripped down to 15 million comments for our evaluation on two Dell PowerEdge R820 with 64 virtual cores each.

I. Introduction

The Internet has undergone dramatic changes in the last two decades, evolving from a mere communication network to a global multimedia platform in which billions of users not only actively exchange information, but increasingly conduct sizable parts of their daily lives. While this transformation has brought tremendous benefits to society, it has also created new threats to online privacy that existing technology is failing to keep pace with. Users tend to reveal personal information without considering the widespread, easy accessibility, potential linkage and permanent nature of online data. Many cases reported in the press show the resulting risks, which range from public embarrassment and loss of prospective opportunities (e.g. when applying for jobs or insurance), to personal safety and property risks (e.g. when sexual offenders or burglars learn users' whereabouts online). The resulting privacy awareness and privacy concerns of Internet users have been further amplified by the advent of the Big-Data paradigm and the aligned business models of personalized tracking and monetizing personal information in an unprecedented manner.

Developing a suitable methodology to reason about Big-Data privacy, as well as corresponding tool support in the next step, requires at its core a formal privacy model for assessing and quantifying to what extent a user is disseminating private information on the Internet. Any adequate privacy model needs to live up to the now increasingly dynamic dissemination of unstructured, heterogeneous user content on the Internet: While users traditionally shared information mostly using public profiles with static information about themselves, nowadays they disseminate personal information in an unstructured, highly dynamic manner, through content they create and share (such as blog entries, user comments, a "Like" on Facebook), or through the people they befriend or follow. Furthermore, ubiquitously available background knowledge about a dedicated user needs to be appropriately reflected within the model and its reasoning tasks, as it makes it possible to decrease a user's privacy by inferring further sensitive information. As an example, Machine Learning and other Big-Data analysis techniques provide comprehensive approaches for profiling a user's actions across multiple online social networks, up to a unique identification of a given user's profiles for each such network.

As of now, *even the basic methodology is missing* for offering users technical means to comprehensively assess the privacy risks incurred by their data dissemination, and their daily online activities in general. Existing privacy models such as k-anonymity [1], l-diversity [2], t-closeness [3] and the currently most popular notion of Differential Privacy [4] follow a database-centric approach that is inherently inadequate to meet the requirements outlined in the previous paragraph.

A. Contribution

We develop a novel formal privacy model that is based on the concept of statistical language models, which is the predominantly used technique in the Information Retrieval (IR) community for characterizing documents with regard to their information content [5], [6], [7], [8], [9]. Grounding our model upon such statistical models allows us to cope with unstructured, heterogeneous data, as well as highly dynamic content generation. Moreover, it allows us to seamlessly incorporate future advances from IR research and other Big-Data technologies into our model.

Our model defines and quantifies privacy by utilizing the notion of entity similarity, i.e., an entity is private in a collection of entities if it is sufficiently similar to its peers. Formally, this intuition is captured by defining corresponding statistical models that allow us to characterize entities based on the information they have disseminated publicly and based on ubiquitously available background knowledge about these entities. At the technical core of our model is the new notion of d-convergence, which measures the similarity of entities within a larger group of entities. It hence provides the formal grounds to quantify the ability of any single entity to blend into the crowd, i.e., to hide amongst peers.

In contrast to existing models, we do not have to differentiate between non-sensitive and sensitive attributes, but rather start from the assumption that all data is equally important and can lead to privacy risks. More specifically, our model captures the fact that the sensitivity of attributes is highly context-dependent, i.e., attributes can be or become sensitive for a specific entity when interacting with its peers.

We show that our model and its underlying notion of d-convergence implies existing privacy notions if one considers a setting with *structured* data only: we define a suitable transformation of our statistical model to a statistical database and subsequently show that a d-convergent database is also t-close, and using previous results, therefore is also differentially private.

Our privacy model is furthermore capable of assessing privacy risks specifically for single entities. To this end, we extend the notion of d-convergence to the novel notion of (k, d)-privacy, which allows for entity-centric privacy assessments by requiring d-convergence in the local neighborhood of a given entity. This definition thus allows us to make user-centric privacy assessments and provide lower bounds for an individual user's privacy irrespective of the whole data set, i.e., these bounds stay valid even when the set is enlarged, e.g., by including new users. Our concepts for extending d-convergence to (k, d)-privacy, and thereby achieving robust privacy guarantees for individual users, are of independent interest and can be applied to existing privacy notions as well.

Finally, we present an instantiation of our privacy model for the important use case of analyzing user-generated text content in order to characterize specific user profiles. We use unigram frequencies extracted from user-generated content as user attributes, and we subsequently demonstrate that the resulting unigram model can indeed be used for quantifying the degree of anonymity of—and ultimately, for differentiating—individual entities. To validate our statistical model approach for evaluating privacy characteristics in real-world settings, we apply this unigram model to a collection of 40 million comments collected from the Online Social Network Reddit, which we stripped down to 15 million comments to keep the evaluation tractable. The computations were performed on two Dell PowerEdge R820 with 64 virtual cores each at 2.60GHz over the course of six weeks.

REFERENCES

[1] L. Sweeney, "K-Anonymity: A Model for Protecting Privacy," *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 10, no. 5, pp. 557–570, 2002.

[2] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "L-Diversity: Privacy Beyond K-anonymity," *ACM Trans. Knowl. Discov. Data*, vol. 1, no. 1, 2007.

[3] N. Li and T. Li, "t-closeness: Privacy beyond k-anonymity and -diversity," in *In Proceedings of the 23rd International Conference on Data Engineering (ICDE*, 2007.

[4] C. Dwork, "Differential Privacy: A Survey of Results," in *Proceedings of the 5th International Conference on Theory and Applications of Models of Computation*, 2008, pp. 1–19.

[5] J. M. Ponte and W. B. Croft, "A Language Modeling Approach to Information Retrieval," in *Proceedings of the 21st Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 1998, pp. 275–281.

[6] V. Lavrenko and W. B. Croft, "Relevance based language models," in *Proceedings of the 24th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, 2001, pp. 120–127.

[7] C. Zhai and J. Lafferty, "A study of smoothing methods for language models applied to information retrieval," *ACM Transactions on Information Systems (TOIS)*, vol. 22, no. 2, pp. 179–214, 2004.

[8] . Uzuner and B. Katz, "A Comparative Study of Language Models for Book and Author Recognition," in *Natural Language Processing IJCNLP 2005*, 2005, pp. 969–980.

[9] C. Zhai, "Statistical language models for information retrieval a critical review," *Found. Trends Inf. Retr.*, vol. 2, no. 3, pp. 137–213, 2008.

Anonymizing Social Graphs via Uncertainty Semantics

Hiep H. Nguyen, Abdessamad Imine, and Michaël Rusinowitch
LORIA/INRIA Nancy-Grand Est, France
Email: {huu-hiep.nguyen,michael.rusinowitch}@inria.fr, abdessamad.imine@loria.fr

I. Introduction

Graphs represent a rich class of data observed in daily life where entities are described by vertices and their connections are characterized by edges. With the emergence of increasingly complex networks, the research community requires large and reliable graph data to conduct in-depth studies. However, this requirement usually conflicts with privacy rights of data contributing entities. Naive approaches like removing user ids from a social graph are not effective, leaving users open to privacy risks, e.g. re-identification attacks [1] [4]. Therefore, many graph anonymization schemes have been proposed[1].

Given an unlabeled undirected graph, the existing anonymization methods fall into five main categories. The first category includes *random* addition, deletion and switching of edges. The methods in the second category provide k-anonymity [6] by *deterministic* edge additions or deletions, assuming attacker's background knowledge regarding certain properties of its target nodes. The third class of techniques, *generalization*, cluster nodes into super nodes of size at least *k*. The methods in the fourth category assign edge probabilities to add uncertainty to the true graph. The edges probabilities may be computed explicitly as in [2] or implicitly via random walks [5]. Finally, several schemes for private graph data release are based on differential privacy [3]. Note that the third and fourth categories induce *possible world* semantics, i.e., we can retrieve sample graphs that are consistent with the anonymized output graph.

The fourth category is a recent class of methods which leverage the semantics of edge probability to inject uncertainty to a given deterministic graph, converting it into an uncertain one. Most of the schemes in this category are scalable, i.e. runnable on million-scale graphs or more. As an example, Boldi et al. [2] introduced the concept of *(k,ϵ)-obfuscation* (denoted as (k, ϵ)-obf), where $k \geq 1$ is a desired level of obfuscation and $\epsilon \geq 0$ is a tolerance parameter. However, the pursuit for minimum standard deviation σ in (k,ϵ)-obf has high impact on node privacy and high privacy-utility tradeoff. Edge rewiring method based on random walks (denoted as *RandWalk*) in [5] also introduces uncertainty to edges. This scheme suffers from high lower bounds for utility error despite its excellent privacy-utility tradeoff.

Motivated by *(k,ϵ)-obf* and *RandWalk*, we propose in this work a general model for anonymizing graphs based on edge uncertainty. Both (k,ϵ)-obf and RandWalk are captured by the model. We point out disadvantages in (k,ϵ)-obf and RandWalk, the tradeoff gap between them and present several elegant techniques to fill this gap.

Our contributions are summarized as follows:

- We propose a general model called *uncertain adjacency matrix* (UAM) for anonymizing graphs via edge uncertainty semantics. The key property of this model is that expected degrees of all nodes must be unchanged.
- We show how *(k,ϵ)-obf* and *RandWalk* fit into UAM and analyze their disadvantages. Then we describe *Mixture* as a simple mitigation.
- We introduce the *Variance Maximizing* (MaxVar) scheme that satisfies all the properties of UAM. It achieves good privacy-utility tradeoff by using two key observations: nearby potential edges and maximization of total node degree variance via a simple quadratic program. Another advantage of MaxVar is its full applicability to directed graphs.
- We conduct a comparative study of aforementioned approaches in a empirical privacy-utility framework by putting forward the distortion measure. We show the effectiveness of our gap-filling solutions on three real million-scale graphs.

II. Proposed Model and Algorithms

Uncertain Adjacency Matrix (UAM) Given the true undirected graph G_0 (Fig. 1a), an uncertain graph $\mathcal{G}$ (Fig. 1b) constructed from G_0 must have its uncertain adjacency matrix $\mathcal{A}$ satisfying

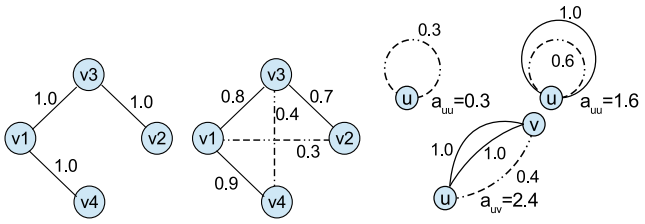
1) $\mathcal{A}_{ij} = \mathcal{A}_{ji}$ (symmetry);
2) $\mathcal{A}_{ij} \in [0, 1]$ and $\mathcal{A}_{ii} = 0$ (no multiedges or selfloops);
3) $\sum_{j=1}^{n} \mathcal{A}_{ij} = d_i(G_0) \;\; i = 1..n$, (*expected degrees* of all nodes must be unchanged).

While the constraints (1) and (2) are straightforward for uncertain undirected graph, the third constraint is novel and central to our model of UAM. It stems from the need of preserving the degree sequence of graph which is useful for degree distribution estimation. In terms of *dK-series*, the degree sequence is *d1-series*, the first moment of graph.

By relaxing (2) to (2'): $\mathcal{A}_{ii} \geq 0$ and $\mathcal{A}_{ij} \geq 0$, we allow graphs with *selfloop*s and *multiedge*s (Fig. 1c).

Via the model of UAM with the constraint of unchanged expected degree for all nodes, we show that *(k,ϵ)-obf*, *RandWalk* and *MaxVar* fit well into the UAM model and explain

[1]This work is accepted to appear in ASIACCS 2015, Singapore



(a) (b) (c)

Fig. 1: (a) True graph (b) An obfuscation with potential edges (dashed) (c) Semantics of selfloops (left), multi-selfloops (right) and multiedges (bottom) in UAM

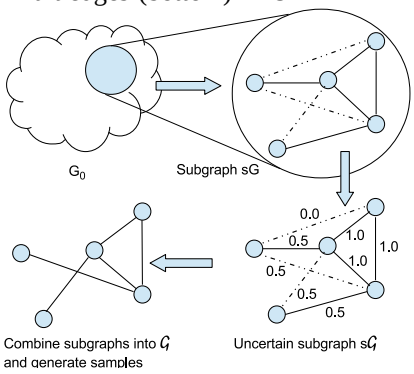


Fig. 2: MaxVar approach

how MaxVar fills the gap between (k,ϵ)-obf and RandWalk by comparing the total degree variance. As future work, we aim at novel constructions based on the proposed UAM.

Variance Maximizing Scheme (MaxVar) The intuition behind the new approach MaxVar is to formulate the perturbation problem as a *quadratic programming* problem. Given the true graph G_0 and the number of potential edges allowed to be added, the scheme has three phases. The first phase tries to partition G_0 into s subgraphs, each one with a given number of potential edges connecting nearby nodes (default distance 2, i.e. *friend-of-friend*). The second phase formulates a quadratic program for each subgraph with the constraint of unchanged node degrees to produce the uncertain subgraphs $s\mathcal{G}$ with maximum edge variance. The third phase combines the uncertain subgraphs $s\mathcal{G}$ into $\mathcal{G}$ and publishes several sample graphs. The algorithm is illustrated in Fig. 2.

Comparison of Schemes All the schemes mentioned in the current work are about *node-level* anonymization. It means we directly manipulate the nodes and their edges. In differentially private schemes such as [7], node signatures are indirectly anonymized via noisy graph statistics/models and graph regeneration. Therefore, all nodes' identities in the output cannot be linked to the original graphs. The incorrectness measure is not applicable, so we exclude such schemes from the comparison (Table I). Only MaxVar and EdgeSwitch satisfy all three properties (1),(2) and (3).

III. Experimental Results

Figures 3a,3b and 3c show that while RandWalk, RandWalk-mod have the best tradeoffs, they suffer from high lower bounds for utility. In other words, if the dataset allows higher privacy risk for better utility (lower rel.err) then the usage of two random walk based solutions may be limited. The simple solution *Mixture* also fills the gap.

In addition to the re-identification scores $H1$ and $H2_{open}$, we also compute ϵ for $k \in \{30, 50, 100\}$ to have a fair comparison with (k, ϵ)-obf. *MaxVar* has the best (k, ϵ) scores. The number of potential edges used in MaxVar could be 20% of $|E_{G_0}|$, much less than that of (k, ϵ)-obf (100% for $c = 2$ [2]).

TABLE I: Comparison of schemes ($\mathcal{A}$: uncertain matrix)

Scheme	Prop.1	Prop.2	Prop.3	$\mathcal{A}$	Directed
RandWalk-mod	○	×	○	○	×
RandWalk [5]	○	○	×	○	×
EdgeSwitch	○	○	○	×	○
(k, ϵ)-*obf* [2]	○	○	×	○	○
MaxVar	○	○	○	○	○
Mixture	depends on the mixed scheme				
Partition	depends on the scheme used in subgraphs				

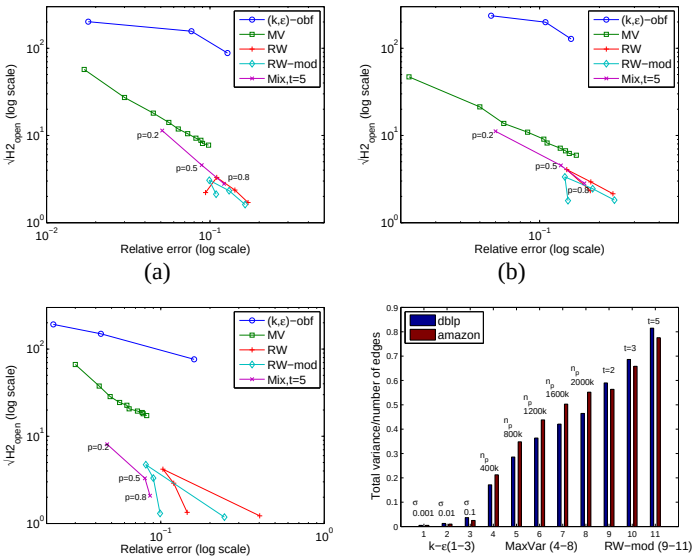


Fig. 3: Tradeoff (log-log) (a) dblp (b) amazon (c) youtube. (d) Comparison of Total Variance (TV)

References

[1] L. Backstrom, C. Dwork, and J. Kleinberg. Wherefore art thou r3579x?: anonymized social networks, hidden patterns, and structural steganography. In *WWW*, pages 181–190. ACM, 2007.

[2] P. Boldi, F. Bonchi, A. Gionis, and T. Tassa. Injecting uncertainty in graphs for identity obfuscation. *Proceedings of the VLDB Endowment*, 2012.

[3] C. Dwork. A firm foundation for private data analysis. *Communications of the ACM*, 54(1):86–95, 2011.

[4] M. Hay, G. Miklau, D. Jensen, D. Towsley, and P. Weis. Resisting structural re-identification in anonymized social networks. *VLDB Endowment*, 2008.

[5] P. Mittal, C. Papamanthou, and D. Song. Preserving link privacy in social network based systems. In *NDSS*, 2013.

[6] L. Sweeney. k-anonymity: A model for protecting privacy. *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 10(05):557–570, 2002.

[7] Q. Xiao, R. Chen, and K.-L. Tan. Differentially private network data release via structural inference. In *KDD*, pages 911–920. ACM, 2014.

A Logical Approach to Restricting Access in Online Social Networks

Marcos Cramer, Jun Pang, Yang Zhang

University of Luxembourg, Luxembourg

***Abstract*—Nowadays in popular online social networks users can blacklist some of their friends in order to disallow them to access resources that other non-blacklisted friends may access. We identify three independent binary decisions to utilize users' blacklists in access control policies, resulting into eight access restrictions. We provide syntactical transformations to rewrite a hybrid logic access control formula when fixing an access restriction. This enables a flexible and user-friendly approach for restricting access in social networks. We develop efficient algorithms for enforcing a subset of access control policies with restrictions. The effectiveness of the access restrictions and the efficiency of our algorithms are evaluated on a Facebook dataset.**

I. Introduction

Online social networks (OSNs) have been the dominating applications in the Internet during the past few years. A user can share a lot of information or resources in OSNs, such as his personal profile and photos. In addition, OSNs have provided access control schemes for users to decide who can view their resources. The access control schemes in OSNs are relationship-based. In simple terms, a user can define access control policies to allow others who are in a certain relationship with him to access his resources.

Sometimes a user can be bothered by others, e.g., due to harassment or different political views. To deal with this, major OSN companies have provided functionalities to allow a user to put someone on his *blacklist*. In Facebook, if a user only allows his friends to view his profile, then friends on his blacklist are disallowed to access his profile directly. In this way, blacklists can be treated as orthogonal to access control policies. However, the use of blacklists for restricting access in OSNs has not been well-understood and formally studied. For instance, suppose Alice and Bob are friends and Charlie is on Bob's blacklist. If Alice wants to share her photo with her friends of friends, should she *also* consider Bob's blacklist to deny Charlie's access? To address such research problems, we propose a logical approach to formalizing blacklist and its utilization in access control policies.

II. Preliminaries

A social network is modeled as a directed graph. Each user represents a node, an edge exists between two users if they are in a certain relationship, for instance, friendship (f). In our context, blacklists are modeled as a special relationship type (b). In a sample social graph depicted in Fig. 1, user B is A's friend and J is on A's blacklist.

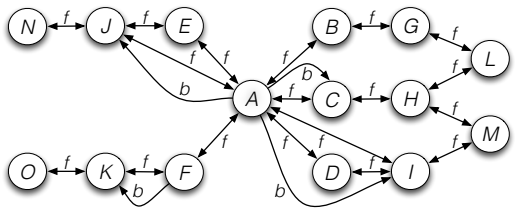


Fig. 1: A social graph example

The owner of a resource can specify an access control policy for determining which users have access to the resource. We adopt the hybrid logic in [1] to specify policies. If an owner only allows his friends or friends of friends to view his profile, then the policy is expressed as $@_{\mathsf{own}}\langle f\rangle\mathsf{req} \vee @_{\mathsf{own}}\langle f\rangle\langle f\rangle\mathsf{req}$, where the two variables own and req represent the owner and the requester, respectively. We refer the policy that regulates the qualified requesters to be 2 (3) friend steps away from the owner as 2-depth (3-depth) policy.

III. Restricting Access in OSNs

Blacklist can be treated orthogonal to access control policies. The basic requirement is that u_{req} (the requester) should never be on u_{own}'s (the owner) blacklist. Beyond this, there exist other decisions to make when blacklist-restricting access control policies. We classify blacklist-restrictions into three dimensions by considering the following questions: (1) whose blacklist should be used, (2) where blacklists should be applied, and (3) how many paths need to be considered.

Whose blacklists should be used? It is clear that the blacklist of u_{own} should always be considered for blacklist-restricting policies, i.e., the user following u_{own} on a path from u_{own} to u_{req} cannot be on u_{own}'s blacklist. Besides, other users' blacklists can be considered as well.

If u_{own} wants the blacklists of everyone on the path to be considered for blacklist-restricting an access control policy, u_{own} should *globally* blacklist-restrict the policy (GL). If on the other hand u_{own} only wants his own blacklist to be considered, he should *locally* blacklist-restrict the access control policy (LO). We name this restriction dimension *globality*.

Where should blacklists be applied? The requester should never be on u_{own}'s blacklist. Besides, u_{own} may want no one on a path from him to u_{req} to be on his blacklist, i.e., he may want to consider his blacklist on the whole path.

If u_{own} wants no one on a path in the set of paths witnessing the access control policy to be on his blacklist, he should

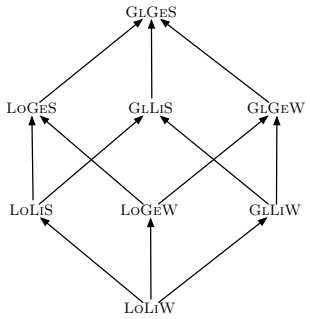


Fig. 2: Black-restriction lattice

perform a *general* blacklist-restriction to the policy (GE). If on the other hand u_{own} only wants u_{req} not to be on his blacklist, he should perform a *limited* blacklist-restriction to the policy (LI). We name this restriction dimension *generality*.

How many paths need to be considered? Having fixed the decisions for the previous two dimensions, u_{own} has determined which set of paths can be considered *free of blacklist problems*. There can still be several paths from u_{own} to u_{req}, some are free of blacklist problems while others are not.

If u_{own} just wants there to be some set of paths free of blacklist problems witnessing the access control policy, he should *weakly* blacklist-restrict the access control policy (W). If on the other hand he wants that every set of paths witnessing the policy should be free of blacklist problems, he should *strongly* blacklist-restrict the access control policy (S). We name this restriction dimension *strength*.

The eight ways of forming blacklist-restricted policies establish a lattice as shown in Fig. 2. If a user's access is denied by one of the blacklist-restricted policies, then the same user's access is denied by any blacklist-restricted policy above this policy in the blacklist-restriction lattice. To illustrate the eight different restrictions, we assume that user A (in Fig. 1) defines a 3-depth policy. Tab. I summarizes the users whose access are denied under different blacklist-restrictions.

Restriction	Denied users	Restriction	Denied users
LOLIW	H	LOLIS	H, L, M
LOGEW	H, M, N	LOGES	H, L, M, N
GLLIW	H, O	GLLIS	H, L, M, O
GLGEW	H, M, N, O	GLGES	H, L, M, N, O

TABLE I: Denied users under different blacklist-restrictions

IV. Syntactical Transformation

In practice, OSN users are not competent in using hybrid logic for restricting their access control policies with blacklists. Therefore, we propose a syntactical transformation algorithm to rewrite a non-restricted policy to the restricted one. The model-checking algorithm from [1] can then be applied for evaluating the policy. Given a policy formula written in the hybrid logic, our algorithm inserts hybrid logic operators into the formula to refer to the nodes of the paths satisfying the formula. We then use the bound variables (specified together with other operators of the hybrid logic) to formulate the conditions of the specified blacklist-restriction. With syntactical transformation, a user only needs to define a non-restricted policy and specifies the restriction, our algorithm will generate the corresponding restricted policy automatically.

V. Path Evaluation Algorithms

A user normally focuses on the length of the path between him the potential requesters when defining access control policies. To evaluate policies of this kind, instead of syntactically transforming their formulas to longer ones and applying model-checking techniques, we can search for the qualified path(s) from u_{own} to u_{req} in the social network. During the path-finding process, we can perform optimizations such as filtering out the users who are on u_{own}'s blacklist on-the-fly.

We develop algorithms for policy evaluation under each restriction and test their efficiency on a Facebook dataset [2] for 2-depth and 3-depth polices. The metric we adopt is the ratio of the time for running a restricted policy and the time for running the non-restricted one. For each user, we sample five different ratios of his friends to be on his blacklist. The results for restrictions GLLIW and GLLIS are presented in Fig. 3. With the increase of blacklist ratio, checking path policies under weak restrictions is getting faster. This is because we filter out the unqualified edges when searching for paths. On the other hand, running 3-depth policies under strong restrictions cost twice more time and the 2-depth case only requires around 30% overhead. We study the effectiveness of the restrictions through the number of users they deny. As shown in Fig. 4, when blacklist ratio is 10%, under restriction LOGES, only 60% (3-depth policy) of the qualified users under the non-restricted policy can access, while the ratio is more than 95% for LOGEW. This is because strong restrictions require every path from the owner to the requester to satisfy the restrictions from the other two dimensions.

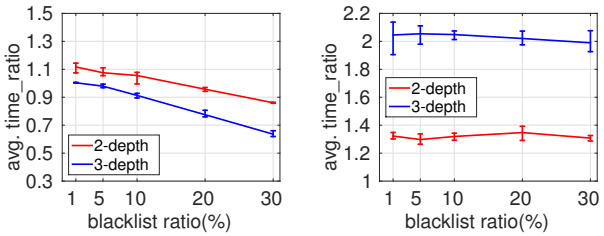


Fig. 3: Efficiency for GLLIW (left) and GLLIS (right)

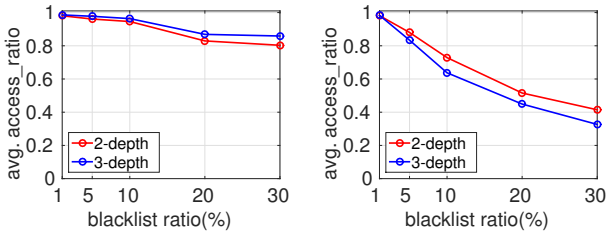


Fig. 4: Effectiveness for LOGEW (left) and LOGES (right)

References

[1] G. Bruns, P. W. L. Fong, I. Siahaan, and M. Huth, "Relationship-based access control: its expression and enforcement through hybrid logic," in *CODASPY 2012*.

[2] J. J. McAuley and J. Leskovec, "Learning to discover social circles in ego networks," in *NIPS 2012*.

Assessing the Effectiveness of Countermeasures Against Authorship Recognition

Michael Backes[1,2] Pascal Berrang[1] Praveen Manoharan[1]

[1]CISPA, Saarland University [2] MPI-SWS

backes@cs.uni-saarland.de
berrang@cs.uni-saarland.de
manoharan@cs.uni-saarland.de

ABSTRACT

Methods for authorship recognition were originally developed to aid in criminal investigations and attribution of historical texts. Nowadays, however, in an age in which the Internet has become the central platform for day-to-day social interactions and communication, authorship recognition technology can be abused to break the anonymity of users by identifying the authors of user-generated text content. While there have been recent advances in *adversarial stylometry*, which investigates the impact of *obfuscation* and *imitation* on current authorship recognition techniques, no comprehensive model for the assessment of countermeasure effectiveness currently exists.

In this work, we introduce a novel measure for assessing the context-dependent importance of writing style features in authorship recognition. From this measure, we furthermore derive an additional measure for assessing the effectiveness of authorship-recognition countermeasures by analysing how well these countermeasures reduce the importance of the affected features.

We then utilise these measures to conduct a large-scale evaluation of four semantics-retaining countermeasures and their combinations on a dataset of 923,997 comments from 3439 users collected from the online social network Reddit. We examine the practical impact of these countermeasures on the importance of standard writing style features in the context of Reddit's subreddits and explore the outcome of combining several countermeasures at the same time.

1. INTRODUCTION

During the last two decades, the Internet evolved from a simple communication network to a global multimedia platform which is part of our everyday life. On this platform, billions of users actively share data, even revealing personal information, without considering the consequences of the easy accessability and the permanent nature of their disseminated data. The detrimental consequences range from personalized advertisements and the sale of personal information up to threats concerning personal safety.

An intuitive, commonly pursued approach to separate sensitive information from one's personal identity, and thus protect one's privacy, would be to disseminate sensitive information only through anonymous or pseudonymous profiles, with the intention of decoupling a user's real-life identity from sensitive information posted under pseudonymous accounts. As literature has shown, however, this approach is not really effective since different profiles are typically linkable using common characteristics [6, 8, 1, 4, 5]. In particular, for user-generated text content, the writing style of a user is often unique across different profiles, and can thereby be used to attribute text content to its corresponding (seemingly anonymous) author [2]. Recent research has shown that this profile linkage can even be conducted at Internet scale [7].

Recent work on *adversarial stylometry* has tried to reduce the likelihood of correctly linking corresponding profiles by investigating the impact of *obfuscation* and *imitation* of text passages on current authorship recognition techniques. These works have mostly focused on the development of manual and semi-automated countermeasures in order to circumvent stylometry. However, none of these works is capable of assessing the actual effectiveness of these countermeasures. Hence, these works do not provide any insights on which countermeasures are particularly well-suited for a given context in which a certain text should be published. The absence of such results is, in particular, due to the lack of a rigorous model for assessing the effectiveness of various types of countermeasures on the identifiability of authors, which currently does not exist. In addition, developing a model of this kind might help in identifying the major challenges that research needs to overcome in order to provide fully-automated assistance for authorship obfuscation.

1.1 Contribution

In this work, we present a novel measure for assessing the importance of stylometric features for the identifiability of authors. We base this assessment on the privacy model introduced by Backes *et al.* [3], which provides a generic data model to cope with heterogeneous information using statistical models. We adapt and extend these statistical to fit our use case to authorship recognition, effectively defining a model for writing style that allows us to capture a comprehensive list of stylometric features, as introduced by Abbasi and Chen [1]. Overall, we develop a model of the authorship recognition problem that allows us to formally reason about authorship recognition in the open setting of the Internet.

We then derive how we can identify important stylometric features that significantly contribute to the identification of the correct author from the context in which text is published by using these writing-style models. We employ standard regression techniques to determine the weights of

each type of stylometric feature, which then correspond to their importance. From this importance assessment we then further derive the *gain* measure for the effectiveness of countermeasures against authorship identification by measuring how well they reduce the importance of stylometric features.

We apply this measure to assess the effectiveness of four countermeasures, namely synonym substitution, spell checking, special character modification and adding/removing misspellings. In this evaluation, we follow a general and comprehensive methodology that structures the evaluation process and is easily extensible for future evaluation.

We perform our experiments on a dataset of 923,997 comments by 3439 users collected from the online social network Reddit, and argue why Reddit's subreddit structure is particularly well suited for research in authorship recognition by automatically providing ground truth for our evaluations.

The computations were performed on three Dell PowerEdge R820s with 64 virtual cores each at 2.60GHz over the course of 2 weeks, assisted by Amazon Web Services (AWS) on the final stretch.

2. REFERENCES

[1] Ahmed Abbasi and Hsinchun Chen. Writeprints: A Stylometric Approach to Identity-level Identification and Similarity Detection in Cyberspace. *ACM Transactions on Information Systems (TOIS)*, 26(2):1–29, 2008.

[2] Sadia Afroz, Aylin Caliskan Islam, Ariel Stolerman, Rachel Greenstadt, and Damon McCoy. Doppelgänger finder: Taking stylometry to the underground. In *Proceedings of the 2014 IEEE Symposium on Security and Privacy*, pages 212–226, 2014.

[3] Michael Backes, Pascal Berrang, and Praveen Manoharan. How well do you blend into the crowd? - d–convergence: a novel paradigm for reasoning about privacy in the age of Big-Data. http://arxiv.org/abs/1502.03346, 2015. eprint arXiv:1502.03346 – cs.CR.

[4] Moshe Koppel, Jonathan Schler, and Shlomo Argamon. Computational Methods in Authorship Attribution. *Journal of the American Society for Information Science and Technology*, 60(1):9–26, 2009.

[5] Mikhail B. Malyutov. Authorship Attribution of Texts: A Review. In *General Theory of Information Transfer and Combinatorics*, pages 362–380. 2006.

[6] Thomas Corwin Mendenhall. The characteristic curves of composition. *Science*, pages 237–249, 1887.

[7] Arvind Narayanan, Hristo Paskov, Neil Zhenqiang Gong, John Bethencourt, Emil Stefanov, Eui Chul Richard Shin, and Dawn Song. On the feasibility of internet-scale author identification. In *Proceedings of the 2012 IEEE Symposium on Security and Privacy (S& P)*, pages 300–314, 2012.

[8] Özlem Uzuner and Boris Katz. A Comparative Study of Language Models for Book and Author Recognition. In *Natural Language Processing – IJCNLP 2005*, pages 969–980. 2005.

Secure Service Mediator Synthesis with Parametrized Automata

Walid Belkhir[1] and Maxime Bride[1] and Yannick Chevalier[2] and Michael Rusinowitch[1]

[1] INRIA Nancy–Grand Est & LORIA
walid.belkhir@inria.fr,maxime.bride@inria.fr,rusi@loria.fr
[2] Université Paul Sabatier & IRIT Toulouse ychevali@irit.fr

Service Oriented Architectures (SOA) consider services as self-contained components that can be published, invoked over a network and combined with other services through standardized protocols in order to dynamically build complex applications [1].Service composition is required when none of the existing services can fulfill some client needs but a suitable coordination of them would satisfy the client requests. The composition synthesis problem we consider can be stated as follows: given a client and a community of available services, compute a mediator which will enable communication between the client and the available services in such a way that each client request is forwarded to an appropriate service. We model services by parametrized automata (PA) where transitions are labelled by constraints on variables and the variables can be reset at some states [3,4]. We have considered in our previous work the simple case of untimed parametrized automata composition synthesis. Here, we introduce timed parametrized automata (TPA) and we incorporate security policies in the computation of the composition synthesis. Hence, the client and the available services exchange data ranging over an infinite domain and they are possibly subject to some *data* and *time constraints*.

Our method to synthesize a mediator is based on *simulation games*. These games allow us to compute a winning strategy in order to synthesize a suitable mediator. A simulation game is a TPA (like the client and the services) which simulates all the interactions between the client and the services (more precisely, between the client and the asynchronous product of the services). The simulation game can be influnced by a security policy expressed as a TPA too. We can express for instance that a file cannot be opened if another one is not closed or the fact that some message should not be circulated on the network. To compute the winning strategy, we transform the simulation game which is a TPA into a finite automaton. This is done in two steps : first, we abstract data of the TPA to obtain a timed automaton and then we use the classical region construction [2] to obtain a finite automaton. In order to abstract data of a TPA, we introduce a notion of quotient automaton which operates on a finite set of equivalence classes over data (analogous to regions for timed automata), to obtain an automaton over a finite alphabet and without guards on data. Since a mediator has to communicate with the client and the services, we have to get a TPA by reintroducing data and time constraints in the finite automaton above.

We achieve this transformation by using the available information from time and data regions and from the original simulation game.

In this paper, all these transformations are formally defined and the correction of all algorithms is proved. In Algorithm 1, we give the general mediator synthesizing algorithm where:

- `Game` computes a symbolic simulation game of a client TPA by the asynchronous product of the TPAs of available services.
- `Quotient` computes a timed automaton from the TPA of its argument.
- `Region` computes a finite automata from the quotient TPA of its argument.
- `Strategy` computes a winning strategy in the given automaton.
- $\texttt{Region}^{-1}$ computes a timed automaton from the finite automaton of its argument and a corresponding simulation game (which is a TPA).
- $\texttt{Quotient}^{-1}$ computes a TPA from the timed automaton of its argument.
- λ is the labelling function to associate a state of the corresponding automaton to the client or to the services.

Algorithm 1: Mediator synthesis algorithm

input : A client TPA $\mathcal{A}$, a community of services $\mathcal{A}_1, \ldots, \mathcal{A}_n$ and a policy $\mathcal{P}$
output: A mediator $\mathcal{M}$ as a TPA satisfying $\mathcal{P}$ that delegates the actions of $\mathcal{A}$ to an appropriate service among the community of services

1 $(\mathcal{G}, \lambda) \leftarrow \texttt{Game}(\mathcal{A}, \mathcal{A}_1 \otimes \ldots \otimes \mathcal{A}_n)$;
2 $(\mathcal{GP}, \lambda) = \mathcal{G} \overset{p}{\times} \mathcal{P}$;
3 $(\mathcal{GP}_{\approx}, \lambda) \leftarrow \texttt{Quotient}(\mathcal{GP})$;
4 $(\mathcal{GP}^r_{/\approx}, \lambda) \leftarrow \texttt{Region}(\mathcal{GP}_{\approx})$;
5 $(\mathcal{S}, \lambda) \leftarrow \texttt{Strategy}(\mathcal{GP}^r_{/\approx})$;
6 $(\mathcal{M}_{/\approx}, \lambda) \leftarrow \texttt{Region}^{-1}(\mathcal{GP}, \mathcal{S})$;
7 $(\mathcal{M}, \lambda) \leftarrow \texttt{Quotient}^{-1}(\mathcal{M}_{/\approx})$;

References

1. Gustavo Alonso, Fabio Casati, Harumi Kuno, and Vijay Machiraju. *Web services: Concepts, Architectures and Applications*. Springer-Verlag, 2004.
2. Rajeev Alur and David L. Dill. A theory of timed automata. *Theoretical Computer Science*, 126:183–235, 1994.
3. Walid Belkhir, Yannick Chevalier, and Michaël Rusinowitch. Parametrized automata simulation and application to service composition. *J. Symb. Comput.*, 69:40–60, 2015.
4. Walid Belkhir, Gisela Rossi, and Michaël Rusinowitch. A parametrized propositional dynamic logic with application to service synthesis. In *Tenth conference on "Advances in Modal Logic," Groningen, The Netherlands, August 5-8, 2014*, pages 34–53, 2014.

Analyzing the BrowserID SSO System with Primary Identity Providers Using an Expressive Model of the Web

Daniel Fett Ralf Küsters Guido Schmitz

University of Trier, Germany
{fett,kuesters,schmitzg}@uni-trier.de

Single sign-on (SSO) systems have become an important building block for authentication in the web. Over the last years, many different SSO systems have been developed, for example, OpenID, OAuth, and proprietary solutions such as Facebook Connect. These systems usually allow a user to identify herself to a so-called relying party (RP), which provides some service, using an identity that is managed by an identity provider (IdP), such as Facebook or Google.

Given their role as brokers between IdPs and RPs, the security of SSO systems is particularly crucial: Numerous attacks have shown that vulnerabilities in SSO systems usually compromise the security of many services (RPs) and users at once (see, e.g., [3]).

BrowserID [2] is a relatively new complex SSO system which allows users to utilize any of their existing email addresses as an identity. BrowserID, which is also known by its marketing name *Persona*, has been developed by Mozilla and provides decentralized and federated login, with the intent to respect users' privacy: While in other SSO systems (such as OpenID), by design, IdPs can always see when and where their users log in, Mozilla's intention behind the design of BrowserID was that such tracking should not be possible. Several web applications support BrowserID authentication. For example, popular content management systems, such as Drupal and WordPress allow users to log in using BrowserID. Also Mozilla uses this SSO system on critical web sites, e.g., their bug tracker Bugzilla and their developer network MDN.

The BrowserID implementation is based solely on native web technologies. It uses many new HTML5 web features, such as web messaging and web storage. For example, BrowserID uses the postMessage mechanism for cross-origin inter-frame communication (i.e., communication within a browser between different windows) and the web storage concept of modern browsers to store user data on the client side.

There are two modes for BrowserID: For the best user experience, email providers (IdPs) can actively support BrowserID; they are then called *primary IdPs*. For all other email providers that do not support BrowserID yet, the user can register her email address at a default IdP, namely Mozilla's `login.persona.org`, the so-called *secondary IdP*.

In previous work [1], we proposed a general and expressive Dolev-Yao style model for the web infrastructure. This web model is designed independently of a specific web application and closely mimics published (de-facto) standards and specifications for the web, for instance, the HTTP/1.1 standard, associated (proposed) standards (mainly RFCs), and the HTML5 W3C candidate recommendation. It is the most comprehensive web model to date. Among others, HTTP(S) requests and responses, including several headers, such as cookie, location, strict-transport-security (STS), and origin headers, are modeled. The model of web browsers captures the concepts of windows, documents, and iframes, including the complex navigation rules, as well as new technologies, such as web storage and cross-document messaging (postMessages). JavaScript is modeled in an abstract way by so-called scripting processes which can be sent around and, among others, can create iframes and initiate XMLHTTPRequests (XHRs). Browsers may be corrupted dynamically by the adversary.

Based on this general web model, we analyzed the security of the secondary IdP mode of BrowserID [1]. The analysis revealed several severe vulnerabilities, which have since been fixed by Mozilla.

Contributions of this Paper. In this paper, we study the primary mode of BrowserID. As mentioned before, in previous work we studied the simpler secondary mode of BrowserID only. The primary model studied here is much more complex than the secondary mode. It involves more components (such as an arbitrary set of IdPs, more iframes), a much more complex communication structure, and requires weaker trust assumptions (for example, some IdPs, and hence, the JavaScript they deliver, might be malicious).

More specifically, the contributions of this paper can be summarized as follows.

Extension of the Web Model. We slightly extend our web model proposed in [1]. We complement the modeling of the web storage concept of modern browsers by adding sessionStorage, which is (besides the already modeled localStorage) heavily used by BrowserID in its primary mode. We also extend the model to include a set of user identities (e.g., user names or email addresses) in addition to user secrets.

Authentication Attack and Security Proof for BrowserID. The authentication properties we analyze are central to any SSO system and correspond to those considered in [1]: i) the attacker should not be able to log in at an RP as an honest user and ii) the attacker should not be able to authenticate an honest user/browser to an RP with an ID not owned by the user (identity injection). While trying to prove these authentication properties for the primary mode of BrowserID, we discovered a new attack which violates property ii). Depending on the service provided by the RP, this could allow the attacker to track the honest user or to obtain user secrets. We confirmed the attack on the actual implementation and reported it to Mozilla, who acknowledged the attack. We note that this attack does not apply to the secondary mode.

We propose a fix and provide a detailed formal proof based on the (extended) web model which shows that the fixed system satisfies the mentioned authentication properties. This constitutes the most complex formal analysis of a web application based on an expressive model of the web infrastructure, in fact, as mentioned, the most comprehensive one to date. We note that other web models are too limited to be applied to BrowserID.

Privacy Attacks on BrowserID. As pointed out before, BrowserID was designed by Mozilla with the explicit intention to respect users' privacy, a property that was not studied in [1]. Unlike in other SSO systems, when using BrowserID, IdPs should not learn to which RP a user logs in. When trying to formally prove this property, we discovered attacks that show that BrowserID cannot live up to this claim. Our attacks allow malicious IdPs to check whether or not a user is logged in at a specific RP with little effort. Interestingly, one variant of these attacks exploits a browser side channel which, to our knowledge, has not received much attention in the literature so far. Just as in the case of authentication, we have confirmed the attacks on the actual implementation and reported them to Mozilla, who acknowledged the attacks. Unfortunately, the attacks exploit a design flaw of BrowserID that does not seem to be easily fixable without a major redesign.

Generic Web Security Properties. Our security analysis of BrowserID and the case study in [1] show that certain security properties of the web model need to be established in most security proofs for web standards and web applications. As another contribution of this paper, we therefore identify and summarize central security properties of generic web features in our extended model and formalize them in a general way such that they can be used in and facilitate future analysis efforts of web standards and web applications.

References

[1] D. Fett, R. Küsters, and G. Schmitz. An Expressive Model for the Web Infrastructure: Definition and Application to the BrowserID SSO System. In *S&P 2014*, pp. 673–688. IEEE Computer Society, 2014.

[2] Mozilla Identity Team. Persona. `https://login.persona.org`.

[3] R. Wang, S. Chen, and X. Wang. Signing Me onto Your Accounts through Facebook and Google: A Traffic-Guided Security Study of Commercially Deployed Single-Sign-On Web Services. In *S&P 2012*, pp. 365–379. IEEE Computer Society, 2012.

One Bitcoin at the Price of Two
Preventing Double-Spending and Equivocation using Bitcoin

Tim Ruffing
CISPA, Saarland University

Aniket Kate
CISPA, Saarland University

Dominique Schröder
CISPA, Saarland University

***Abstract*—In this work, we present a novel application of the Bitcoin currency network towards preventing equivocation (i.e., making conflicting statements to different honest parties) in distributed systems. In particular, we develop a non-equivocation functionality that prevents a rational attacker from equivocating by monetarily penalizing it if the equivocation is detected. We realize this goal in a *completely* decentralized manner using non-interactive deposits in Bitcoin and a novel primitive called *accountable authenticators*. We demonstrate the utility of the non-equivocation functionality by applying it to two real-life application scenarios: the protection of append-only logs in the public-key infrastructure, and data integrity in cloud storage as well as social networks.**

We also apply our non-equivocation functionality to the Bitcoin network itself. In particular, we propose a practical solution to the problem of double-spending in fast Bitcoin payments, where the payee cannot wait for tens of minutes to get the payment confirmed. Our double-spending prevention solution not only makes fast Bitcoin payments practical, but also significantly reduces the communication cost of fair MPC protocols based on Bitcoin.

Over the last half decade we have been observing an unprecedented and rather surprising growth of currency networks such as Bitcoin [1]. Their decentralized nature as well as their ability to perform transaction across the globe in a matter of minutes have been pivotal to the success of these networks so far.

In Bitcoin as well as in other crypto-currencies, users transfer coins by signing their transactions. These transactions, however, are confirmed only when they are included in the *blockchain*, which basically is a distributed time-stamping service based on proof-of-work (POW) performed by currency *miners*. Although it is possible for a malicious owner to sign over the same coin to multiple receivers through multiple transactions, this *double-spending* is prevented by eventually approving only one transaction to be added to the publicly verifiable blockchain. The peer-to-peer nature of Bitcoin communication network ensures that all published transactions and their POW-based order of execution decided by pools of miners are visible universally.

Nevertheless, due to POW computations and the decentralized nature of Bitcoin, the above transaction confirmation process takes tens of minutes to complete [2]. Therefore, Bitcoin, in its current form, is inappropriate for so called "fast transactions", i.e., transactions requiring fast clearing of payments. The Bitcoin community acknowledges the double-spending problem with fast payments, and suggests that "merchants who need to sell things automatically and instantly are most likely to adjust the price to include the cost of reversal fraud, or elect to use special insurance" [3]. At the same time, fast transactions such as paying in supermarkets or buying products from vending machines using bitcoins are in great demand.

Beyond the immediate utility for performing financial transactions, decentralized payment systems have been also found useful towards resolving fairness with traditional distributed systems. For example, Bitcoin has been used as a building block to define fair multi-party computation (MPC) protocols [4], [5]. Bitcoin is employed to create deposits (on bonds), which results in monetary losses to the parties behaving unfairly. Nevertheless, these solutions will be practical for real-life MPC systems only if double-spending preventing fast transactions are possible.

Although the above scenarios demonstrate the need to finding a reliable solution to the double-spending problem with crypto-currencies, their decentralized nature makes the tasks very challenging. With BitUndo [6], there is even now a double-spending provider available that offers to run sophisticated double-spending attacks as service.

As observed in [7], [8], it is possible to mitigate the double-spending problem up to some level by making some amendments to the receiver's behavior over the Bitcoin communication network. In presence of these modifications performing double-spending turns non-trivial; however, due to attacks such as the Finney attack [9] performed by a double-spending miner, the merchant can never be sure of protection from a determined attacker. A solution can be obtained using *interactive deposits* [10] between a payer and his payee such that the payee can claim the deposits after detecting the misbehavior by the payer; however, the deposit has to be created tens of minutes in advance, and this method completely restricts the payers from using the same deposit for multiple payees.

Double-spending in Bitcoin can be seen as an instance of *equivocation* [11], [12], which we speak of in general when a malicious party in a distributed system makes conflicting statements to different protocol parties. Therefore, a generic solution to the equivocation problem in distributed systems can easily be adopted to the double-spending problem in Bitcoin. In this work we address the equivocation problem in distributed systems, and then employ it as a step towards double-spending prevention in Bitcoin.

Contributions. We observe that the equivocation problem can be solved efficiently by using *non-interactive* deposits requiring no involvements from payees. Here, the payee can no longer claim the deposit, but instead can penalize the payer by making her lose the deposit to the Bitcoin miners. Non-interactive deposits are possible using a forthcoming feature in Bitcoin called `CHECKLOCKTIMEVERIFY` [13]. Nevertheless, non-interactive deposits are not sufficient to penalize equivocation or double-

spending as the attacker's credential (i.e., the signing key) employed while creating the deposit cannot to be made available to the victim through the Bitcoin network.

To overcome this issue, we introduce a cryptographic primitive *accountable authenticators* that binds a *statement* to a *context* in an accountable way. In the Bitcoin setting, a context of accountable authenticators corresponds to a coin, a statement refers to a transaction, and accountability means that whenever the user authenticates different transactions to the same coin, then any observer can extract a *private deposit key* (i.e., the private key for a Bitcoin deposit). For accountable authenticators it is important the private deposit key gives access only to a predetermined deposit such that in case of equivocation, the victim can open one deposit and cannot steal other coins of the equivocating malicious party. We propose a construction of accountable authenticators based on chameleon hash functions [14] and chameleon authentication trees [15], which is secure under the discrete logarithm assumption and can be of independent interest to the field of cloud security.

We combine non-interactive deposits in Bitcoin and accountable authenticators to design a practical solution to the equivocation problem in distributed systems. In particular, we employ the combination to ensure linearity properties in append-only logs for the public-key infrastructure, e.g., Certificate Transparency [16].

As a further application, we apply the above non-equivocation solution based on non-interactive deposits and accountable authenticators to mitigate the double-spending problem in Bitcoin. In particular, we demonstrate that double-spending can be effectively prevented in order to allow fast Bitcoin transactions. We can achieve this by combining non-interactive deposits with accountable authenticators, or alternatively, with an additional Bitcoin consensus rule as observed in [17], [18]. These changes could be implemented in the Bitcoin system using a backwards-compatible change of the consensus rules.

As an application, we observe that our fast transaction protocol can help to avoid waiting times in fair multi-party computation (MPC) protocols such as the fair lottery protocol [4], which improves their execution time drastically.

Finally, we evaluate the computation and communication cost of accountable authenticators as it is the most expensive component of our functionality. We expect authentication and verification algorithms to take approximately below 100 msec to complete in a practical scenario.

References

[1] S. Nakamoto, *Bitcoin: a peer-to-peer electronic cash system*, Technical report, 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf.

[2] *Confirmation*, Entry in Bitcoin Wiki. [Online]. Available: https://en.bitcoin.it/w/index.php?title=Confirmation&oldid=51011.

[3] *Bitcoin FAQ*. [Online]. Available: https://en.bitcoin.it/wiki/FAQ.

[4] M. Andrychowicz, S. Dziembowski, D. Malinowski, and L. Mazurek, "Secure multiparty computations on Bitcoin," in *Proceedings of the 2014 IEEE Symposium on Security and Privacy*, 2014, pp. 443–458.

[5] I. Bentov and R. Kumaresan, "How to use Bitcoin to design fair protocols," in *Advances in Cryptology – CRYPTO 2014*, 2014.

[6] *Bitundo*. [Online]. Available: http://www.bitundo.com.

[7] G. O. Karame, E. Androulaki, and S. Capkun, "Double-spending fast payments in Bitcoin," in *Proceedings of the 2012 ACM Conference on Computer and Communications Security*, vol. 2012, 2012, pp. 906–917.

[8] T. Bamert, C. Decker, L. Elsen, R. Wattenhofer, and S. Welten, "Have a snack, pay with bitcoins," in *13th IEEE International Conference on Peer-to-Peer Computing*, 2013, pp. 1–5.

[9] H. Finney, *Re: Best practice for fast transaction acceptance - how high is the risk?* Post on Bitcoin forum. [Online]. Available: https://bitcointalk.org/index.php?topic=3441.msg48384#msg48384.

[10] *Rapidly-adjusted (micro)payments to a pre-determined party*, Entry in Bitcoin Wiki. [Online]. Available: https://en.bitcoin.it/w/index.php?title=Contracts&oldid=50633 # Example _ 7 : _Rapidly - adjusted _ .28micro . 29payments_to_a_pre-determined_party.

[11] A. Clement, F. Junqueira, A. Kate, and R. Rodrigues, "On the (limited) power of non-equivocation," in *Proceedings of the 2012 ACM Symposium on Principles of Distributed Computing*, 2012, pp. 301–308.

[12] B.-G. Chun, P. Maniatis, S. Shenker, and J. Kubiatowicz, "Attested append-only memory: making adversaries stick to their word," in *Proceedings of Twenty-first ACM SIGOPS Symposium on Operating Systems Principles*, 2007, pp. 189–204.

[13] P. Todd, *OP_CHECKLOCKTIMEVERIFY*, Draft for Bitcoin Improvement Proposal. [Online]. Available: https://github.com/petertodd/bips/blob/checklocktimeverify/bip-checklocktimeverify.mediawiki.

[14] H. Krawczyk and T. Rabin, "Chameleon signatures," in *2000*, 2000.

[15] D. Schröder and H. Schröder, "Verifiable data streaming," in *ACM Conference on Computer and Communications Security*, 2012, pp. 953–964.

[16] B. Laurie, "Certificate transparency," *ACM Queue*, vol. 12, no. 8, 10:10–10:19, 2014.

[17] A. Back, *Alternate proposal opt-in miner takes double-spend (re: replace-by-fee v0.10.0rc4)*, Message on Bitcoin development mailing list. [Online]. Available: https://www.mail-archive.com/bitcoin-development@lists.sourceforge.net/msg07122.html.

[18] P. Todd, *Re: alternate proposal opt-in miner takes double-spend (re: replace-by-fee v0.10.0rc4)*, Message on Bitcoin development mailing list. [Online]. Available: https://www.mail-archive.com/bitcoin-development@lists.sourceforge.net/msg07125.html.

CoinShuffle: Practical Decentralized Coin Mixing for Bitcoin

Tim Ruffing
CISPA, Saarland University

Pedro Moreno-Sanchez
CISPA, Saarland University

Aniket Kate
CISPA, Saarland University

***Abstract*—The decentralized currency network Bitcoin is emerging as a potential new way of performing financial transactions across the globe. Its use of pseudonyms towards protecting users' privacy has been an attractive feature to many of its adopters. Nevertheless, due to the inherent public nature of the Bitcoin transaction ledger, users' privacy is severely restricted to *linkable anonymity*, and a few transaction deanonymization attacks have been reported thus far.**

In this paper we propose CoinShuffle, a completely decentralized Bitcoin mixing protocol that allows users to utilize Bitcoin in a truly anonymous manner. CoinShuffle is inspired by the accountable anonymous group communication protocol Dissent and enjoys several advantages over its predecessor Bitcoin mixing protocols. It does not require any (trusted, accountable or untrusted) third party and it is perfectly compatible with the current Bitcoin system. CoinShuffle introduces only a small communication overhead for its users, while completely avoiding additional anonymization fees and minimalizing the computation and communication overhead for the rest of the Bitcoin system.

Bitcoin [3] is a fully decentralized digital crypto-currency network that does not require any central bank or monetary authority. Over the last few years we have observed an unprecedented and rather surprising growth of Bitcoin and its competitor currency networks. Many now believe that the concept of decentralized crypto-currencies is here to stay.

Nevertheless, these decentralized currency systems are far from perfect. Traditional payment systems rely on a trusted third party (such as a bank) to ensure that money cannot be spent twice. Decentralized currencies such as Bitcoin employ a global replicated append-only transaction log and proof-of-work (POW) instead to rule out double-spending. This requires managing a public ledger such that every transaction is considered valid only after it appears in the ledger.

However, given that the Bitcoin transactions of a user (in particular, of her pseudonyms, called *Bitcoin addresses*) are linkable, the public transaction ledger constitutes a significant privacy concern: Bitcoin's reliance on the use of pseudonyms to provide anonymity is severely restricted.

Several studies analyzing the privacy implications of Bitcoin indicate that Bitcoin's built-in privacy guarantees are not satisfactory. Barber et al. [4] observe that Bitcoin exposes its users to the possible linking of their Bitcoin addresses, which subsequently leads to a weak form of anonymity. Meiklejohn et al. [5] demonstrate how to employ a few basic heuristics to classify Bitcoin addresses that are likely to belong to the same user; this is further refined by Spagnuolo, Maggi, and Zanero [6]. Koshy, Koshy, and McDaniel [7] show that it is possible to identify ownership relationships between Bitcoin addresses and IP addresses.

Recently, some efforts have been made towards overcoming the above attacks and providing stronger privacy to the Bitcoin users by *mixing* multiple transactions to make input and output addresses of transactions unlinkable to each other. In this direction, some third-party mixing services [8], [9], [10] were first to emerge, but they have been prone to thefts [5]. Mixcoin [11] allows to hold these mixing services accountable in a reactive manner; however, the mixing services still remain single points of failure and typically require additional mixing fees. Zerocoin [12] and its successors [13], [14], [15] provide strong anonymity without any third party, but lack compatibility with the current Bitcoin system.

Maxwell proposes CoinJoin [16] to perform mixing in a manner that is perfectly compatible with Bitcoin, while ensuring that coins cannot be stolen. Naive centralized implementations of CoinJoin are actively used in practice [17] but suffer from a substantial drawback: The central mixing server still needs to be trusted to ensure anonymity, because it learns the relation between input and output addresses. While the use of blind signatures has been proposed to mitigate this problem [18], the security of the resulting protocols still relies critically on the existence of a reliable anonymous communication channel.

To avoid trusting the central server, it is desirable to implement CoinJoin in a decentralized manner. However, all secure decentralized protocols that have been proposed thus far use generic secure multi-party computation or multi-party sorting as a building block [19], [20] and consequently lack efficiency.

As a result, defining a practical and fully secure mixing scheme is considered an open problem by the Bitcoin community [21], [22], [23].

A. Contribution

We present CoinShuffle, a completely decentralized protocol that allows users to mix their coins with those of other interested users. CoinShuffle is inspired by CoinJoin [16] to ensure security against theft and by the accountable anonymous group communication protocol Dissent [24] to ensure anonymity as well as robustness against DoS attacks. The key idea is similar to decryption mix networks, and the protocol requires only standard cryptographic primitives such as signatures and

The full version corresponding to this extended abstract appeared at ESORICS 2014 [1]. The most recent technical report is available at [2].

public-key encryption. CoinShuffle is a practical solution for the Bitcoin mixing problem and its distinguishing features are as follows:

a) No Third Party: CoinShuffle preserves Bitcoin's decentralized trust ideology: it is executed exclusively by the Bitcoin users interested in unlinkability for their Bitcoin transactions, and it does not require any trusted, accountable, or untrusted third party. The unlinkability of transactions is protected as long as at least any two participants in a run of the protocol are honest.

b) Compatibility: CoinShuffle is fully compatible with the existing Bitcoin network. Unlike other decentralized solutions, it works immediately on top of the Bitcoin network without requiring any change to the Bitcoin rules or scripts.

c) No Mixing Fee: In absence of a third party that acts as a service provider, CoinShuffle does not charge its users any additional mixing fees. It also performs well in terms of Bitcoin transaction fees, because the participants are only charged the fee for a single mixing transaction.

d) Small Overhead: Our performance analysis demonstrates that CoinShuffle introduces only a small communication overhead for a participant (less than a minute for an execution with 20 participants), while the computation overhead remains close to negligible. Finally, CoinShuffle introduces only minimal additional overhead for the rest of the Bitcoin network.

We have developed a proof-of-concept implementation [2] of CoinShuffle leveraging an existing implementation of the Dissent protocol. We tested our implementation in Emulab [25], a testbed for distributed systems, in which network parameters such as topology or bandwidth of links can be easily configured. In this setting, we have run several experiments under controlled network conditions.

In the an setting with 20 Mbit/s and 100 msec latency, 50 participants need approximately 30 seconds to run CoinShuffle. The computation overhead constitutes only a small fraction of the overall time. In the case of 50 participants, the average computation time is slightly larger than 3 seconds, which constitutes approximately less than 1% of the time.

REFERENCES

[1] T. Ruffing, P. Moreno-Sanchez, and A. Kate, "CoinShuffle: Practical decentralized coin mixing for Bitcoin," in *Proc. of the 19th European Symposium on Research in Computer Security (ESORICS'14)*. Springer, 2014, pp. 345–364.

[2] "Project page of CoinShuffle providing the prototype implementation of coinshuffle and the most recent revision of this paper." [Online]. Available: http://crypsys.mmci.uni-saarland.de/projects/CoinShuffle

[3] S. Nakamoto, "Bitcoin: A peer-to-peer electronic cash system," Technical report, 2008. [Online]. Available: https://bitcoin.org/bitcoin.pdf

[4] S. Barber, X. Boyen, E. Shi, and E. Uzun, "Bitter to better — how to make Bitcoin a better currency," in *Proc. of the 15th Conference on Financial Cryptography and Data Security*, ser. FC'12. Springer, 2012, pp. 399–414. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-32946-3_29

[5] S. Meiklejohn, M. Pomarole, G. Jordan, K. Levchenko, D. McCoy, G. M. Voelker, and S. Savage, "A fistful of bitcoins: Characterizing payments among men with no names," in *Proc. of the 2013 Conference on Internet Measurement Conference*, ser. IMC'13. ACM, 2013, pp. 127–140. [Online]. Available: http://doi.acm.org/10.1145/2504730.2504747

[6] M. Spagnuolo, F. Maggi, and S. Zanero, "BitIodine: Extracting intelligence from the Bitcoin network," in *Proc. of the 17th Conference on Financial Cryptography and Data Security*, ser. FC'14. Springer, 2014. [Online]. Available: http://www.bitcoinsecurity.org/wp-content/uploads/2014/01/Koshy-FC141.pdf

[7] P. Koshy, D. Koshy, and P. McDaniel, "An analysis of anonymity in Bitcoin using P2P network traffic," in *Proc. of the 17th Conference on Financial Cryptography and Data Security*, ser. FC'14. Springer, 2014. [Online]. Available: http://www.bitcoinsecurity.org/wp-content/uploads/2014/01/Koshy-FC141.pdf

[8] "Bitcoin Fog." [Online]. Available: http://www.bitcoinfog.com

[9] "BitLaundry." [Online]. Available: http://app.bitlaundry.com

[10] "BitLaunder." [Online]. Available: https://bitlaunder.com

[11] J. Bonneau, A. Narayanan, A. Miller, J. Clark, J. A. Kroll, and E. W. Felten, "Mixcoin: Anonymity for Bitcoin with accountable mixes," in *Proc. of the 17th International Conference on Financial Cryptography and Data Security*, ser. FC'14. Springer, 2014. [Online]. Available: https://eprint.iacr.org/2014/077.pdf

[12] I. Miers, C. Garman, M. Green, and A. D. Rubin, "Zerocoin: Anonymous distributed e-cash from Bitcoin," in *Proc. of the 34th Symposium on Security and Privacy*, ser. S&P'13. IEEE, 2013, pp. 397–411. [Online]. Available: http://dx.doi.org/10.1109/SP.2013.34

[13] G. Danezis, C. Fournet, M. Kohlweiss, and B. Parno, "Pinocchio Coin: Building Zerocoin from a succinct pairing-based proof system," in *Proc. of the 1st ACM Workshop on Language Support for Privacy-enhancing Technologies*, ser. PETShop'13. ACM, 2013, pp. 27–30. [Online]. Available: http://doi.acm.org/10.1145/2517872.2517878

[14] C. Garman, M. Green, I. Miers, and A. D. Rubin, "Rational Zero: Economic security for Zerocoin with everlasting anonymity," 1st Workshop on Bitcoin Research, 2014. [Online]. Available: https://fc14.ifca.ai/bitcoin/papers/bitcoin14_submission_12.pdf

[15] E. Ben-Sasson, A. Chiesa, C. Garman, M. Green, I. Miers, E. Tromer, and M. Virza, "Zerocash: Decentralized anonymous payments from Bitcoin," in *Proc. of the 35th Symposium on Security and Privacy*, ser. S&P'14. IEEE, 2014.

[16] G. Maxwell, "CoinJoin: Bitcoin privacy for the real world," Post on Bitcoin Forum, Aug. 2013. [Online]. Available: https://bitcointalk.org/index.php?topic=279249

[17] Qkos Services Ltd, "Shared Coin." [Online]. Available: https://sharedcoin.com

[18] G. Maxwell, Post on Bitcoin Forum, 2013. [Online]. Available: https://bitcointalk.org/index.php?topic=279249.msg2984051#msg2984051

[19] E. Z. Yang, "Secure multiparty Bitcoin anonymization," Blog posting, Jul. 2012. [Online]. Available: http://blog.ezyang.com/2012/07/secure-multiparty-bitcoin-anonymization/

[20] Murphant (pseudonym), Post on Bitcoin Forum, Aug. 2013. [Online]. Available: https://bitcointalk.org/index.php?topic=279249.msg3013913#msg3013913

[21] M. Rosenfeld, "Using mixing transactions to improve anonymity," Post on Bitcoin Forum, Dec. 2011. [Online]. Available: https://bitcointalk.org/index.php?topic=54266

[22] Natanael L., Post on Bitcoin Forum, Aug. 2013. [Online]. Available: https://bitcointalk.org/index.php?topic=279249.msg3057216#msg3057216

[23] G. Maxwell, Post on Bitcoin Forum, Sep. 2013. [Online]. Available: https://bitcointalk.org/index.php?topic=279249.msg3013970#msg3013970

[24] H. Corrigan-Gibbs and B. Ford, "Dissent: Accountable anonymous group messaging," in *Proc. of the 17th Conference on Computer and Communications Security*, ser. CCS'10. ACM, 2010, pp. 340–350. [Online]. Available: http://doi.acm.org/10.1145/1866307.1866346

[25] B. White, J. Lepreau, L. Stoller, R. Ricci, S. Guruprasad, M. Newbold, M. Hibler, C. Barb, and A. Joglekar, "An integrated experimental environment for distributed systems and networks," in *OSDI'02*. USENIX Assoc., Dec. 2002, pp. 255–270.

Privacy Preserving Payments in Credit Networks*

Enabling trust with privacy in online marketplaces

Pedro Moreno-Sanchez
CISPA, Saarland University
pedro@mmci.uni-saarland.de

Aniket Kate
CISPA, Saarland University
aniket@mmci.uni-saarland.de

Matteo Maffei
CISPA, Saarland University
maffei@cs.uni-saarland.de

Kim Pecina
CISPA, Saarland University
pecina@cs.uni-saarland.de

***Abstract*—A credit network models trust between agents in a distributed environment and enables payments between arbitrary pairs of agents. With their flexible design and robustness against intrusion, credit networks form the basis of several Sybil-tolerant social networks, spam-resistant communication protocols, and payment systems. Existing systems, however, expose agents' trust links as well as the existence and volumes of payment transactions, which is considered sensitive information in social environments or in the financial world. This raises a challenging privacy concern, which has largely been ignored by the research on credit networks so far.**

This paper presents PrivPay, the first provably secure privacy-preserving payment protocol for credit networks. The distinguishing feature of PrivPay is the obliviousness of transactions, which entails strong privacy guarantees for payments. PrivPay does not require any trusted third party, maintains a high accuracy of the transactions, and provides an economical solution to network service providers. It is also general-purpose trusted hardware-based solution applicable to all credit network-based systems. We implemented PrivPay and demonstrated its practicality by privately emulating transactions performed in the Ripple payment system over a period of four months.

I. Introduction

Credit networks [2]–[4] exemplify a flexible yet robust design for distributed trust through pairwise credit allocations, indicating commitments to possible payments. In credit networks, agents (or users) express trust in each other numerically in terms of the credit they are willing to extend each other. By introducing suitable definitions of payments, credit networks may support a variety of applications [5]–[10].

Indeed, several systems based on the concept of credit networks have been proposed in the last few years, such as Bazaar [5], Iolaus [7], Ostra [6], Ripple [9], and SocialCloud [8]. Among these, the Google-backed [11] payment system Ripple [9] is emerging as an economical, fast method for performing financial transactions online. Ripple may serve as a complement to decentralized currency systems like Bitcoin [12], and a few banks have started to use Ripple in online payment systems [13], [14].

Despite its promising future, the concept of credit networks is still in an early stage and there is room for improvement. System issues such as liquidity [15], network formation [16], [17] and routing scalability [5], [18] of credit networks have been addressed in the recent literature; however, the important issue of credit networks' privacy has not been thoroughly investigated yet. While employing credit network-based payment systems, businesses and customers strive to ensure the privacy of their credit links and transactions from the prying eyes of competitors, authorities, and even service providers; patients want to protect the privacy of their medical bills; in Sybil-tolerant social networks based on credit networks [18], users naturally demand to keep some of their social links and interactions hidden from others. In general, privacy of credit links and payments is crucial for credit network based systems.

* The full version of this paper has been accepted at the Network Distributed Systems Security Symposium (NDSS 2015), and it is available at the project webpage [1].

II. Challenges

Designing a privacy-preserving solution for credit networks is technically challenging. Simple anonymization methods such as the pseudonyms employed in Ripple [19] are ineffective, as all transactions remain linkable to each other and they are susceptible to deanonymization attacks. For instance, Minkus et al. [20] were recently able to successfully identify and reveal highly sensitive information about eBay users by accessing their public profiles in the eBay Feedback System and correlating these with social network profiles on Facebook. In decentralized solutions where only the system users are entrusted with their credit links, the system's availability and efficiency is significantly hampered, as users are not online all the time and service providers cannot perform any transaction without the users. Providing the service provider only with the topological network graph while keeping credit values private still leads to a privacy loss. Besides revealing the transaction partners' pseudonyms, a public topological network graph also opens the system up to correlation attacks that ultimately reveal the partners' real identities [21], [22]. Perturbing the links or their credit values by means of differential privacy techniques [23], [24] would yield stronger privacy guarantees, but this is often unacceptable in payment scenarios as it implies unconsented redistribution of credit. Finally, precomputing the transitive closure of the network and then accessing it through a data-oblivious protocol is infeasible as the credit network is highly dynamic (e.g., credit links typically get modified with every transaction).

III. Our Contribution

We present PrivPay, a novel architecture for credit networks that preserves the privacy of transactions, maintains a high rate of transaction accuracy, and provides high performance. The distinguishing feature of our architecture is a novel

data-oblivious algorithm for computing the maximal credit between two agents, without revealing any information about the credit network, the transaction, or the agents themselves. This algorithm is implemented by employing a minimal secure and verifiable server-side execution environment, such as the IBM 4765 cryptographic co-processor [25]. PrivPay does not introduce significant computational or financial overhead to either the credit network service provider or the users. In particular, we avoid computationally burdensome cryptography at the user ends, which paves the way for deploying PrivPay on mobile devices such as smartphones.

We formalized for the first time the privacy properties of interest for credit networks and prove that PrivPay achieves them. In particular, we demonstrate that no third party, including the service provider, can identify the transaction values or the parties performing the transaction. Furthermore, the network is concealed from both the users and the server.

We thereby characterize two fundamental privacy properties for transactions in a credit network, namely, value privacy and receiver privacy. Intuitively, we say that a credit network maintains *value privacy* if the adversary cannot determine the value of a transaction between two non-compromised users. We say that a credit network maintains *receiver privacy* if the adversary cannot determine the receiver of a transaction, as long as this is issued by a non-compromised sender. We formalize these two privacy definitions as cryptographic games.

Notice that, for our definitions, we assume that transactions are executed by the senders and thus we define receiver privacy. It is, however, easily possible to define the complementary sender privacy property if in some credit network setting transactions are executed by the receiver.

We have implemented our system in multithreaded C++ code. For our experiments, we have extracted payment transactions from the real-world Ripple payment ledgers from October 2013 until January 2014, conveying a dataset with more than 14,000 users and more than 8,000 transactions. Our experiments show that a payment operation on average can be done in a privacy-preserving manner within 1.5 seconds, while adding a link between two users into the network on average requires only 0.1 seconds. The execution of our data-oblivious algorithm within the universe creator module takes approximately 22 seconds. Several instances of the algorithm, however, can be run in parallel as a background process to enlarge the set of paths used for checking the maximal credit between the transacting agents. Therefore, PrivPay is suitable to deploy as an online real-time payment system.

REFERENCES

[1] "The PrivPay Project Webpage," http://crypsys.mmci.uni-saarland.de/projects/PrivPay/.

[2] D. DeFigueiredo and E. T. Barr, "TrustDavis: A Non-Exploitable Online Reputation System," in *7th IEEE International Conference on E-Commerce Technology*, 2005, pp. 274–283.

[3] A. Ghosh, M. Mahdian, D. M. Reeves, D. M. Pennock, and R. Fugger, "Mechanism Design on Trust Networks," in *WINE'07*, 2007, pp. 257–268.

[4] D. Karlan, M. Mobius, T. Rosenblat, and A. Szeidl, "Trust and Social Collateral," *The Quarterly Journal of Economics*, vol. 124, no. 3, pp. 1307–1361, 2009.

[5] A. Post, V. Shah, and A. Mislove, "Bazaar: Strengthening User Reputations in Online Marketplaces," in *NSDI'11*, 2011, pp. 14–14.

[6] A. Mislove, A. Post, P. Druschel, and K. P. Gummadi, "Ostra: Leveraging Trust to Thwart Unwanted Communication," in *NSDI'08*, 2008, pp. 15–30.

[7] A. M. Kakhki, C. Kliman-Silver, and A. Mislove, "Iolaus: securing online content rating systems," in *WWW*, 2013, pp. 919–930.

[8] A. Mohaisen, H. Tran, A. Chandra, and Y. Kim, "Trustworthy distributed computing on social networks," in *ASIACCS*, 2013, pp. 155–160.

[9] "Ripple," https://ripple.com/.

[10] A. Mohaisen, N. Hopper, and Y. Kim, "Keep your friends close: Incorporating trust into social network-based Sybil defenses," in *INFOCOM, 2011 Proceedings IEEE*, 2011, pp. 1943–1951.

[11] "IDG Capital Partners and Google Ventures Invest in Ripple Developer OpenCoin," http://online.wsj.com/article/PR-CO-20130514-908271.html.

[12] "Ripple for Bitcoin Exchanges," https://ripple.com/wiki/Introduction_to_Ripple_for_Bitcoiners.

[13] P. Rizzo, "Fidor Becomes First Bank to Use Ripple Payment Protocol," http://www.coindesk.com/fidor-becomes-first-bank-to-use-ripple-payment-protocol/, 2014.

[14] A. Liu, "Ripple Labs Signs First Two US Banks," https://ripple.com/ripple-labs-signs-first-two-us-banks/, 2014.

[15] P. Dandekar, A. Goel, R. Govindan, and I. Post, "Liquidity in credit networks: a little trust goes a long way." in *ACM Conference on Electronic Commerce*, 2011, pp. 147–156.

[16] P. Dandekar, A. Goel, M. P. Wellman, and B. Wiedenbeck, "Strategic Formation of Credit Networks," in *WWW '12*, 2012, pp. 559–568.

[17] M. P. Wellman and B. Wiedenbeck, "An empirical game-theoretic analysis of credit network formation," in *Allerton Conference*, 2012, pp. 386–393.

[18] B. Viswanath, M. Mondal, K. P. Gummadi, A. Mislove, and A. Post, "Canal: Scaling Social Network-based Sybil Tolerance Schemes," in *EuroSys '12*, 2012, pp. 309–322.

[19] "Ripple FAQ," https://ripple.com/wiki/FAQ.

[20] T. Minkus and K. W. Ross, "I Know What You're Buying: Privacy Breaches on eBay," in *PETS'14*, 2014.

[21] A. Narayanan and V. Shmatikov, "Robust De-anonymization of Large Sparse Datasets," in *IEEE S&P (Oakland'08)*, 2008, pp. 111–125.

[22] K. Sharad and G. Danezis, "An Automated Social Graph De-anonymization Technique," *CoRR*, vol. abs/1408.1276, 2014.

[23] A. Sala, X. Zhao, C. Wilson, H. Zheng, and B. Y. Zhao, "Sharing Graphs Using Differentially Private Graph Models," in *IMC '11*, 2011, pp. 81–98.

[24] P. Mittal, C. Papamanthou, and D. X. Song, "Preserving Link Privacy in Social Network Based Systems," in *NDSS*, 2013.

[25] IBM Systems, "IBM Systems cryptographic hardware products," http://www-03.ibm.com/security/cryptocards/.

Mining Apps for Abnormal Usage of Sensitive Data

Vitalii Avdiienko · Konstantin Kuznetsov ·
Alessandra Gorla · Andreas Zeller
Software Engineering Chair
Saarland University
Saarbrücken, Germany
Email: ⟨lastname⟩@cs.uni-saarland.de

Steven Arzt[1] · Siegfried Rasthofer[1] · Eric Bodden[1,2]
Secure Software Engineering Group
[1]TU Darmstadt
[2]Fraunhofer SIT
Darmstadt, Germany
Email: ⟨firstname.lastname⟩@cased.de

***Abstract*—What is it that makes an app malicious? One important factor is that malicious apps *treat sensitive data differently from benign apps.* To capture such differences, we mined the top 2,866 benign Android applications for their data flow from sensitive sources, and compare these flows against those found in malicious apps. We find that (a) for every sensitive source, the data ends up in a small number of typical sinks; (b) these sinks differ considerably between benign and malicious apps; (c) these differences can be used to flag malicious apps due to their abnormal data flow; and (d) malicious apps can be identified by their abnormal data flow alone, without requiring known malware samples. In our evaluation, our MUDFLOW prototype correctly identified 73.7% of all novel malware, and 86.4% of novel malware leaking sensitive data.**

I. Extended Abstract

Most existing malware detectors work *retrospectively,* checking an unknown app against features and patterns known to be malicious. Such patterns can either be given explicitly ("Text messages must only be sent after user's consent"), or induced implicitly from samples of known malware ("This app contains code known to be part of the TDSS trojan."). If a novel app is sufficiently different from known malware, though, this retrospective detection can fail.

In this work, we thus conversely investigate the idea that, given access to a sufficiently large set of "benign" apps, one might be able to detect novel malware not by its similarity with respect to existing malware, but rather through its *dissimilarity with respect to those benign applications.* Checking for dissimilarity is different from checking for similarity, though, because in terms of functionality or code fragments, we already have lots of dissimilarity across benign applications themselves. As a measure for establishing similarity or dissimilarity with respect to the norm, we thus explore the *usage of sensitive data* in an app. Specifically, we apply *static taint analysis* on the 2,866 most popular Android apps from the Google Play Store to determine, for every sensitive data source, the sensitive APIs to which this data flows. We consider these flows to constitute the "normal" usage of sensitive data; as we assume the most popular Google Play Store apps to be benign, these flows also resemble "benign" usage.

As an example of such flows, consider the well known Android *Twitter* app. Table I shows its extracted data flows. We can see that, while the Twitter app accesses sensitive account information, it uses this information only to manage synchronization across multiple devices. Network information is being accessed (as part of the main functionality of the app), saved in logs, and passed on to other components.

TABLE I
Flows in Android Twitter App

AccountManager.get()	⇝ *ContentResolver.setSyncAutomatically()*
AccountManager.get()	⇝ *AccountManager.addOnAccountsUpdatedListener()*
AccountManager.get()	⇝ *Activity.setResult()*
AccountManager.get()	⇝ *Log.w()*
AccountManager.getAccountsByType()	⇝ *ContentResolver.setSyncAutomatically()*
AccountManager.getAccountsByType()	⇝ *Activity.setResult()*
AccountManager.getAccountsByType()	⇝ *Log.w()*
Uri.getQueryParameter()	⇝ *Activity.startActivity()*
Uri.getQueryParameter()	⇝ *Activity.setResult()*
Uri.getQueryParameter()	⇝ *Activity.startActivityForResult()*
Uri.getQueryParameter()	⇝ *Log.d()*
Uri.getQueryParameter()	⇝ *Log.v()*
Uri.getQueryParameter()	⇝ *Log.w()*
SQLiteDatabase.query()	⇝ *Log.d()*
SQLiteOpenHelper.getReadableDatabase()	⇝ *Log.d()*
SQLiteOpenHelper.getWritableDatabase()	⇝ *Log.d()*

TABLE II
Flows in com.keji.danti604 malware

TelephonyManager.getSubscriberId()	⇝ *URL.openConnection()*
TelephonyManager.getDeviceId()	⇝ *URL.openConnection()*

In contrast, consider the *com.keji.danti604* malware from the VirusShare database. Table II shows the two flows in that application; both leak the subscriber and device ID to a Web server. Both these flows are very uncommon for benign applications; furthermore, *danti604* does not contain any of the flows that would normally come with apps that use the *TelephonyManager* for legitimate reasons. Thus, *danti604* is an anomaly—not only because it may be similar to known malware, but in particular because its data flows are *dissimilar* to flows found in benignware such as Twitter.

We have built a tool called MUDFLOW[1] which determines such flows for all sensitive Android sources, building on our earlier work on massive app mining [2]. Internally, MUDFLOW uses the static taint tracking tool FLOWDROID [1] to identify data flows in Android applications. We chose FLOWDROID because of its precise analysis and because it accurately models the lifecycle of Android applications and their interactions with the operating system.

Listing 1 shows an example of how an Android application can obtain leaked data. The example reads the phone's unique

[1]MUDFLOW = Mining Unusual Data Flow

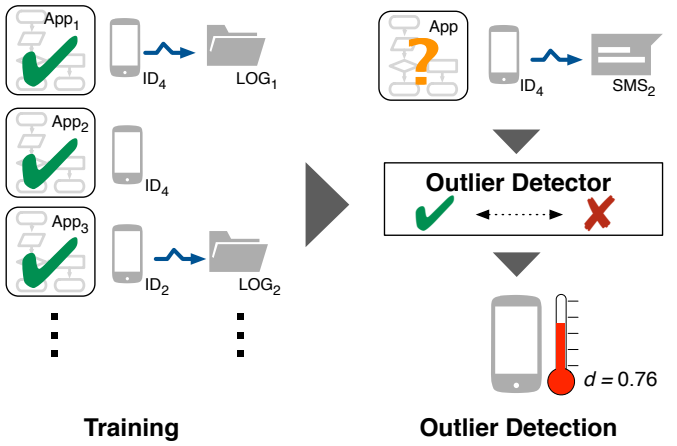


Fig. 1. Per-category outlier detection. For each SUSI category such as UNIQUE_ID (shortened to "ID"), MUDFLOW identifies the apps that use APIs of that category as source, and trains a classifier from the originating flows. The classifier then takes a new unknown app, and determines its distance from the "normal" flows trained with. The higher the distance, the less "normal" its features are.

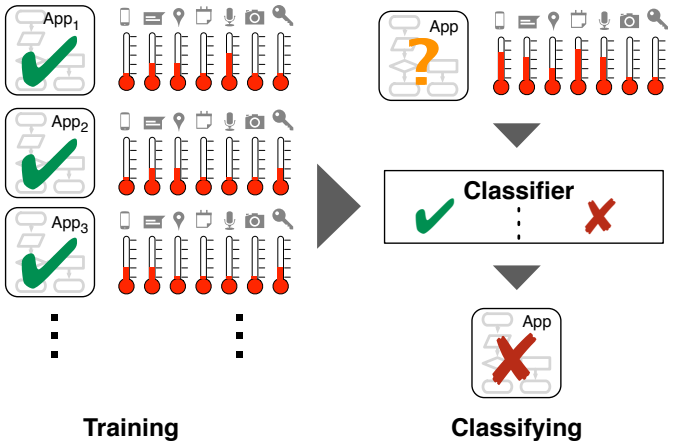


Fig. 2. Classifying apps across multiple categories. For each "benign" app in the Google Play store, we determine its vector of probabilities of being an outlier in each SUSI category. A one-way classifier trained from these vectors can label an unknown app as "likely benign" if it is normal across all categories, or "likely malicious" instead.

identifier and sends it to the example telephone number "+1 234" using an SMS message. In real-world applications, the path between source (the call to *getDeviceId()* and sink (the call to *sendTextMessage()* can be substantially longer, and may include field and array accesses, polymorphic (library) method calls, conditionals, etc. FLOWDROID follows this path, modeling that "*a* is tainted" at Line 5 due to normal forward propagation, and finds that the information flows into the SMS at Line 8.

MUDFLOW extracts such flows between sources and sinks from applications, and then implements multiple *classifiers,* trained on the data flow of benign apps, to automatically flag apps with suspicious features. Specifically, for each SUSI category of sensitive APIs, we extract its sensitive flows (Figure 1) from multiple apps, obtaining an *outlier score* for an app in this category: The higher the score, the less "normal" the app's flows in that category are.

Aggregating these scores across all API categories produces a vector of outlier scores (dubbed a "maliciogram"). By training a one-class SVM on these vectors, we obtain a classifier (Figure 2) that determines for an app whether its information flows are in line with the norm or whether it differs; in the latter case, it is likely to be malware.

```
1  void onCreate() {
2      TelephonyManager mgr = (TelephonyManager)
3          this.getSystemService(TELEPHONY_SERVICE);
4      String devId = mgr.getDeviceId();
5      String a = devId;
6      String str = prefix(a);
7      SmsManager sms = SmsManager.getDefault();
8      sms.sendTextMessage("+1 234", null, str, null, null);
9  }
10 String prefix(String s) {
11     return "DeviceId: " + s;
12 }
```

Listing 1. Android Data Leak Example

To the best of our knowledge, MUDFLOW is the first approach to massively mine application collections for patterns of "normal" data flow, and to use these mined patterns to detect malicious behavior.

By applying MUDFLOW on the 2,866 most popular apps collected from the Google Play Store, we have extracted *typical usage of sensitive sources* across these apps. Furthermore, we can contrast it against the usage found in common collections of malicious apps. Our experiments show that dissimilarity with benign apps, determined through data flow from sensitive sources, can be a significant factor in characterizing malware. In our experiment on a set of 10,552 malicious apps leaking sensitive data, MUDFLOW recognized 86.4% of the malware as such, with a false positive rate of 11.7%, which is remarkable given that MUDFLOW is not trained on malware samples.

Our full paper has been accepted at the *International Conference on Software Engineering* (ICSE) 2015 and been nominated as best paper (result pending). To access the full paper as well as all our mined data as well as the scripts for our statistical analysis, see our project page

http://www.st.cs.uni-saarland.de/appmining/

REFERENCES

[1] S. Arzt, S. Rasthofer, C. Fritz, E. Bodden, A. Bartel, J. Klein, Y. Le Traon, D. Octeau, and P. McDaniel. FlowDroid: Precise context, flow, field, object-sensitive and lifecycle-aware taint analysis for Android apps. In *Proceedings of the 35th ACM SIGPLAN Conference on Programming Language Design and Implementation*, PLDI '14, pages 259–269, New York, NY, USA, 2014. ACM.

[2] A. Gorla, I. Tavecchia, F. Gross, and A. Zeller. Checking app behavior against app descriptions. In *Proceedings of the 36th International Conference on Software Engineering*, ICSE 2014, pages 1025–1035, New York, NY, USA, June 2014. ACM.

Using An Instrumentation based Approach to Detect Inter-Component Leaks in Android Apps

Li Li, Tegawendé F. Bissyandé, Jacques Klein, Yves Le Traon
SnT, University of Luxembourg, Luxembourg
{li.li, tegawende.bissyande, jacques.klein, yves.letraon}@uni.lu

I. Introduction

The success of the Android OS in its user base as well as in its developer base can partly be attributed to its communication model, named Inter-Component Communication (ICC), which promotes the development of loosely-coupled applications. By dividing applications into components that can exchange data within a single application and across several applications, Android encourages software reuse, and thus reduces developer burden.

Unfortunately, the ICC model, which provides a message passing mechanism for data exchange among components, can be misused by malicious apps to threaten user privacy. Indeed, researchers have shown that Android apps frequently send users private data outside the device without their prior consent. Those applications are said to leak private data [2]. However, there is still a lack of a comprehensive study on the characteristics of the usage of ICCs by Android malware. Typically, what is the extent of the presence of privacy leaks in Android malware?

To answer such a question, an Android analysis tool has to be developed for tracking privacy leaks. Although, most of the privacy leaks are simple, i.e., easily identifiable as they operate within a single component. Thus, analyzing components separately is not enough to detect leaks: it is necessary to perform an inter-component analysis of applications. Android app analysts could leverage such a tool to identify malicious apps that leak private data. For the tool to be useful, it has to be highly precise and minimize the false positive rate when reporting applications leaking private data.

Thus, we propose IccTA[1] , an Inter-component communication Taint Analysis tool, for a sound and precise detection of ICC links and leaks. Although our approach is generic and can be used for any data-flow analysis, we focus in this paper on using IccTA to detect ICC-based privacy leaks. we test IccTA on 15,000 real-world apps randomly selected from Google Play market in which we detect 337 apps with 2,395 ICC leaks. We also launch IccTA on the MalGenome [5] set containing 1260 malware, where IccTA reports 108 apps with 534 ICC leaks. By comparing the detecting rate $r = \frac{\#\ of\ detected\ apps}{\#\ of\ tested\ apps}$ of the two data sets, we found that $r_{MalGenome} = 8.6\%$ is much higher than $r_{GooglePlay} = 2.2\%$. Thus, we can conclude that *malware are using ICC to leak private data more than benign apps*, making ICC a potential feature for malware detection. This paper is an extended abstract version of our research paper [3], where interested readers can find more details of this work.

[1] https://sites.google.com/site/icctawebpage/

II. ICC Problem

We define a privacy leak as a path from sensitive data, called *source*, to statements sending this data outside the application or device, called *sink*. A path may be within a single component or across multiple components.

```
1  //TelephonyManager telMnger; (default)
2  //SmsManager sms; (default)
3  class Activity1 extends Activity {
4   void onCreate(Bundle state) {
5    Button to2 = (Button) findViewById(to2a);
6    to2.setOnClickListener(new OnClickListener(){
7     void onClick(View v) {
8      String id = telMnger.getDeviceId();
9      Intent i = new
       Intent(Activity1.this,Activity2.class);
10     i.putExtra("sensitive", id);
11     Activity1.this.startActivity(i);
12 }});}}
13 class Activity2 extends Activity {
14  void onStart() {
15   Intent i = getIntent();
16   String s = i.getStringExtra("sensitive");
17   sms.sendTextMessage(number,null,s,null,null);
18 }}
```

Listing 1: A Running Example.

Listing 1 illustrates the concept of ICC leak through a concrete example. The code snippets present two Activities: $\texttt{Activity}_1$ and $\texttt{Activity}_2$. $\texttt{Activity}_1$ registers an anonymous button listener for the *to2* button (lines 5-11). An ICC method `startActivity` is used by this anonymous listener. When button *to2* is clicked, the `onClick` method is executed and the user interface will change to $\texttt{Activity}_2$. An `Intent` containing the device ID (lines 15), considered as sensitive data, is then exchanged between the two components by first attaching the data to the Intent with the `putExtra`

(A)
```
  // modifications of Activity1
-  Activity1.this.startActivity(i);
+  IpcSC.redirect0(i);
```

(B)
```
// creation of a helper class
+class IpcSC {
+ static void redirect0(Intent i) {
+  Activity2 a2 = new Activity2(i);
+  a2.dummyMain();
+ }
+}
```

(C)
```
// modifications in Activity2
+public Activity2(Intent i) {
+  this.intent_for_ipc = i;
+}
 public Intent getIntent() {
+ return this.intent_for_ipc;
 }
+public void dummyMain() {
+ // lifecycle and callbacks
+ // are called here
+}
```

Fig. 1: Handling `startActivity` ICC method.

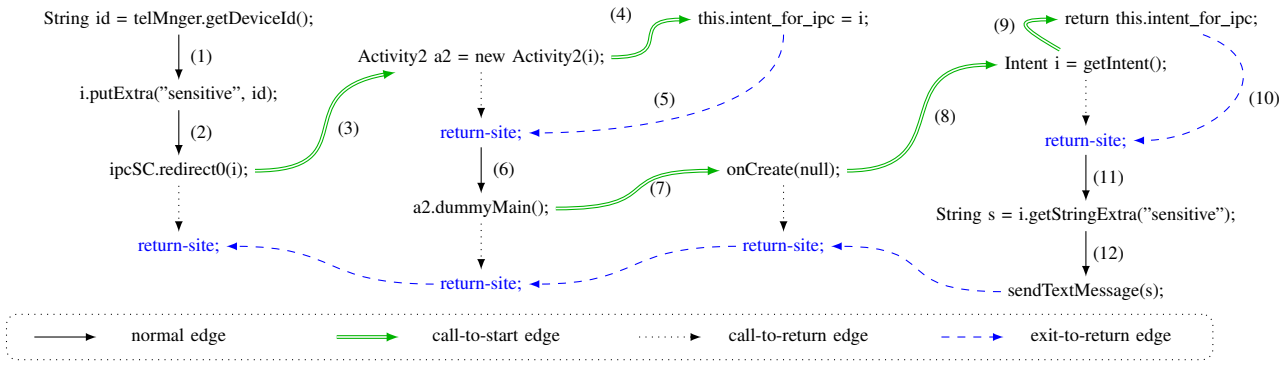


Fig. 2: The control-flow graph of the instrumented running example.

method (lines 10) and then by invoking the ICC method `startActivity` (lines 11). Note that the Intent is created by explicitly specifying the target class ($\texttt{Activity}_2$).

In this example, `sendTextMessage` is systematically executed when $\texttt{Activity}_2$ is loaded since `onStart` is in the execution lifecycle of an `Activity`. The data retrieved from the `Intent` is thus sent as a SMS message to the specified phone number: *there is an ICC leak triggered by button* to2. When *to2* is clicked, the device ID is transferred from $\texttt{Activity}_1$ to $\texttt{Activity}_2$ and then outside the application.

III. Instrumentation based Approach

In this section we briefly introduce our instrumentation based approach, IccTA, which first modifies an Android app's code representation to directly connect components (through ICC links [4]) and then uses a modified version of FlowDroid [1] to build a complete control-flow graph (CFG) of the whole application. This allows propagating the context (e.g., the value of Intents) between Android components and yielding a highly precise data-flow analysis.

Fig. 1 shows the code transformation done by IccTA for the ICC link between $\texttt{Activity}_1$ and $\texttt{Activity}_2$ of our running example. IccTA first creates a helper class named `IpcSC` (B in Fig. 1) which acts as a bridge connecting the source and destination components. Then, the `startActivity` ICC method is removed and replaced by a statement calling the generated helper method (`redirect0`) (A).

In (C), IccTA generates a constructor method taking an `Intent` as parameter, a `dummyMain` method to call all related methods of the component (i.e., lifecycle and callback methods) and overrides the `getIntent` method. An Intent is transferred by the Android system from the caller component to the callee component. We model the behavior of the Android system by explicitly transferring the Intent to the destination component using a customized constructor method, $\texttt{Activity}_2\texttt{(Intent i)}$, which takes an `Intent` as its parameter and stores the Intent to a newly generated field `intent_for_ipc`. The original `getIntent` method asks the Android system for the incoming Intent object. The new `getIntent` method models the Android system behavior by returning the Intent object given as parameter to the new constructor method.

The helper method `redirect0` constructs an object of type $\texttt{Activity}_2$ (the target component) and initializes the new object with the `Intent` given as parameter to the helper method. Then, it calls the `dummyMain` method of $\texttt{Activity}_2$.

To resolve the target component, i.e., to automatically infer what is the type that has to be used in the method `redirect0` (in our example, to infer $\texttt{Activity}_2$), IccTA uses the ICC links extracted by our extended Epicc [4] in which not only the explicit Intents but also the implicit Intents are resolved. Therefore, there is no difference for IccTA to handle explicit or implicit Intents based ICCs.

Fig. 2 represents the CFG of the instrumented running example presented in Listing 1. In the CFG, `getDeviceId` is a *source* method in the anonymous `OnClickListener` class (line 6) called by $\texttt{Activity}_1$. Method `sendTextMessage` is a *sink* in $\texttt{Activity}_2$. There is an intra-component tainted statement path from the *source* method to *sink* method (represented by edges 1 to 12). Fig. 2 also shows that IccTA builds a precise cross-component control-flow graph. Since we use an technique instrumenting the code to build the CFG, the context of a static analysis is kept between components. This enables IccTA to analyze data-flows between components and thereby enables IccTA to have a better precision than existing approaches.

References

[1] Steven Arzt, Siegfried Rasthofer, Christian Fritz, Eric Bodden, Alexandre Bartel, Jacques Klein, Yves Le Traon, Damien Octeau, and Patrick McDaniel. "FlowDroid: Precise Context, Flow, Field, Object-sensitive and Lifecycle-aware Taint Analysis for Android Apps". In: *PLDI*. 2014.

[2] Li Li, Alexandre Bartel, Jacques Klein, and Yves Le Traon. "Automatically Exploiting Potential Component Leaks in Android Applications". In: *IEEE TrustCom*. 2014.

[3] Li Li, Alexandre Bartel, Tegawendé F Bissyandé, Jacques Klein, Yves Le Traon, Steven Arzt, Rasthofer Siegfried, Eric Bodden, Damien Octeau, and Patrick Mcdaniel. "IccTA: Detecting Inter-Component Privacy Leaks in Android Apps". In: *ICSE*. 2015.

[4] Damien Octeau, Patrick McDaniel, Somesh Jha, Alexandre Bartel, Eric Bodden, Jacques Klein, and Yves Le Traon. "Effective inter-component communication mapping in android with epicc: An essential step towards holistic security analysis". In: *USENIX Security*. 2013.

[5] Yajin Zhou and Xuxian Jiang. "Dissecting android malware: Characterization and evolution". In: *IEEE Security and Privacy*. 2012.

Privacy and Access Control for Outsourced Personal Records

Matteo Maffei, Giulio Malavolta, Manuel Reinert, Dominique Schröder
Saarland University & CISPA
{maffei,malavolta,reinert,schroeder}@cs.uni-saarland.de

***Abstract*—Cloud storage has rapidly become a cornerstone of many IT infrastructures, constituting a seamless solution for the backup, synchronization, and sharing of large amounts of data. Putting user data in the direct control of cloud service providers, however, raises security and privacy concerns related to the integrity of outsourced data, the accidental or intentional leakage of sensitive information, the profiling of user activities and so on. Furthermore, even if the cloud provider is trusted, users having access to outsourced files might be malicious and misbehave. These concerns are particularly serious in sensitive applications like personal health records and credit score systems.**

To tackle this problem, we present GORAM, a cryptographic system that protects the secrecy and integrity of outsourced data with respect to both an untrusted server and malicious clients, guarantees the anonymity and unlinkability of accesses to such data, and allows the data owner to share outsourced data with other clients, selectively granting them read and write permissions. GORAM is the first system to achieve such a wide range of security and privacy properties for outsourced storage. In the process of designing an efficient construction, we developed two new, generally applicable cryptographic schemes, namely, batched zero-knowledge proofs of shuffle and an accountability technique based on chameleon signatures, which we consider of independent interest. We implemented GORAM in Amazon Elastic Compute Cloud (EC2) and ran a performance evaluation demonstrating the scalability and efficiency of our construction.

This paper has been accepted for presentation at IEEE S&P 2015 [1].

I. Introduction

Cloud storage has rapidly gained a central role in the digital society, serving as a building block of consumer-oriented applications (e.g, Dropbox, Microsoft SkyDrive, and Google Drive) as well as particularly sensitive IT infrastructures, such as personal record management systems. For instance, credit score systems rely on credit bureaus (e.g., Experian, Equifax, and TransUnion in US) collecting and storing information about the financial status of users, which is then made available upon request. As a further example, personal health records (PHRs) are more and more managed and accessed through web services (e.g., private products like Microsoft HealthVault and PatientsLikeMe in US and national services like ELGA in Austria), since this makes PHRs readily accessible in case of emergency even without the physical presence of the e-health card and eases their synchronization across different hospitals.

Despite its convenience and popularity, cloud storage poses a number of security and privacy issues. The first problem is related to the *secrecy* of user data, which are often sensitive (e.g., PHRs give a complete picture of the health status of citizens) and, thus, should be concealed from the server. A crucial point to stress is that preventing the server from reading user data (e.g., through encryption) is necessary but not sufficient to protect the privacy of user data. Indeed, as shown in the literature [2], [3], the capability to link consecutive accesses to the same file can be exploited by the server to learn sensitive information: for instance, it has been shown that the access patterns to a DNA sequence allow for determining the patient's disease. Hence the *obliviousness* of data accesses is another fundamental property for sensitive IT infrastructures: the server should not be able to tell whether two consecutive accesses concern the same data or not, nor to determine the nature of such accesses (read or write). Furthermore, the server has in principle the possibility to modify client's data, which can be harmful for several reasons: for instance, it could drop data to save storage space or modify data to influence the statistics about the dataset (e.g., in order to justify higher insurance fees or taxes). Therefore another property that should be guaranteed is the *integrity* of user data.

Finally, it is often necessary to share outsourced documents with other clients, yet in a controlled manner, i.e., selectively granting them read and write permissions: for instance, PHRs are selectively shared with the doctor before a medical treatment and a prescription is shared with the pharmacy in order to buy a medicine. *Data sharing* complicates the enforcement of secrecy and integrity properties, which have to be guaranteed not only against a malicious server but also against malicious clients. Notice that the simultaneous enforcement of these properties is particularly challenging, since some of them are in seeming contradiction. For instance, *access control* seems to be incompatible with the obliviousness property: if the server is not supposed to learn which file the client is accessing, how can he check that the client has the rights to do so?

A. Our Contributions

In this work, we present GORAM, a novel framework for privacy-preserving cloud-storage. Users can share outsourced data with other clients, selectively granting them read and write permissions, and verify the integrity of such data. These are hidden from the server and access patterns are oblivious. GORAM is the first system to achieve such a wide range of security and privacy properties for storage outsourcing. More specifically, the contributions of this work are the following:

- We formalize the problem statement by introducing the notion of Group Oblivious RAM (GORAM). GORAM extends the concept of Oblivious RAM [4] (ORAM) [1]

[1]ORAM is a technique originally devised to protect the access pattern of software on the local memory and then used to hide the data and the user's access pattern in storage outsourcing services.

by considering multiple, possibly malicious clients, with read and/or write access to outsourced data, as opposed to a single client. We propose a formal security model that covers a variety of security and privacy properties, such as data integrity, data secrecy, obliviousness of access patterns, and anonymity.

- We first introduce a cryptographic instantiation based on a novel combination of ORAM [5], predicate encryption [6], and zero-knowledge (ZK) proofs (of shuffle) [7], [8]. This construction is secure, but building on off-the-shelf cryptographic primitives is not practical. In particular, clients prove to the server that the operations performed on the database are correct through ZK proofs of shuffle, which are expensive when the entries to be shuffled are tuples of data, as opposed to single entries.
- As a first step towards a practical instantiation, we maintain the general design, but we replace the expensive ZK proofs of shuffle with a new proof technique called *batched* ZK proofs of shuffle. A batched ZK proof of shuffle significantly reduces the number of ZK proofs by "batching" several instances and verifying them together. Since this technique is generically applicable in any setting where one is interested to perform a zero-knowledge proof of shuffle over a list of entries, each of them consisting of a tuple of encrypted blocks, we believe that it is of independent interest. This second realization greatly outperforms the first solution and is suitable for databases with relatively small entries, accessed by a few users, but it does not scale to large entries and many users.
- To obtain a scalable solution, we explore some trade-offs between security and efficiency. First, we present a new accountability technique based on chameleon signatures. The idea is to let clients perform arbitrary operations on the database, letting them verify each other's operation a-posteriori and giving them the possibility to blame misbehaving parties. Secondly, we replace the relatively expensive predicate encryption, which enables sophisticated role-based and attribute-based access control policies, with the more efficient broadcast encryption, which suffices to enforce per-user read/write permissions, as required in the personal record management systems we consider. This approach leads to a very efficient solution that scales to large files and thousands of users, with a combined communication-computation overhead of only 7% (resp. 8%) with respect to state-of-the-art, single-client ORAM constructions for reading (resp. writing) on a 1GB storage with 1MB block size (for larger datasets or block sizes, the overhead is even lower).

We have implemented GORAM in Amazon Elastic Compute Cloud (EC2) and conducted a performance evaluation demonstrating the scalability and efficiency of our construction. Although GORAM is generically applicable, the large spectrum of security and privacy properties, as well as the efficiency and scalability of the system, make GORAM particularly suitable for the management of large amounts of sensitive data, such as personal records.

REFERENCES

[1] M. Maffei, G. Malavolta, M. Reinert, and D. Schröder, "Privacy and Access Control for Outsourced Personal Records," 2015, to appear in *Proc. IEEE Symposium on Security and Privacy (S&P'15)*, preliminary version online at sps.cs.uni-saarland.de/publications/oakland15.pdf.

[2] M. Islam, M. Kuzu, and M. Kantarcioglu, "Access Pattern Disclosure on Searchable Encryption: Ramification, Attack and Mitigation," in *Proc. Annual Network & Distributed System Security Symposium (NDSS'12)*. Internet Society, 2012.

[3] B. Pinkas and T. Reinman, "Oblivious RAM Revisited," in *Proc. Advances in Cryptology (CRYPTO'10)*, ser. LNCS. Springer Verlag, 2010, pp. 502–519.

[4] O. Goldreich and R. Ostrovsky, "Software Protection and Simulation on Oblivious RAMs," *Journal of the ACM*, vol. 43, no. 3, pp. 431–473, May 1996.

[5] E. Stefanov, M. van Dijk, E. Shi, C. Fletcher, L. Ren, X. Yu, and S. Devadas, "Path ORAM: An Extremely Simple Oblivious RAM Protocol," in *Proc. Conference on Computer and Communications Security (CCS'13)*. ACM, 2013.

[6] J. Katz, A. Sahai, and B. Waters, "Predicate Encryption Supporting Disjunctions, Polynomial Equations, and Inner Products," in *Proc. Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT'08)*. Springer Verlag, 2008, pp. 146–162.

[7] J. Groth and A. Sahai, "Efficient Noninteractive Proof Systems for Bilinear Groups," *SIAM Journal on Computing*, vol. 41, no. 5, pp. 1193–1232, 2012.

[8] S. Bayer and J. Groth, "Efficient Zero-Knowledge Argument for Correctness of a Shuffle," in *Proc. Conference on the Theory and Applications of Cryptographic Techniques (EUROCRYPT'12)*, ser. LNCS. Springer Verlag, 2012, pp. 263–280.

A logic of trust for reasoning about delegation and revocation

Marcos Cramer, Agustín Ambrossio

University of Luxembourg, Luxembourg

Abstract

In ownership-based access control frameworks with the possibility of delegating permissions and administrative rights, chains of delegated accesses will form. There are different ways to treat these delegation chains when revoking rights, which give rise to different revocation schemes. Hagström et al. [2] proposed a framework for classifying revocation schemes, in which the different revocation schemes are defined graph-theoretically. Our work is based on the observation that there are some problems with Hagström et al.'s definitions of the revocation schemes, which have led us to propose a refined framework with new graph-theoretic definitions of the revocation schemes. In order to study the merits of various definitions of revocation schemes, we propose to apply the axiomatic method originating in social choice theory to revocation schemes. For formulating a desirable property of revocation frameworks, we propose a logic, *Trust Delegation Logic* (*TDL*), with which one can formalize the different reasons an agent may have for performing a revocation. Our refined graph-theoretic definitions of the revocation schemes, unlike Hagström et al.'s original definitions, satisfy the desirable property that can be formulated using TDL.

1 Introduction

In ownership-based frameworks for access control, it is common to allow principals (users or processes) to grant both permissions and administrative rights to other principals in the system. Often it is desirable to grant a principal the right to further grant permissions and administrative rights to other principals. This may lead to delegation chains starting at a *source of authority* (the owner of a resource) and passing on certain permissions to other principals in the chain.

Furthermore, such frameworks commonly allow a principal to revoke a permission that she granted to another principal. Depending on the reasons for the revocation, different ways to treat the chain of principals whose permissions depended on the second principal's delegation rights can be desirable. For example, if one is revoking a permission given to an employee because he is moving to another position in the company, it makes sense to keep in place the permissions of principals who received their permissions from this employee; but if one is revoking a permission from a user who has abused his rights and is hence distrusted by the user who granted the permission, it makes sense to delete the permissions of principals who received their permission from this user. Any algorithm that determines which permissions to keep intact and which permissions to delete when revoking a permission is called a *revocation scheme*. Revocation schemes are usually defined in a graph-theoretical way on the graph that represents which authorizations between the principals are intact.

2 Hagström et al.'s framework

Hagström et al. [2] have presented a framework for classifying possible revocation schemes along three different dimensions:

- **Propagation:** The decision of a principal i to revoke an authorization granted to a principal j may either be intended to affect only the direct recipient j (*local* revocation), or to propagate and affect all the other users in turn authorized by j (*global* revocation).
- **Dominance.** This dimension deals with the case when a principal losing a permission in a revocation still has permissions from other grantors. If these other grantors' revocation rights are dependent on the revoker, the revoker can dominate over these grantors and revoke the permissions from them (*strong* revocation). The revoker can also choose to make a *weak* revocation, where permissions from other grantors to a principal losing a permission are kept.
- **Resilience.** This dimension distinguishes revocation by *deletion* of positive authorizations from revocation by issuing a *negative* authorization which just inactivates positive authorizations. In the first case another principal may grant a similar authorization to the one that had been revoked, so the effect of the revocation does not persist in time. In the second case a negative authorization will overrule any (new) positive permission given to the same principal.

3 The revised framework

We identify a number of problems with Hagström et al.'s framework and the definitions of the revocation schemes included in the framework, which have motivated us to propose a refined framework:

- The behaviour of the revocation schemes is dependent on the conflict resolution policy of the system, which is not integrated into the framework. In our refined framework, it is integrated by extending the dominance dimension to include three options, *weak*, *predecessor-takes-precedence* (*p-t-p*, which corresponds to Hagström et al.'s "strong" under a positive-takes-precedence conflict resolution policy) and *strong* (which corresponds to the behaviour of all revocations in Hagström et al.'s framework under a negative-takes-precedence policy).
- In Hagström et al.'s framework, delete revocations are supposed be non-resilient, which according to Hagström et al. means that "another user may issue the same permission that was just revoked, and the effect of the revocation disappears". We have discovered that this property fails to be satisfied in global deletion revocations as defined by Hagström et al. We avoid this problem by inactivating instead of deleting the forward chain in a delete revocation.
- Hagström et al. motivate the distinction between delete and negative revocations mainly through the notion of resilience. However, in weak revocations there can be no difference between a resilient and a non-resilient revocation. They motivate the usage of weak negatives by pointing out that they are useful for temporary revocations. Because of the changes proposed in the previous item, a delete can also be easily undone in our refined framework. Hence we do not need weak negative revocations.
- We have identified some undesirable properties of strong and p-t-p delete revocations. To avoid this problem, the strong and p-t-p revocations in our framework are always performed by issuing negative authorizations, but we distinguish between resilient and non-resilient negative authorizations in order to still have non-resilient revocations.
- Hagström et al. do not allow negative authorizations to be inactivated. The reason they give is that they "do not want a revocation to result in a subject having more permissions than before the revocation". However, the deletion of negative authorizations is allowed, even though it may have the same effect.

	$K_i^t \forall k T_j^t T_k \mathbb{D}$	$\neg B_i^t \forall k T_j^t T_k \mathbb{D} \wedge \neg B_i^t \forall t' \forall k \neg T_j^{t'} T_k \mathbb{D}$	$B_i^t \forall t' \forall k \neg T_j^{t'} T_k \mathbb{D} \wedge \neg K_i^t \forall t' \forall k \neg T_j^{t'} T_k \mathbb{D}$	$K_i^t \forall t' \forall k \neg T_j^{t'} T_k \mathbb{D}$
$K_i^t T_j \mathbb{D}$	delegate	X	X	X
$\neg B_i^t T_j \mathbb{D} \wedge \neg B_i^t \neg T_j \mathbb{D}$	WLD	WGD	X	X
$B_i^t \neg T_j \mathbb{D} \wedge \neg K_i^t \neg T_j \mathbb{D}$	PLN $\circ$ SLN	WGD $\circ$ PLN $\circ$ SLN	PGN $\circ$ SGN	X
$K_i^t \neg T_j \mathbb{D}$	PLR $\circ$ SLR	WGD $\circ$ PLR $\circ$ SLR	PGN $\circ$ PLR $\circ$ SGN $\circ$ SLR	PGR $\circ$ SGR

Table 1: The correspondence between the revocation framework and reasons for revocating formalized in TDL

The revised revocation framework that we have developed avoids all of these problems.

In order to avoid that our refined framework turns out to have undesirable properties like those we identified in Hagström et al.'s framework, we propose to formally study the merits and demerits of various definitions of revocation schemes using the axiomatic method originating in social choice theory. Which behaviour is desirable for a revocation scheme depends on the reasons for performing the revocation. So in order to formulate an axiom, i.e. a desirable property of revocation schemes, we propose a logic, *Trust Delegation Logic* (*TDL*), with which one can formalize the different reasons an agent may have for performing a revocation. We show that our modified graph-theoretic definitions of the revocation schemes, unlike Hagström et al.'s original definitions, satisfy the desirable property that can be formulated using TDL.

4 Trust Delegation Logic (TDL)

In order to motivate the design decisions of TDL, we have studied the reasons for revocation and the corresponding choice of a revocation scheme in various scenarios. Let us consider an example of such a scenario:

Scenario 1 *User C is leaving to join the rival company. When user A notices the situation, she preemptively blocks C's capabilities (but keeping the authorizations previously issued by C).*

In this scenario user A had trusted user C in the past, thus issuing him an authorization. Since C is leaving to the rival company, A now distrusts C to access files or to newly delegate access right to others, but since A never misused any rights, A still has trust in the delegation authorization previously issued by C. So A will perform a *P-t-p Local Resilient* revocation and – if possible – a *Strong Local Resilient* revocation, in order to remove the authorizations that had been granted to C and to forbid as many other principals as possible to grant new authorizations to C, at the same time preserving the effect of authorizations that C had previously delegated.

It is this kind of reasoning about revocations that we intend to formalize with the help of TDL.

One central idea of TDL is that A grants B the right to further delegate some right only if A trusts B to make correct judgments about who should be given that right. By expressing her trust in B to make correct judgments about something, A commits herselves to the truth of judgments that she has not made herself, namely the judgments that B has committed himself to. When A makes a judgment herself, we say that A has *explicit belief* in the judgment, whereas a judgment that A is committed to in the light of a principal trusted by A believing the statement is an *implicit belief* of A. Trust of principal A in principal B is modelled as A's belief in B's trustworthy. Depending on whether A's belief is explicit or implicit, we can also call this trust explicit or implicit. For example, if A expresses trust in B concerning the action of expressing trust in other principals, and B expresses trust in C, then A explicitly trusts B and implicitly trusts C.

A further central idea is that a principal A should have access right of access type α iff the SOA of that object trusts A, either explicitly or implicitly, concerning access α. Delegation chains correspond to chains of principals along whom an implicit trust in some principal can project upwards towards the SOA. A revocation takes place when at some point along such a chain of principals, a principal stops trusting in the next principal on the chain, thus disabling this upward projection of implicit trust.

TDL allows us to model different ways in which a principal can stop trusting or start distrusting another principal. The various revocation schemes correspond to these various ways of stopping to trust.

One important distinction in TDL is that between i's lack of trust in j, modelled as i's lack of belief in the trustworthiness of j ($\neg B_i T_j$) and i's distrust in j, modelled as i's belief in the non-trustworthiness of j ($B_i \neg T_j$). Furthermore, we take over the ditinction between *belief* ($B_i \varphi$) and *strong belief* ($K_i \varphi$) from [1]. Using this distinction, we can distinguish between five different levels of trust between a principal i and a principal j: *Strong trust*, where i strongly believes that j is trustworthy ($K_i^t T_j$), *weak trust* ($B_i^t T_j \wedge \neg K_i^t T_j$), *lack of trust* ($\neg B_i^t T_j \wedge \neg B_i^t \neg T_j$), *weak distrust* ($B_i^t \neg T_j \wedge \neg K_i^t \neg T_j$), and *strong distrust* ($K_i^t a \neg T_j$). The distinction between weak trust and lack of trust is not relevant for modelling the reasoning about delegation and revocation, but the distinction between the remaining four levels of trust is relevant.

5 Relation between TDL and revocation

For modelling the reasons for local revocation schemes (as in the scenario discussed above), we have to distinguish between the i's level of trust concerning j's right to delegate ($B_i T_j \mathbb{D}$) and i's level of trust in authorizations previously issued by j (which we can write in TDL as $B_i^t \forall k T_j^t T_k \mathbb{D}$). Table 1 shows which granting-revocation behaviour corresponds to each possible combinations of trust levels (in the table we use W, P, S, L, G, N, R and D as abbreviations for *weak*, *p-t-p (predecessor-takes-precedence)*, *strong*, *local*, *global*, *non-resilient*, *resilient* and *delete* respectively). Some cells contain multiple revocation schemes. This means that the granting-revocation behaviour corresponding to the combination of trust levels represented by that cell consists of performing multiple revocation schemes at the same time.

Given these explanations, we can now sketch how TDL allows us to formulate a desirable property for graph-theoretic definitions of revocation schemes: The graph-theoretic definitions of the revocation schemes should be such that for any given delegation and revocation interaction between the principals, an active authorization to a principal A should exist in a graph if and only if – translating the delegation and revocation behaviour to TDL – the SOA believes A to be trustworthy for the access in question.

This property is satisfied by the graph-theoretic definition of the revocation schemes in our refined framework, but not by the original definitions in Hagström et al.'s framework.

References

[1] R. Demolombe. Reasonig about trust: A formal logical framework. In C. Jensen, S. Poslad, and T. Dimitrakos, editors, *Trust Management*, volume 2995 of *Lecture Notes in Computer Science*, pages 291–303. 2004.

[2] Å. Hagström, S. Jajodia, F. Parisi-Presicce, and D. Wijesekera. Revocations-A Classification. In *Proceedings of the 14th IEEE Workshop on Computer Security Foundations*, CSFW '01, pages 44–, Washington, DC, USA, 2001. IEEE Computer Society.

Saving Re-Encryption Randomized Partial Checking Mix Nets for Risk-Avoiding Adversaries

Ralf Küsters
University of Trier, Germany
kuesters@uni-trier.de

Tomasz Truderung
University of Trier, Germany
truderung@uni-trier.de

Mix nets often play a central role in modern e-voting systems. In such systems, voters' ballots, which typically include the voters' choices in an encrypted form, are posted on a bulletin board. Then, the ballots are first put through a mix net, which consists of several mix servers, before they are decrypted, in order to hide the link between a voter's ballot and its (plaintext) choice, and hence, in order to guarantee privacy of votes. In the context of e-voting, besides privacy, it is also crucial that potential manipulations are detected. That is, if ballots have been dropped or manipulated by a mix server, this should be detected. This property is called verifiability.

Many schemes have been proposed in the literature to obtain verifiable mix nets (see, e.g., [13], [11], [4], [15], [5], [6]). Most of the schemes are quite complex and some have been broken (see, e.g., [14]).

In 2002, Jakobsson, Juels, and Rivest proposed a particularly simple and efficient construction [5], the so-called re-encryption randomized partial checking (RPC) mix nets. Such mix nets consist of several mix servers, where the mix servers use a public key encryption scheme with distributed decryption, with ElGamal being a common choice. That is, at the beginning each mix server generates a public/private key share. The public key shares are (publicly) combined to a single public key, say pk; the private key shares are kept secret by every mix server. Now, the input to a re-encryption RPC mix net is a list of ciphertexts (e.g., encrypted votes), where each ciphertext is obtained by encrypting a plaintext under pk and comes with a non-interactive zero knowledge proof (NIZKP) of knowledge of the plaintext. Now, the first mix server shuffles the ciphertexts and re-encrypts the ciphertexts.[1] The shuffled and re-encrypted ciphertexts form the output of this mix server and the input to the next mix server, which again shuffles and re-encrypts the ciphertexts, and so on. Each mix server also commits to the permutation it has used for shuffling. Once the last mix server has processed the list of ciphertexts in the described way, the mix servers together decrypt each ciphertext in the list output by the last mix server in a distributed way. In order to check if a mix server cheated, i.e., manipulated/replaced a ciphertext so that it carries a different plaintext, so-called random partial checking is performed for each mix server. For this purpose, every mix server is supposed to reveal some partial information about the input/output relation. Which information is to be revealed is randomly chosen by auditors. The auditing might take place before the decryption phase (possibly right after each mixing step) or only after the decryption phase.

We note that in the same paper, Jakobsson, Juels, and Rivest also consider Chaumian RPC mix nets, where the input ciphertexts are obtained by nested encryption (using different public keys for each layer of encryption) and every mix server, instead of performing re-encryption, peels of one layer of encryption.

From the design of RPC mix nets it is clear that they do not provide perfect security: there is some non-negligible probability that cheating goes undetected and some partial information about the input/output relation is revealed. As already argued by Jakobsson, Juels, and Rivest, in the context of e-voting the penalties for cheating would be so severe that being caught with some (even small) probability should deter a mix server from cheating.

Due to the simplicity and efficiency of re-encryption RPC mix nets, these mix nets have been used in implementations of several prominent e-voting systems, including Civitas [2] and Prêt à Voter [12]. They are, for example, also used in a variant of Prêt à Voter, which was used in the election of the Australian state of Victoria [3] in November 2014. According to the authors of [3], re-encryption RPC mix nets are attractive because of their efficiency and "the ease of explaining to the public how the mix net works". Some systems, such as Scantegrity [1], which has also been employed in real political elections, have used a similar technique.

Recently, Khazaei and Wikström [7] have pointed out several severe attacks on RPC mix nets as specified in the original work [5] and as implemented in several e-voting systems. They therefore suggested that re-encryption RPC mix nets should not be employed at all, but left as an open problem to prove or disprove that, with the fixes they suggest, Chaumian RPC mix nets provide sufficient security. Meanwhile, positive results for Chaumian RPC mix nets have been provided [10]. Yet, the findings by Khazaei and Wikström suggested the end of re-encryption RPC mix nets altogether.

In this paper, we show, however, that, under assumptions that seem reasonable in many practical situations, re-encryption RPC mix nets are still a viable option for the use in e-voting systems. More precisely, the main contributions of this paper are as follows.

Contributions of this paper. As mentioned, RPC mix nets can only provide restricted forms of verifiability and privacy. Therefore, we need security notions that allow us to measure

[1] Re-encryption is an operation that can be performed without knowledge of the private key or the plaintext. Given a ciphertext $\mathsf{Enc}_{pk}^{r}(m)$ obtained using the public key pk, the plaintext m, and the random coins r, re-encryption yields a ciphertext of the form $\mathsf{Enc}_{pk}^{r'}(m)$, i.e., one with a different random coin.

the level of security re-encryption RPC mix nets provide. For this purpose, we use a definition of privacy which has been used in the context of e-voting before (see, e.g., [9]) and which has also been employed for the analysis of Chaumian RPC mix nets in [10]. It focuses on the level of privacy for individual senders/voters and basically requires that for every pair of messages an adversary should not be able to tell which of two messages a sender has sent. As for verifiability, we study a stronger notion, namely accountability. While verifiability requires merely that misbehavior should be detectable, accountability, in addition, ensures that specific misbehaving parties can be blamed. This is crucial in order to deter parties from misbehaving. Our definition of accountability for re-encryption RPC mix nets follows the one proposed in [10], which in turn is based on a general domain independent definition of accountability proposed in [8].

We show that re-encryption RPC mix nets enjoy a reasonable level of accountability. Essentially, our accountability definition requires that the multiset of plaintexts in the input ciphertexts should be the same as the multiset of plaintexts in the output ciphertexts. We show that, if in the output ciphertexts k or more plaintexts have been modified (compared to the input), then this remains undetected with a probability of at most $(\frac{3}{4})^k$. If the manipulation is detected (which happens with a probability of at least $1-(\frac{3}{4})^k$), then at least one mix server can (rightly) be blamed for misbehaving. In order to prove this result, it was essential to take into account the improvements suggested by Khazaei and Wikström [7].

As mentioned before, Khazaei and Wikström [7] pointed out severe attacks on privacy for re-encryption RPC mix nets. In this paper, we make the following key observation, which is related to our result of accountability. If an adversary does not follow the protocol in an essentially semi-honest way, e.g., he does not commit to a permutation (but to some other function) or does not perform re-encryption of the ciphertexts, then he will be caught with a probability of at least $1/4$. Hence, whenever an adversary decides to deviate from this semi-honest behavior, he knows that he takes a relatively high risk of being caught. So, when penalties are severe and/or reputation can be lost, this risk will in many cases be sufficiently high to deter adversaries from deviating from this semi-honest behavior. We call adversaries that want to avoid being caught, but otherwise might be willing to cheat if this goes unnoticed, risk-avoiding. Now, for risk-avoiding adversaries, we show that re-encryption RPC mix nets provide a reasonable level of privacy, which, in fact, is quite close to the ideal case, where the adversary only learns the final output of the mix net.

Our results hold true no matter whether auditing is done before or after the decryption phase.

References

[1] R. Carback, D. Chaum, J. Clark, adn J. Conway, E. Essex, P.S. Herrnson, T. Mayberry, S. Popoveniuc, R. L. Rivest, E. Shen, A. T. Sherman, and P.L. Vora. Scantegrity II Municipal Election at Takoma Park: The First E2E Binding governmental Elecion with Ballot Privacy. In *USENIX Security Symposium/ACCURATE Electronic Voting Technology (USENIX 2010)*. USENIX Association, 2010.

[2] M. R. Clarkson, S. Chong, and A. C. Myers. Civitas: Toward a Secure Voting System. In *2008 IEEE Symposium on Security and Privacy (S&P 2008)*, pages 354–368. IEEE Computer Society, 2008.

[3] Chris Culnane, Peter Y. A. Ryan, Steve Schneider, and Vanessa Teague. vVote: a Verifiable Voting System (DRAFT). *CoRR*, abs/1404.6822, 2014. Available at http://arxiv.org/abs/1404.6822.

[4] Philippe Golle, Sheng Zhong, Dan Boneh, Markus Jakobsson, and Ari Juels. Optimistic Mixing for Exit-Polls. In Yuliang Zheng, editor, *Advances in Cryptology - ASIACRYPT 2002, 8th International Conference on the Theory and Application of Cryptology and Information Security, Proceedings*, volume 2501 of *Lecture Notes in Computer Science*, pages 451–465. Springer, 2002.

[5] M. Jakobsson, A. Juels, and R. L. Rivest. Making Mix Nets Robust for Electronic Voting by Randomized Partial Checking. In *USENIX Security Symposium*, pages 339–353, 2002.

[6] Shahram Khazaei, Tal Moran, and Douglas Wikström. A Mix-Net from Any CCA2 Secure Cryptosystem. In Xiaoyun Wang and Kazue Sako, editors, *Advances in Cryptology - ASIACRYPT 2012 - 18th International Conference on the Theory and Application of Cryptology and Information Security, Proceedings*, volume 7658 of *Lecture Notes in Computer Science*, pages 607–625. Springer, 2012.

[7] Shahram Khazaei and Douglas Wikström. Randomized Partial Checking Revisited. In Ed Dawson, editor, *Topics in Cryptology - CT-RSA 2013 - The Cryptographers' Track at the RSA Conference 2013. Proceedings*, volume 7779 of *Lecture Notes in Computer Science*, pages 115–128. Springer, 2013.

[8] Ralf Küsters, Tomasz Truderung, and Andreas Vogt. Accountability: Definition and Relationship to Verifiability. In *Proceedings of the 17th ACM Conference on Computer and Communications Security (CCS 2010)*, pages 526–535. ACM, 2010.

[9] Ralf Küsters, Tomasz Truderung, and Andreas Vogt. Verifiability, Privacy, and Coercion-Resistance: New Insights from a Case Study. In *32nd IEEE Symposium on Security and Privacy (S&P 2011)*, pages 538–553. IEEE Computer Society, 2011.

[10] Ralf Küsters, Tomasz Truderung, and Andreas Vogt. Formal Analysis of Chaumian Mix Nets with Randomized Partial Checking. In *35th IEEE Symposium on Security and Privacy (S&P 2014)*, pages 343–358. IEEE Computer Society, 2014.

[11] C. Andrew Neff. A verifiable secret shuffle and its application to e-voting. In Michael K. Reiter and Pierangela Samarati, editors, *8th ACM Conference on Computer and Communications Security (CCS 2001)*, pages 116–125. ACM, 2001.

[12] Peter Y. A. Ryan, David Bismark, James Heather, Steve Schneider, and Zhe Xia. The Prêt à Voter Verifiable Election System. Technical report, University of Luxem- bourg, University of Surrey, 2010. http://www.pretavoter.com/publications/PretaVoter2010.pdf.

[13] K. Sako and J. Kilian. Receipt-Free Mix-Type Voting Scheme — A practical solution to the implementation of a voting booth. In *Advances in Cryptology — EUROCRYPT '95, International Conference on the Theory and Application of Cryptographic Techniques*, volume 921 of *Lecture Notes in Computer Science*, pages 393–403. Springer-Verlag, 1995.

[14] Douglas Wikström. Five Practical Attacks for "Optimistic Mixing for Exit-Polls". In Mitsuru Matsui and Robert J. Zuccherato, editors, *Selected Areas in Cryptography, 10th Annual International Workshop, SAC 2003, Revised Papers*, volume 3006 of *Lecture Notes in Computer Science*, pages 160–175. Springer, 2003.

[15] Douglas Wikström. A Universally Composable Mix-Net. In Moni Naor, editor, *Theory of Cryptography, First Theory of Cryptography Conference, TCC 2004, Proceedings*, volume 2951 of *Lecture Notes in Computer Science*, pages 317–335. Springer, 2004.

MATor: Towards Measuring the Degree of Anonymity in Tor

Michael Backes, Aniket Kate, Praveen Manoharan, Sebastian Meiser, Esfandiar Mohammadi

CISPA, Saarland University

I. Introduction

The onion routing network Tor is a widely employed low-latency anonymous communication service [11]. To provide anonymity Tor routes a user's traffic through anonymizing proxies. In Tor the trust in these anonymizing proxies (also called nodes) is distributed over three nodes, instead of one proxy, which are chosen from more than 5000 volunteer nodes. Using these anonymizing proxies, Tor creates an anonymous channel for the user, which leads to the following central question from a user perspective:

> How anonymous is this channel that Tor creates, i.e., how likely is it that an adversary can deanonymize me?

Deriving the degree of a user's anonymity is challenging for such a complex system where each of the 5000 fluctuating nodes is entrusted with different bandwidth, and each node offers a different set of ports for a communication. Previous mathematically founded analyses abstract the Tor network by ignoring characteristics of Tor, such as the path selection algorithm, the varying entrusted bandwidth of different Tor nodes, or the user's requested ports [2], [5], [6], [9], [7], [8]. However, these real-life characteristics of Tor significantly influence a user's anonymity, which renders the previously proven bounds inaccurate.

Contribution. In this paper, we present MATor: the first system to derive sender, recipient and relationship anonymity guarantees based on Tor's real-life characteristics, such as its actual path selection strategy. MATor entails light-weight real-time monitors that compute sender, recipient and relationship anonymity guarantees based on the actual Tor consensus data and the user requested ports.

We apply our analysis technique to Tor Metrics data [10] to perform a comprehensive analysis of Tor's anonymity guarantees. To this end, we conduct a large scale evaluation of different path selection algorithms for a broad variety of trust models, ranging from simple adversaries that compromise a given number of Tor nodes, over geographic adversaries (e.g., adversaries that compromise all nodes within certain countries), up to complex adversary models that follow economic reasoning. Due to space restrictions we focus on Tor's standard path selection algorithm and on two adversaries: one that compromises a number of k arbitrary nodes and a simple geographical adversary, which compromises all nodes within a certain country.

II. MATor: Measuring anonymity guarantees

We developed the anonymity measurement tool MATor [3] which computes the impact of the path selection algorithm on the anonymity of a user. The tool uses the actual Tor metrics data for the measurement and enables the specification of a wide variety of adversary classes. In the full version [4], we use our theoretical framework AnoA [1], we prove that the results of MATor are secure.

Due to space constraints we present only two (very simple) example adversaries, namely an adversary that compromises a fixed number of k nodes and a geographical adversary that compromises all nodes within a country. In a technical report [4], we conducted extensive experiments with more complex adversary classes such as bandwidth-compromising adversaries, botnet-adversaries and adversaries that have a monetary budget.

A. A class of adversaries: budget adversaries

We do not only consider k-of-n adversaries, i.e., adversaries that freely compromise k arbitrary nodes within a set of n nodes. We also aim to capture more sophisticated adversary classes for different types of adversarial corruptions, such as corruption based on geo-locality, bandwidth, or cost-functions for every node n. Defining appropriate classes ensures that the adversary compromises nodes according to the considered restrictions.

Instead of defining an individual class for each of these considered adversary scenarios, we define a parametric adversary class that we call *budget-adversary* class, out of which we will instantiate all relevant individual adversary classes. The budget-adversary is parametric in a given *cost function* f that assigns costs to every node n within the Tor network, and in a *budget* G that the adversary may spend to corrupt nodes.

B. k-of-n adversaries

We begin with the k-of-N adversary model in which the adversary may compromise up to k nodes of its choice. This worst-case adversary is useful for estimating the maximal impact that a collaboration of a certain number

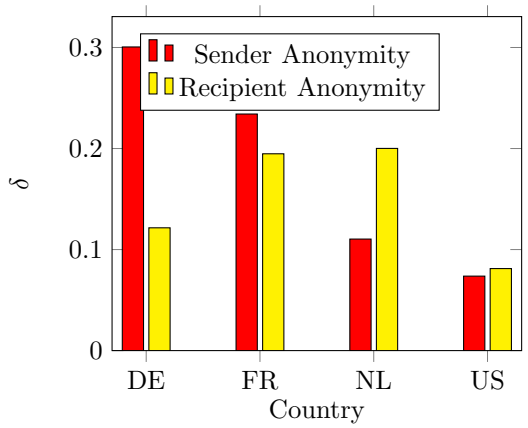


Figure 1. Advantage δ of a country based adversary, depending on the country for which it compromises all nodes, for the countries France (FR), Germany (DE), Netherlands (NL) and United States (US), ordered by the guarantee for Tor's path selection algorithm – results for sender anonymity and recipient anonymity. [4]

of participants can have on the anonymity within the Tor network. Such an adversary typically compromises the nodes with the largest weight and thus we expect this adversary to be stronger whenever the trust is not distributed evenly over the nodes. Formally, we instantiate our budget adversary class to model that the adversary may compromise k arbitrary nodes (out of all $N = |\mathcal{N}|$ Tor nodes), independent of their properties, by using $f^{k\text{-of-}N}(x) := 1$ for all nodes $x \in \mathcal{N}$. The adversary class is then $A^{k}_{f^{k\text{-of-}N}}$.

C. Geographic adversaries

We define a geographical adversary that is completely independent of bandwidth or a specific budget, as it can compromise all nodes that are located within a specific country. Such an adversary model can reflect the fear of a user that an oppressive regime tries to deanonymize the communication to find out either the sender or the recipient of the communication. In such scenarios, the user might fear that all nodes that lie within the geographical (or jurisdictional) border of a country can be compromised (e.g., forced to reveal information) by the regime. We formalize this intuition of geographical adversaries by first introducing a slight variant of budget adversary classes $A^{B=1}_{f^{\Pi}}$ for boolean predicates Π, where f^{Π} is defined as:

$$f^{\Pi}(x) = 0, \text{ if } \Pi(x) = 1 \text{ and } f^{\Pi}(x) = \infty \text{ otherwise.}$$

We then instantiate the predicate Π by country predicates Π_{C} for countries C, defined as

$$\Pi_{\mathrm{C}}(x) = 1, \text{ if } x.\mathsf{country} = \mathrm{C} \text{ and } \Pi_{\mathrm{C}}(x) = 0 \text{ otherwise.}$$

By choosing a country C, we can formally define an adversary that can eavesdrop on all nodes within this country, e.g., the adversary $A^{B=1}_{f^{\Pi_{\mathrm{NL}}}}$ with

$$f^{\Pi_{\mathrm{NL}}}(x) = \begin{cases} 0 & \text{if } x.\mathsf{country} = \mathrm{NL} \\ \infty & \text{otherwise} \end{cases}$$

compromises all nodes within the Netherlands (NL). In our analysis, we instantiate Π_{C} for all countries C we wish to analyze.

These geographical cost functions assume that the adversary can (basically for free) compromise all nodes within the country, but it cannot compromise other nodes. Such an adversary allows to evaluate how much impact a country has on the Tor network in terms of anonymity. We show the advantage δ of the geographical adversary for the four countries Germany (DE), France (FR), Netherlands (NL) and United States of America (US) in Figure 1. (This selection was made since, to improve readability, we have ordered the countries by the advantage of PSTor and selected the top four countries.)

References

[1] Michael Backes, Aniket Kate, Praveen Manoharan, Sebastian Meiser, and Esfandiar Mohammadi. AnoA: A Framework For Analyzing Anonymous Communication Protocols — Unified Definitions and Analyses of Anonymity Properties. available at http://www.infsec.cs.uni-saarland.de/~meiser/paper/anoa.html.

[2] Michael Backes, Aniket Kate, Praveen Manoharan, Sebastian Meiser, and Esfandiar Mohammadi. Anoa: A framework for analyzing anonymous communication protocols. In *Computer Security Foundations Symposium (CSF), 2013 IEEE 26th*, pages 163–178. IEEE, 2013.

[3] Michael Backes, Aniket Kate, Sebastian Meiser, and Esfandiar Mohammadi. (Nothing else) MATor(s): Monitoring the Anonymity of Tor's Path Selection. In *Proceedings of the 21st ACM Conference on Computer and Communications Security (CCS)*, pages 513–524. ACM, 2014.

[4] Michael Backes, Sebastian Meiser, and Marcin Slowik. Your Choice MATor(s): Assessing Tor Anonymity Guarantees for Different Path Selection Algorithms and Trust Models. available at http://www.infsec.cs.uni-saarland.de/~meiser/paper/anoa3.pdf.

[5] Joan Feigenbaum, Aaron Johnson, and Paul F. Syverson. A Model of Onion Routing with Provable Anonymity. In *Proc. 11th Conference on Financial Cryptography and Data Security (FC)*, pages 57–71, 2007.

[6] Joan Feigenbaum, Aaron Johnson, and Paul F. Syverson. Probabilistic Analysis of Onion Routing in a Black-Box Model. In *Proc. 6th ACM Workshop on Privacy in the Electronic Society (WPES)*, pages 1–10, 2007.

[7] Joan Feigenbaum, Aaron Johnson, and Paul F. Syverson. Probabilistic Analysis of Onion Routing in a Black-Box Model. *ACM Transactions on Information and System Security (TISSEC)*, 15(3):14, 2012.

[8] Nethanel Gelernter and Amir Herzberg. On the limits of provable anonymity. In *Proc. 12th ACM Workshop on Privacy in the Electronic Society (WPES)*, pages 225–236, 2013.

[9] Alejandro Hevia and Daniele Micciancio. An Indistinguishability-Based Characterization of Anonymous Channels. In *Proc. 8th Privacy Enhancing Technologies Symposium (PETS)*, pages 24–43, 2008.

[10] Tor Metrics Portal. https://metrics.torproject.org/. Accessed Feb 2014.

[11] The Tor Project. https://www.torproject.org/, 2003. Accessed Feb 2014.

Isabelle and Security

Jasmin Christian Blanchette[1,2] and Andrei Popescu[3]

[1] Inria Nancy & LORIA, Villers-lès-Nancy, France
[2] Max-Planck-Institut für Informatik, Saarbrücken, Germany
[3] Department of Computer Science, School of Science and Technology, Middlesex University, UK

Abstract. Isabelle/HOL is a general-purpose proof assistant based on higher-order logic. Its main strengths are its simple-yet-expressive logic and its proof automation. Security researchers make up a significant fraction of Isabelle's users. In the past few years, many exciting developments have taken place, connecting programming languages, operating system kernels, and security.

1 Isabelle

Isabelle [26, 27] is a generic theorem prover developed since the 1980s under the leadership of Lawrence Paulson (University of Cambridge), Tobias Nipkow (Technische Universität München), and Makarius Wenzel. Its built-in metalogic is a fragment of higher-order logic. The HOL object logic is a more elaborate version of higher-order logic, complete with the familiar connectives and quantifiers. Isabelle/HOL is the most developed instance of Isabelle. HOL is not quite as powerful as set theory or type theory (e.g., Coq's calculus of inductive constructions), but it can comfortably accommodate most applications, whether they are oriented toward mathematics or computer science.

Isabelle adheres to the tradition initiated in the 1970s by the LCF system: All inferences are derived by a small trusted kernel; types and functions are defined rather than axiomatized to guard against inconsistencies. High-level specification mechanisms let users define important classes of types and functions safely. When the user introduces a new (co)datatype or (co)recursive function, the system internally synthesizes an elaborate construction, from which it derives the characteristic (co)datatype properties and (co)recursive specifications as theorems.

Proof automation is another cornerstone of the Isabelle approach. The system provides proof methods based on term rewriting, tableaux, and arithmetic decision procedures. In addition, the Sledgehammer [32] tool integrates third-party automatic theorem provers, helping to discover more difficult proofs automatically. The counterexample generators Nitpick [7] and Quickcheck [9] complete the picture. They help identify invalid conjectures early, before the user invests hours in a doomed proof attempt.

2 Security in Isabelle

Several groups of researchers have formalized security-related results in Isabelle/HOL. The following partial survey attempts to give an overview of that work.

Already in the 1990s, Lawrence Paulson [31] verified a number of cryptographic protocols in Isabelle, notably (shared-key) Otway–Rees and (public-key) Needham–

Schroeder. He modeled each protocol as an inductively defined set of traces. Paulson's inductive approach has been highly influential, and despite the emergence of automatic tools such as ProVerif, it remains popular thanks to its flexibility and expressiveness [5].

David Basin and his team [4] developed the ProtoVeriPhy framework in Isabelle, for analyzing physical protocols. They verified protocols such as authenticated ranging, ultrasound distance bounding, delayed key disclosure, and secure time synchronization. They also proved the vulnerability of the Brands–Chaum protocol, including in a wrongly fixed version, and proved a revised proposal correct [10].

Gerwin Klein and his team proved functional correctness of the seL4 microkernel, consisting of 8830 lines of C code [20]. Equipped with an abstract, correct specification of seL4, they started proving security properties, including integrity [37] and information-flow enforcement [25].

In a separate but related line of work, Burkhart Wolff and his collaborators contributed the formalization of realistic security frameworks relevant for operating system verification, covering access control [8] and intransitive noninterference [40].

Heiko Mantel and his team developed several Isabelle formalizations related to information-flow security. The I-MAKS framework and tool is a mechanization of Mantel's Modular Assembly Kit for Security (MAKS), designed for the specification and verification of event systems (a form of I/O automata) with security requirements. Mantel's team also developed formal proofs for language-based security in the presence of concurrency: a rely–guarantee paradigm [15], security type systems for noninterference [14], and declassification [13].

Gregor Snelting and his team are developing JOANA [38], a mature software security analysis framework for Java, covering both source code and bytecode. Its features are based on complex program analysis techniques, whose correctness is difficult to comprehend. The team uses Isabelle to verify different aspects of JOANA [41], based on the formal semantics of a large fragment of concurrent Java [22]. Snelting's team has also looked into language-based security, contributing, in parallel with Barthe and Nieto's [3] and Beringer and Hofmann's [6] works, the first Isabelle formalizations of Volpano–Smith-style security type systems [39].

Tobias Nipkow, Andrei Popescu, and their colleagues in Munich and Saarbrücken formalized possibilistic and probabilistic noninterference for a multithreaded while language [34, 35]. They unified various security concepts and type systems from the literature and simplified their proofs of correctness [28]. They also formalized HyperCTL*, a temporal logic for expressing security properties [36].

Another major development by the Munich team is CoCon, a conference management system with document confidentiality guarantees [19]. The system's kernel is written and certified in Isabelle and extracted as functional code. The dozen of submissions to the Isabelle 2014 workshop were managed using CoCon, and the TABLEAUX 2015 conference is expected to use it as well. Other events are welcome to use it.[1]

Further recent security formalizations in Isabelle include noninterference for process algebras [30], cryptography [21], and network security [12].

Finally, Isabelle is a main verification tool of Reliably Secure Software Systems (RS^3) [24], a priority program of the Deutsche Forschungsgemeinschaft, coordinated

[1] `http://www21.in.tum.de/~popescua/rs3/GNE.html`

by Heiko Mantel. RS^3 encompasses twelve main projects and six associated projects, all focused on different aspects of information-flow security. Within RS^3, Isabelle is used for verifying actor implementations of multi-agent systems (Poetzsch-Heffter et al. [33]), workflow management systems (Hutter et al. [17] and Nipkow et al. [29]), and security types (Mantel et al. [23] and Nipkow et al. [29]).

3 Security in Other Proof Assistants

Isabelle is by no means the only proof assistant employed in security verification. Impressive recent verification achievements using other systems include an aircraft microprocessor [16] (in ACL2), a hardware architecture with information-flow primitives [1] (in Coq), a separation kernel [11] (in HOL4), a browser kernel [18] (in Coq), and a quasi-automatic tool for reasoning about cryptographic protocols [2] (based on Coq).

References

[1] de Amorim, A.A., Collins, N., DeHon, A., Demange, D., Hritcu, C., Pichardie, D., Pierce, B.C., Pollack, R., Tolmach, A.: A verified information-flow architecture. In: Principles of Programming Languages (POPL 2014). pp. 165–178 (2014)

[2] Barthe, G., Crespo, J.M., Grégoire, B., Kunz, C., Béguelin, S.Z.: Computer-aided cryptographic proofs. In: Beringer, L., Felty, A. (eds.) Interactive Theorem Proving (ITP 2012). LNCS, vol. 7406, pp. 11–27. Springer (2012)

[3] Barthe, G., Nieto, L.P.: Secure information flow for a concurrent language with scheduling. J. Comput. Sec. 15(6), 647–689 (2007)

[4] Basin, D.A., Capkun, S., Schaller, P., Schmidt, B.: Formal reasoning about physical properties of security protocols. ACM Trans. Inf. Syst. Secur. 14(2), 16 (2011)

[5] Bella, G.: Inductive study of confidentiality. In: Klein, G., Nipkow, T., Paulson, L. (eds.) Archive of Formal Proofs. `http://afp.sf.net/entries/Inductive_Confidentiality.shtml` (2012)

[6] Beringer, L., Hofmann, M.: Secure information flow and program logics. In: Klein, G., Nipkow, T., Paulson, L. (eds.) Archive of Formal Proofs. `http://afp.sf.net/entries/SIFPL.shtml` (2008)

[7] Blanchette, J.C., Nipkow, T.: Nitpick: A counterexample generator for higher-order logic based on a relational model finder. In: Kaufmann, M., Paulson, L.C. (eds.) Interactive Theorem Proving (ITP 2010). LNCS, vol. 6172, pp. 131–146. Springer (2010)

[8] Brucker, A.D., Brügger, L., Wolff, B.: The Unified Policy Framework (UPF). In: Klein, G., Nipkow, T., Paulson, L. (eds.) Archive of Formal Proofs. `http://afp.sf.net/entries/UPF.shtml` (2014)

[9] Bulwahn, L.: The new Quickcheck for Isabelle—Random, exhaustive and symbolic testing under one roof. In: Hawblitzel, C., Miller, D. (eds.) Certified Programs and Proofs (CPP 2012). LNCS, vol. 7679, pp. 92–108. Springer (2012)

[10] Cremers, C.J.F., Rasmussen, K.B., Schmidt, B., Čapkun, S.: Distance hijacking attacks on distance bounding protocols. In: IEEE Symposium on Security and Privacy (SP 2012). pp. 113–127. IEEE (2012)

[11] Dam, M., Guanciale, R., Khakpour, N., Nemati, H., Schwarz, O.: Formal verification of information flow security for a simple ARM-based separation kernel. In: Computer and Communications Security (CCS '13). pp. 223–234 (2013)

[12] Diekmann, C.: Network security policy verification. In: Klein, G., Nipkow, T., Paulson, L. (eds.) Archive of Formal Proofs. http://afp.sf.net/entries/Network_Security_Policy_Verification.shtml (2014)

[13] Grewe, S., Lux, A., Mantel, H., Sauer, J.: A formalization of declassification with WHAT-and-WHERE-security. In: Klein, G., Nipkow, T., Paulson, L. (eds.) Archive of Formal Proofs. http://afp.sf.net/entries/WHATandWHERE_Security.shtml (2014)

[14] Grewe, S., Lux, A., Mantel, H., Sauer, J.: A formalization of strong security. In: Klein, G., Nipkow, T., Paulson, L. (eds.) Archive of Formal Proofs. http://afp.sf.net/entries/Strong_Security.shtml (2014)

[15] Grewe, S., Mantel, H., Schoepe, D.: A formalization of assumptions and guarantees for compositional noninterference. In: Klein, G., Nipkow, T., Paulson, L. (eds.) Archive of Formal Proofs. http://afp.sf.net/entries/SIFUM_Type_Systems.shtml (2014)

[16] Hardin, D.S., Smith, E.W., Young, W.D.: A robust machine code proof framework for highly secure applications. In: Manolios, P., Wilding, M. (eds.) The ACL2 Theorem Prover and Its Applications (ACL2 2006). pp. 11–20. ACM (2006)

[17] Hutter, D., et al.: MORES: Modelling and Refinement of Security Requirements on Data and Processes. http://www-cps.hb.dfki.de/research/projects/MORES (2014)

[18] Jang, D., Tatlock, Z., Lerner, S.: Establishing browser security guarantees through formal shim verification. In: Kohno, T. (ed.) USENIX Security '12. pp. 113–128. USENIX (2012)

[19] Kanav, S., Lammich, P., Popescu, A.: A conference management system with verified document confidentiality. In: Biere, A., Bloem, R. (eds.) Computer Aided Verification (CAV 2014). LNCS, vol. 8559, pp. 167–183. Springer (2014)

[20] Klein, G., Andronick, J., Elphinstone, K., Heiser, G., Cock, D., Derrin, P., Elkaduwe, D., Engelhardt, K., Kolanski, R., Norrish, M., Sewell, T., Tuch, H., Winwood, S.: seL4: Formal verification of an operating-system kernel. Commun. ACM 53(6), 107–115 (2010)

[21] Lindenberg, C., Wirt, K., Buchmann, J.: Formal proof for the correctness of RSA-PSS. IACR Cryptology ePrint Archive 2006, 11 (2006)

[22] Lochbihler, A.: Making the Java memory model safe. ACM Trans. Program. Lang. Syst. 35(4), 12:1–65 (2014)

[23] Mantel, H., et al.: Reliable Security for Concurrent Programs (RSCP). http://www.mais.informatik.tu-darmstadt.de/RS3-RSCP.html (2014)

[24] Mantel, H., et al.: Reliably Secure Software Systems (RS^3). http://www.spp-rs3.de/ (2014)

[25] Murray, T.C., Matichuk, D., Brassil, M., Gammie, P., Bourke, T., Seefried, S., Lewis, C., Gao, X., Klein, G.: seL4: From general purpose to a proof of information flow enforcement. In: IEEE Symposium on Security and Privacy (SP 2013). pp. 415–429. IEEE (2013)

[26] Nipkow, T., Klein, G.: Concrete Semantics—with Isabelle/HOL. Springer (2014)

[27] Nipkow, T., Paulson, L.C., Wenzel, M.: Isabelle/HOL: A Proof Assistant for Higher-Order Logic, LNCS, vol. 2283. Springer (2002)

[28] Nipkow, T., Popescu, A.: Making security type systems less ad hoc. it—Information Technology 56(6), 267–272 (2014)

[29] Nipkow, T., Weidenbach, C., et al.: Security Type Systems and Deduction (SecDed). http://www4.informatik.tu-muenchen.de/proj/theoremprov/local_projects/rs3.html (2014)

[30] Noce, P.: Noninterference security in communicating sequential processes. In: Klein, G., Nipkow, T., Paulson, L. (eds.) Archive of Formal Proofs. http://afp.sf.net/entries/Noninterference_CSP.shtml (2014)

[31] Paulson, L.C.: The inductive approach to verifying cryptographic protocols. J. Comput. Secur. 6(1–2), 85–128 (1998)

[32] Paulson, L.C., Blanchette, J.C.: Three years of experience with Sledgehammer, a practical link between automatic and interactive theorem provers. In: Sutcliffe, G., Schulz, S., Ternovska, E. (eds.) International Workshop on the Implementation of Logics (IWIL 2010). EPiC Series, vol. 2, pp. 1–11. EasyChair (2012)

[33] Poetzsch-Heffter, A., et al.: Modular Verification of Security Properties in Actor Implementations (MoVeSPAcI). https://softech.informatik.uni-kl.de/homepage/en/research/MoVeSPAcI/ (2014)

[34] Popescu, A., Hölzl, J., Nipkow, T.: Proving concurrent noninterference. In: Hawblitzel, C., Miller, D. (eds.) Certified Programs and Proofs (CPP 2012). LNCS, vol. 7679, pp. 109–125. Springer (2012)

[35] Popescu, A., Hölzl, J., Nipkow, T.: Formalizing probabilistic noninterference. In: Gonthier, G., Norrish, M. (eds.) Certified Programs and Proofs (CPP 2013). LNCS, vol. 8307, pp. 259–275. Springer (2013)

[36] Rabe, M.N., Lammich, P., Popescu, A.: A shallow embedding of HyperCTL*. In: Klein, G., Nipkow, T., Paulson, L. (eds.) Archive of Formal Proofs. http://afp.sf.net/entries/HyperCTL.shtml (2014)

[37] Sewell, T., Winwood, S., Gammie, P., Murray, T.C., Andronick, J., Klein, G.: seL4 enforces integrity. In: van Eekelen, M.C.J.D., Geuvers, H., Schmaltz, J., Wiedijk, F. (eds.) Interactive Theorem Proving (ITP 2011). LNCS, vol. 6898, pp. 325–340. Springer (2011)

[38] Snelting, G., Giffhorn, D., Graf, J., Hammer, C., Hecker, M., Mohr, M., Wasserrab, D.: Checking probabilistic noninterference using JOANA. it—Information Technology 56, 280–287 (2014)

[39] Snelting, G., Wasserrab, D.: A correctness proof for the Volpano/Smith security typing system. In: Klein, G., Nipkow, T., Paulson, L. (eds.) Archive of Formal Proofs. http://afp.sf.net/entries/VolpanoSmith.shtml (2008)

[40] Verbeek, F., Tverdyshev, S., Havle, O., Blasum, H., Langenstein, B., Stephan, W., Nemouchi, Y., Feliachi, A., Wolff, B., Schmaltz, J.: Formal specification of a generic separation kernel. In: Klein, G., Nipkow, T., Paulson, L. (eds.) Archive of Formal Proofs. http://afp.sf.net/entries/CISC-Kernel.shtml (2014)

[41] Wasserrab, D., Lochbihler, A.: Formalizing a framework for dynamic slicing of program dependence graphs in Isabelle/HOL. In: Mohamed, O.A., Muñoz, C.A., Tahar, S. (eds.) Theorem Proving in Higher Order Logics (TPHOLs 2008). LNCS, vol. 5170, pp. 294–309. Springer (2008)

Sensitivity Assessment of Personal Information in Heterogeneous Data

Michael Backes
CISPA, Saarland University & MPI-SWS
backes@cs.uni-saarland.de

Praveen Manoharan
CISPA, Saarland University
manoharan@cs.uni-saarland.de

***Abstract*—Billions of users use the Internet and its various services on a daily basis. In the course of these activities, they disseminate a plethora of personal information, often without recognizing and being able to assess the potentially detrimental effects on their privacy. In fact, rigorously assessing the likelihood of a specific user revealing sensitive information to an adversary that observes public actions of the user presents significant scientific challenges, such as dealing with the heterogeneous nature of information disseminated online, as well as formalizing the notions of sensitivity and criticality of user information.**

In this work, we propose *ASPI*: a formal framework for Assessing the Sensitivity of Personal Information in heterogeneous data, and for reasoning about the dissemination of sensitive information and the adversarial learning process that reveals this sensitive information to the adversary. We show that this learning process can be greatly simplified if we assume an adversary that makes statistical decisions based on maximum likelihood, and further provide a taxonomy of critical information that allows an adversary to infer sensitive information about the user. We propose a method to automatically identify the special type of linking-critical information based on the information the user has already disseminated. Finally, we provide an instantiation of *ASPI* to link our approach to the widely explored statistical-database setting, and show how to identify critical database columns that enable an adversary to link a user to his database entry.

I. Introduction

Billions of users use the Internet and its various services on a daily basis. In the course of these activities, they disseminate a multitude of personal information, often without recognizing and being able to assess the potentially detrimental effects on their privacy. In fact, an abundance of cases have become known in recent years in which knowledge about sensitive personal information has been abused, ranging from identity theft and online fraud to severe cases of cyber-bullying and actual life-threatening actions. Users have started to react by keeping sensitive information separate from their real persona, typically by using different pseudonyms for different services. However, these pseudonyms can often still be linked back to the user using inference-based reasoning on the variety of available user information that is deemed security-uncritical; see the de-anonymization of Netflix data as a particularly illustrative and impactful example [1].

While there has been significant work on the protection of sensitive information, typically in the context of statistical databases [2], [3], [4], [5], identifying sensitive information in the open setting of the Internet with its highly dynamic, heterogeneous data constitutes a highly ambitious challenge. In particular, in such open settings like the Internet, even gaining a rigorous understanding of what it means for personal information to be sensitive (or better: under which conditions personal information should be considered sensitive) is a largely unexplored field. Amongst several other challenges, soundly assessing the sensitivity and criticality of information contained in data disseminated through the Internet requires a suitable formalization of online interactions that copes with the heterogeneous nature of data shared in the Internet, the subjective nature of information sensitivity and the approximation of information inference through a possible adversary. In particular, the differentiation between actually sensitive information that needs to be protected, and critical information that allows an adversary to infer sensitive information, has, to our knowledge, not been targeted in the privacy literature.

A. Contribution

In this work, we introduce *ASPI*, a formal framework for assessing the sensitivity and criticality of personal information in heterogeneous data shared in day-to-day online activities. We base our framework on an abstract data model introduced by Backes et al. [6] that allows us to cope with the heterogeneous nature of information disseminated throughout the Internet.

In order to retain full generality, *ASPI* defines sensitivity of personal information through user specified privacy policies: these policies define exactly which information should not be connected to the user, and is therefore perceived as sensitive.

ASPI then formalizes the adversarial learning process in a probabilistic transition system to suitably capture inference of sensitive information: each state in these transition systems represents the adversary's knowledge about the user, and each transition between different states is caused by a public user action from which the adversary infers further personal information about the user. While these transitions are inherently probabilistic due to the uncertainty of the inference process, we show that this probabilistic transition system can be greatly simplified for an adversary that only accepts inferred user information if he is sufficiently confident in the correctness of this information.

This formalization of the adversarial learning process allows us to identify the different types of information that lead the adversary to learn sensitive information about the user: through *ASPI*, we derive a taxonomy of critical information which differentiates between three types of critical information: sen-

sitive information, inference-critical information which allows for the direct inference of sensitive information, and linking-critical information, which allows for the inference of sensitive information by linking different profiles of the same user.

For linking-critical information, in particular, we propose a novel method for identifying such information based on the context in which the information is used. We then show that, if the user avoids disseminating information identified by this method, she also avoids revealing sensitive information to the adversary by allowing him to link the user's pseudonymous profiles.

Last, we apply our framework to the traditional statistical database setting: given a set of sensitive columns in the database, we identify which columns are critical and could cause a specific user to be linked to this database entry. Furthermore we show that the absence of critical columns implies k-anonymity for the database, thereby validating the notions introduced in *ASPI*.

REFERENCES

[1] A. Narayanan and V. Shmatikov, "Robust de-anonymization of large sparse datasets," in *Proceedings of the 2008 IEEE Symposium on Security and Privacy (S& P)*, 2008, pp. 111–125.

[2] L. Sweeney, "K-Anonymity: A Model for Protecting Privacy," *Int. J. Uncertain. Fuzziness Knowl.-Based Syst.*, vol. 10, no. 5, pp. 557–570, 2002.

[3] A. Machanavajjhala, D. Kifer, J. Gehrke, and M. Venkitasubramaniam, "L-Diversity: Privacy Beyond K-anonymity," *ACM Trans. Knowl. Discov. Data*, vol. 1, no. 1, 2007.

[4] N. Li and T. Li, "t-closeness: Privacy beyond k-anonymity and -diversity," in *In Proceedings of the 23rd International Conference on Data Engineering (ICDE*, 2007.

[5] C. Dwork, "Differential Privacy: A Survey of Results," in *Proceedings of the 5th International Conference on Theory and Applications of Models of Computation*, 2008, pp. 1–19.

[6] M. Backes, P. Berrang, and P. Manoharan, "How well do you blend into the crowd? - d–convergence: a novel paradigm for reasoning about privacy in the age of Big-Data," http://arxiv.org/abs/1502.03346, 2015, eprint arXiv:1502.03346 – cs.CR.

Assessing IT Security Standards Against the Upcoming GDPR for Cloud Systems

Cesare Bartolini, Gabriela Gheorghe, Andra Giurgiu
Mehrdad Sabetzadeh, Nicolas Sannier
{*firstname.lastname*}@uni.lu

The protection of personal data has seen a major upheaval in the last years, with a growing attention from legislators, entrepreneurs, developers, authorities and the general public. This is related to the increasing adoption of cloud-based services, and the focus on personal data as a pivotal asset in modern business models. The fact that personal data have a significant monetary value is proven by the emergence of many "free" services. The benefit for a company providing such services (and possibly its only source of income) stems from processing such personal data, especially selling them to third parties.

The main EU legal instrument that sets the general rules for the processing of personal data is Directive 95/46/EC, which gives Data Subjects (DSs) a set of rights with respect to such processing, states the obligations controllers and processors have to comply with when dealing with personal data and foresees oversight authorities and mechanisms meant to safeguard adherence to these rules. The same general rules apply when data is stored or otherwise processed in the cloud. However, the fast-paced evolution of technology over the last two decades has exposed several weaknesses of the current legal framework, calling for an adaptation of the legislation. A reform is currently under development, and after more than two years since its official release it is reaching its final stages. It is expected to be finalized by the end of 2015, thus entering into force in 2017, at the earliest.

The reform is composed of a Directive for judicial cooperation in criminal matters, as well as a widely-applicable General Data Protection Regulation (GDPR). The latter shall replace the current Directive 95/46/EC. The Regulation builds on the principles and rules of the pre-existing Directive, but aims to enhance the rights of the DS. Also, it emphasizes the responsibility of the data controllers and processors and increases the sanctions for violations of its provisions.

The new Regulation will place a significant burden on businesses involved in the processing of personal data. Enterprises will be required to comply with a new regime which is rather vague, building on concepts such as *appropriate measures* or *legitimate purpose*. While enterprises will have a significant interest in being compliant with the GDPR, they are faced with the absence of any concrete guideline or consolidated approach to defining compliance with these requirements. At this point, what is missing is an understanding of the legal and technical challenges in achieving compliance with GDPR requirements. We believe that the academic community needs to discover the overlapping topics where legal requirements meet business practices in cloud service provisioning.

If we can identify gaps between what data protection regulations stipulate and what is it technically achievable in terms of compliance, we can better understand where to direct meaningful research focus.

While taking into account security concerns for IT systems, the current industrial practice needs to consider the ISO/IEC 27000 standard series that provides a framework to handle concepts such as security policy and objectives, risk definitions and assessment, commitment for continuous evaluation and documentation. In particular, the IT community has already widely accepted the ISO/IEC 27001-2005 (and its last revision ISO/IEC 27001-2013), not only as a business competitive advantage, but also a must-have standard certification for enterprises. The ISO/IEC 27001 certification has progressively become a client requirement: for instance, it is required for the PFS (Professionals of the Financial Sector) agreement delivered by CSSF (Commission de Surveillance du Secteur Financier, an agency that monitor the financial sector in Luxembourg), a business requirement in the Luxembourgish financial sector. The IT community is also showing interest to the new ISO/IEC 27018-2014[1], a standard targeting cloud services. Unfortunately, it lacks practices and return of experience. Since December 2008, the Cloud Security Alliance (CSA) gathers cloud practitioners and companies in order to promote the use of best practices for providing security assurance within cloud computing. They also propose training, based on their open certification framework CSA STAR (Security, Trust & Assurance Registry), that leverages the requirements and control points of ISO/IEC 27001. The CSA is also aware of the difficulties that the new Regulation will entail [1]. In its updated report on the survey about the top threats in cloud computing [2], privacy and data protection are not listed, but many of the threats and suggested best practices therein match some of the provisions and duties within the GDPR, such as risk assessment and data integrity.

The idea of the present approach is to analyze the ISO 27001 standard and the Regulation, extracting the main concepts from both texts. We aim to find a mapping of the concepts expressed by each of these documents. Such an analysis can be used as a starting point to define criteria for GDPR compliance.

In the absence of clear rules and constraints, identifying security standards that can be applied to data protection to bridge the gap between the current practices and the future legal requirements can increase the DSs' trust and provide competitive advantages. It can also ease the transition to a new, consolidated approach to personal data protection.

References

[1] Francoise Gilbert. *What the Proposed EU Data Protection Regulation Means for Cloud Users*. Tech. rep. `https://downloads.cloudsecurityalliance.org/initiatives/clic/CLIC-Proposed-EUDataProtection-20120202.pdf`. Retrieved on 2015, February 9. CSA Legal Information Center (CLIC), Feb. 2013.

[1] Information technology – Security techniques – Code of practice for protection of personally identifiable information in public clouds.

[2] Top Threats Working Group. *The Notorious Nine: Cloud Computing Top Threats in 2013*. Tech. rep. `https://downloads.cloudsecurityalliance.org/initiatives/top_threats/The_Notorious_Nine_Cloud_Computing_Top_Threats_in_2013.pdf`. Retrieved on 2015, February 9. Cloud Security Alliance, Feb. 2013.

Security on medical data sharing
(a literature review)

Dayana Spagnuelo, Gabriele Lenzini
University of Luxembourg
Email: {dayana.spagnuelo, gabriele.lenzini} @uni.lu

Keywords—literature review, security, medical data, electronic health records.

I. Introduction

Medical records (e.g., test results and health reports) are about patients. Hospitals and healthcare institutions generate them after a patient's visit. Today they are digitized, stored electronically, and accessed remotely by professionals.

European directives suggest that patients should access these records too. Besides, they say, patients should have control over these data and be informed if and when their records are shared and how secure they are [1]. These requirements are hard to be met.

From a patient's perspective, the viewpoint of this paper, it may be easier to address at least one of such requirements: to inform patients about how secure their data are. This is a property usually referred as *transparency*, but a clear meaning of the word is still missing. According to [2] transparency ought to be regarded as an additional feature that qualifies security. So, security can be said to be transparent when is intelligible to human user. It opposes an opaque security, which holds technically but without the user's being aware of it. Thus, transparency is a socio-technical security property.

Transparency, is not a new term. It has been proposed in relation to *Transparency Enhancing Tools* (TETs) [3]. These are usually browser extensions that read out web server's privacy policies and inform users concisely, for instance, that a web server records the user's whereabouts and may sell the user's data to third parties. TETs have been discussed in relation to electronic health records [2], but no concrete solution has been proposed. Transparency in the medical domain is still an unfulfilled requirement.

Contribution. We survey the literature in medical data sharing and discusses what are the main security concerns in it. We intend also to figure out whether transparency is debated in that domain, in relation to which other properties, and which meaning and role are given to it.

II. Methodology and tools

We browsed the state of the art by searching for papers via "Findit.lu"[1]. This is the largest library portal in Luxembourg, and it is entirely dedicated on searching for electronic contents. It indexes a large number of important scientific digital libraries such as, among many others, LNCS, the ACM Digital Library, IEEEXplore, ScienceDirect, Scopus, and Medline.

[1]The portal is accessible via www.bibnet.lu, or directly at, www.findit.lu

We queried for "Security" and "Medical Data Sharing", and we looked for papers containing them in the title, in the abstract, in the list of keywords, and in the entire body. We chose "Security" because it is a general term: we expect that a paper that addresses more precise security properties will also mention "security" somewhere its text. We chose "Medical Data Sharing" to refine our domain to papers that discuss sharing medical data.

First, we queried without constraints on the year of publication. We got as many as 526 articles, too many for us to be able to read or scan them all. Thus, we restricted the focus to the last ten years, from 2004 to now. Excluding the repeated results and the papers not available for download, our pool shrank down to a total of 75 papers. We read the abstract and skimmed through the content of all of them. It turned out that 20 papers were about medical data sharing but with no focus on "security": the word appeared to be mentioned but the concept is not discussed. We discarded those papers and, after this skimming, we were left with a pool of 55 papers.

We organized our findings around one question: "*what particular security property the paper is about?*". To answer this question helped us to classify the papers depending on the property, or properties, they debate. It also helps us to understand whether transparency is considered as a security requirement and, if it is, in relation to which other property.

III. Major findings

Answering our main question, and so looking into what security properties our pool of papers is about, lead us to identify eight main security categories, each concerning policies, tools, or techniques meant to guarantee, preserve, or enforce a specific property. The 8 categories are the following: *Privacy*, concerning to provide anonymity to the data owner or to empower her to define who can operate on the data; *User authentication*, concerning to enhance the way in which users are authenticated electronically; *Access control*, concerning better ways to define who can access medical data and in what circumstances; *Data authenticity*, concerning to prove that the data origin is authentic, that is coming from the source as it is claimed; *Data Integrity*, concerning solutions to guarantee and prove that the data have not been manipulated or tampered with; *Confidentiality*, concerning to prevent the disclosure of data content to non-authorized third parts; *Auditability*, concerning to help the data owner to retrieve information clarifying how her data is being used; *Transparency*, concerning to guarantee openness about security policies and processes.

Most of the surveyed papers argue about data confidentiality (see Figure 1). This property is invoked in relation

to protect the data transmitted in open channels, such as the internet, or stored in open data bases, such as the cloud. One comment is mandatory: in the pool "confidentiality" there are 27 papers, namely [4]–[30]. Some of those were, per keywords, first gathered under "privacy". A closer look revealed that they are using the term inappropriately since their concern is mainly about encrypting data. But, encryption *per se* is insufficient to guarantee that the user's personal and sensitive information remains private during the whole data life cycle; more sophisticated techniques have to be in place for privacy to be protected. Thus, we decided to re-classify those works as being about confidentiality, adding those up to the ones already in that category.

Confidentiality is constantly discuss together with data integrity and data authenticity. That is because encryption is the technique that is more often adopted to enforce confidentiality in medical systems and the same technique is also proposed for data authenticity and integrity. In a total of 16 papers about data integrity (i.e., [5], [6], [8]–[10], [12], [13], [16], [18], [22], [23], [25], [29], [31]–[33]) only three works do not discuss confidentiality. We observed a very similar scenario with the category data authenticity. Only three works do not discuss confidentiality, out of 9 papers discussing data authenticity (i.e., [8], [9], [12], [22], [25], [29], [31]–[33]). Also, all works that examine data authenticity discuss data integrity too.

After confidentiality, the second and third most discussed security properties are privacy and access control. We found out that 20 works discuss privacy (the correct interpretation of this term) [14], [20], [25], [26], [30], [33]–[47], and that 19 papers discuss access control [11], [13], [19], [22], [23], [25], [29], [34], [37], [41]–[43], [45], [48]–[53].

User authentication seems not a major concerns as it is present only in 3 papers [13], [37], [54]. We do not have enough data to justify this lack of interest in authentication, but we can speculate on it. An hypothesis we have is that most of the works give for granted that medical data are accessed only by professionals and that they are considered trustworthy. Similarly, we claim that the lack of interest in user authentication may indicate that there is not yet a widespread concern about opening the access of the health data to patients. This is, indeed, a requirement that only very recently has been debated and brought to the attention of the society. If concrete actions to open up access to patients were taken

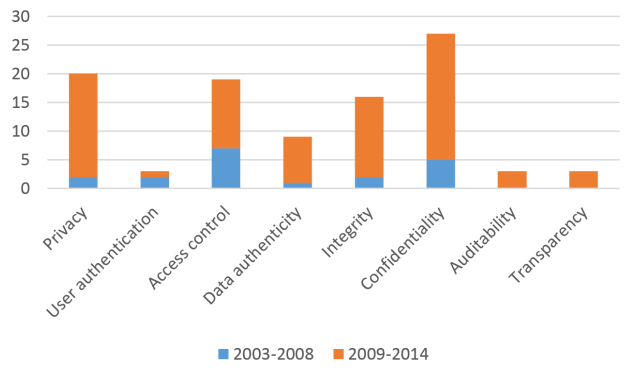


Fig. 1. Number of papers published per category from 2004 to now. We distinguished the first from the second 5 years.

into consideration, it would, we expect, raise more attention about identification and authentication. Indeed the works which discuss such a feature have identification and authentication as their main requirement (e.g., see [55]). A similar speculation, i.e., that the patient-centred approach is not yet under the bull's-eye in medical data security, concerns also the last two properties, transparency – the one of interest for this paper– and auditability. Auditability is subject of discussion of only 3 papers [33], [47], [56], *ex equo* (so to speak) with transparency which is mentioned as well in 3 papers [36], [42], [47].

Transparency is regarded as openness about policies and processes (we quote, "*there should be openness and transparency about policies, procedures, and technologies that directly affect individuals and/or their individually identifiable health information*" [36]) as well as a predisposition to increase responsibility and therefore presented with accountability (we quote, "*Transparency and accountability will be critical to helping society manage the privacy risks that accumulate from expeditious progress in communication, storage, and search technology*" [47]). Relevantly for this work, Routsalainen et al [42] propose transparency as the property to be informative towards patient. In fact they point out the lack of transparency since " *[the] patient is not automatically aware which professionals or entities are processing her EHR and for what purposes. [The] patient are not aware of all disclosures of the content of her EHR".*

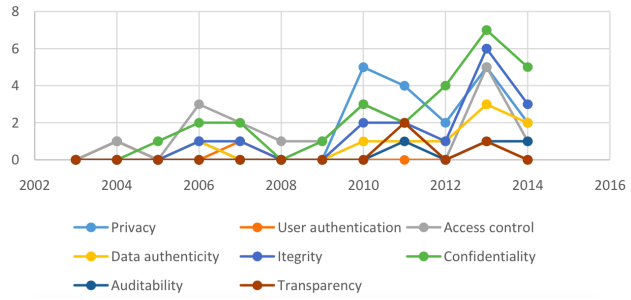
IV. Discussion and Conclusion

Our review has an obvious limitation: it considers papers that matched only two key-phrases, "security" and "medical data sharing". However, "security" is a generic terms under which we were able to find papers discussing more specific properties and requirements. "Medical data sharing" is our target, so this choice is justified. Still one could question why we did not searched for synonyms, and whether, in not doing so, we missed some important papers. Our searching on the whole body of the paper, however, was sufficient to catch works about electronic health records, bio-medical data, health care information systems, health-grid. Therefore, we judged the choice of our key-phrases sufficiently good for our scope.

This survey, organized around the works published in the last 10 years, shows that confidentiality and privacy are the major concerns in security for medical data (see also Figure 2). This comes with no surprise. About transparency, the survey shows that this requirement has just began to be addressed; all the considered papers see transparency related to inform users and make policies and processes openly available. This seems to be the interpretation of "transparency" in the medical domain, a meaning which matches what we propose. However, there is no formalization of it and no standard solution that makes a medical system compliant to it.

We also observed that the majority of papers were published in the last 5 years, which endorses the hypothesis that security is a relative young concern in medical systems engineering. Although we already had some hint of it, after having looked at the recent growth of interest as this survey reports, it is evident that there is still little attention from the security community towards auditability, transparency, and user authentication, at least in relation to medical data systems. (We

Fig. 2. Number of papers per year per category



did not searched into the literature of auditability and checked for use cases on medical data (e.g., as in [57]). Auditability and transparency are essential wherever humans need to be informed about practices in sharing sensitive personal data. No solution exists to comply with current EU regulations on this. Our first impression is that both categories are relatively understudied in the medical sectors. We expect a growth in attention to these properties as the idea of user empowerment will get more popular. User authentication seems suspiciously undervalued in the papers we surveyed. It is hard, from the data we have, to infer why. It may be that there are already good-enough authentication solutions to which medical systems can resort to. But, if we have to attempt another explanation, we are keen to suppose that current medical data are accessed mainly by professionals and that these roles are assumed to be trustworthy. Authentication is therefore implemented by simple login and password. Similarly as what we claimed while discussing transparency, if the EU directive suggesting to let users access their medical data should take off, we expect the problem of user authentication to became a pillar for the working of other several security features, and to foster a renewed interest.

REFERENCES

[1] E. P. E. Commission, "EU Directive 95/46/EC - The Data Protection Directive - IP/12/46 - 25/01/2012," October 2005 and 2012.

[2] A. Ferreira and G. Lenzini, "Can Transparency Enhancing Tools support patients accessing Electronic Health Records?" in *Proc. of the 3rd World Conference on Information Systems and Technologies, to be held at Ponta Delgada, So Miguel, Azores, Portugal, 1 - 3 April 2015*, 2015, (to appear).

[3] M. Janic, J. Wijbenga, and T. Veugen, "Transparency Enhancing Tools (TETs): An Overview," in *Socio-Technical Aspects in Security and Trust (STAST), 2013 Third Workshop on*, June 2013, pp. 18–25.

[4] C. A. Cassa, R. A. Miller, and K. D. Mandl, "A novel, privacy-preserving cryptographic approach for sharing sequencing data." *JAMIA*, vol. 20, no. 1, pp. 69–76, 2013.

[5] F. E.-Z. A. Elgamal, N. A. Hikal, and F. E. Z. Abou-Chadi, "Secure medical images sharing over cloud computing environment," *International Journal of Advanced Computer Science and Applications(IJACSA)*, vol. 4, no. 5, 2013.

[6] S. H. Han, M. H. Lee, S. G. Kim, J. Y. Jeong, B. N. Lee, M. S. Choi, I. K. Kim, W. S. Park, K. Ha, E. Cho, Y. Kim, and J. B. Bae, "Implementation of Medical Information Exchange System Based on EHR Standard," *Healthcare informatics research*, 2010.

[7] R. Kettimuthu, R. Schuler, D. Keator, M. Feller, D. Wei, M. Link, J. Bresnahan, L. Liming, J. Ames, A. Chervenak, I. Foster, and C. Kesselman, "A data management framework for distributed biomedical research environments," in *Proceedings of the 2010 Sixth IEEE International Conference on e-Science Workshops*, 2010, pp. 72–79.

[8] P. Rewagad and Y. Pawar, "Use of digital signature and rijndael encryption algorithm to enhanced security of data in cloud computing services," *IJCA Proceedings on Emerging Trends in Computer Science and Information Technology*, no. 2, pp. 5–7, April 2012.

[9] H. Satoh, N. Niki, K. Eguchi, H. Ohmatsu, M. Kusumoto, M. Kaneko, R. Kakinuma, and N. Moriyama, "Teleradiology network system using the web medical image conference system with a new information security solution," in *Proc. SPIE*, 2013.

[10] D. Thilakanathan, S. Chen, S. Nepal, R. Calvo, and L. Alem, "A platform for secure monitoring and sharing of generic health data in the cloud," *Future Gener. Comput. Syst.*, pp. 102–113, Jun. 2014.

[11] F. Al-Nayadi and J. H. Abawajy, "An authorization policy management framework for dynamic medical data sharing," in *The International Conference on Intelligent Pervasive Computing*, Oct 2007, pp. 313–318.

[12] R. Basavegowda and S. Seenappa, "Electronic medical report security using visual secret sharing scheme," in *15th International Conference on Computer Modelling and Simulation*, April 2013, pp. 78–83.

[13] F. Al-Nayadi and J. H. Abawajy, "An authentication framework for e-health systems," in *IEEE International Symposium on Signal Processing and Information Technology*, Dec 2007, pp. 616–620.

[14] W. K. Seng, R. Besar, and F. Abas, "Collaborative support for medical data mining in telemedicine," in *2nd Information and Communication Technologies*, vol. 1, 2006, pp. 1894–1899.

[15] K. Chida, G. Morohashi, H. Fuji, F. Magata, A. Fujimura, K. Hamada, D. Ikarashi, and R. Yamamoto, "Implementation and evaluation of an efficient secure computation system using 'r' for healthcare statistics," *Journal of the American Medical Informatics Association*, vol. 21, pp. e326–e331, 2014.

[16] T. Ermakova and B. Fabian, "Secret sharing for health data in multi-provider clouds," in *IEEE 15th Conference on Business Informatics*, July 2013, pp. 93–100.

[17] M. A. Hajjaji, S. Ajili, A. Mtibaa, and E.-B. Bourennane, "A new system for watermarking based on the turbo-codes and wavelet 5/3," in *13th International conference on Sciences and Techniques of Automatic control & computer engineering*, Tunisia, Dec. 2012.

[18] M. A. Hajjaji, A. Mtibaa, and E. bey Bourennane, "A watermarking of medical image: New approach based on "multi-layer " method," 2011.

[19] S. Hameed, H. Yuchoh, and W. Al-Khateeb, "A model for ensuring data confidentiality: In healthcare and medical emergency," in *4th International Conference On Mechatronicsw*, May 2011, pp. 1–5.

[20] A. Hossain, S. Ferdous, S. Islam, and N. Maalouf, "Rapid cloud data processing with healthcare information protection," in *IEEE World Congress on Services*, June 2014, pp. 454–455.

[21] W. Lee, S. Kim, M. Noh, and H. Kim, "A virtualized network model for wellness information technology research," in *International Conference on IT Convergence and Security*, Dec 2013, pp. 1–3.

[22] J. Liu, X. Huang, and J. K. Liu, "Secure sharing of personal health records in cloud computing: Ciphertext-policy attribute-based signcryption," *Future Generation Computer Systems*, 2014.

[23] S. N. Bharti Ratan Madnani, "Attribute based encryption for scalable and secure sharing of medical records in cloud computing design and implementation," 2013.

[24] M. Mohanty, P. Atrey, and W. T. Ooi, "Secure cloud-based medical data visualization," in *Proceedings of the 20th ACM International Conference on Multimedia*, 2012, pp. 1105–1108.

[25] R. Neame, "Effective sharing of health records, maintaining privacy: a practical schema," 2013.

[26] I. Nwankwo, S. Hanold, and N. Forgo, "Legal and ethical issues in integrating and sharing databases for translational medical research within the eu," in *IEEE 12th International Conference on Bioinformatics Bioengineering*, Nov 2012, pp. 428–433.

[27] L. Seitz, J. M. Pierson, and L. Brunie, "Encrypted storage of medical data on a grid," in *Methods Inf Med*, 2005, pp. 198–201.

[28] Y. Tian, H. Lei, L. Wang, K. Zeng, and T. Fukushima, "A fast search method for encrypted medical data," in *IEEE International Conference on Communications Workshops*, June 2009, pp. 1–5.

[29] P. M. Vieira-Marques, R. J. Cruz-Correia, S. Robles, J. Cucurull, G. Navarro, and R. Marti, "Secure integration of distributed medical

data using mobile agents," *IEEE Intelligent Systems*, no. 6, pp. 47–54, Nov 2006.

[30] P. de Vlieger, J. Y. Boire, V. Breton, Y. Legre, D. Manset, J. Revillard, D. Sarramia, and L. Maigne, "Sentinel e-health network on grid: developments and challenges," *Stud Health Technolol Inform*, vol. 159, 2010.

[31] S. A. K. Mostafa, N. El-Sheimy, A. S. Tolba, F. M. Abdelkader, and H. M. Elhindy, "Wavelet packets-based blind watermarking for medical image management," *The open biomedical engineering journal*, vol. 4, pp. 93–98, 2010.

[32] G. Coatrieux, C. Quantin, F.-A. Allaert, B. Auverlot, and C. Roux, "Watermarking - a new way to bring evidence in case of telemedicine litigation," 2011.

[33] V. Goudar and M. Potkonjak, "A robust watermarking technique for secure sharing of basn generated medical data," in *IEEE International Conference on Distributed Computing in Sensor Systems*, May 2014, pp. 162–170.

[34] H. B. Rahmouni, T. Solomonides, M. C. Mont, and S. Shiu, "Privacy aware access controls for medical data disclosure on european healthgrids," 2010.

[35] G. Haddow, A. Bruce, S. Sathanandam, and J. C. Wyatt, "nothing is really safe': a focus group study on the processes of anonymizing and sharing of health data for research purposes," *Journal of Evaluation in Clinical Practice*, vol. 17, no. 6, pp. 1140–1146, 2011.

[36] K. K. Kim, D. McGraw, L. Mamo, and L. Ohno-Machado, "Development of a privacy and security policy framework for a multistate comparative effectiveness research network," 2013.

[37] H. Lambert and C. F. Leonhardt, "Federated authentication to support information sharing: Shibboleth in a bio-surveillance information grid," *Proceedings of the 18th International Congress and Exhibition*, vol. 1268, no. 0, pp. 135–140, 2004.

[38] S. Lohiya and L. Ragha, "Privacy preserving in data mining using hybrid approach," in *Fourth International Conference on Computational Intelligence and Communication Networks*, Nov 2012, pp. 743–746.

[39] T. Neubauer and J. Heurix, "A methodology for the pseudonymization of medical data," *International Journal of Medical Informatics*, pp. 190–204, 2011.

[40] C. Quantin, M. Fassa, E. Benzenine, D.-O. Jaquet-Chiffelle, G. Coatrieux, and F.-A. Allaert, "The mixed management of patients' medical records: responsibility sharing between the patient and the physician," *Studies in health technology and informatics*, vol. 156, p. 189200, 2010.

[41] H. B. Rahmouni, T. Solomonides, M. C. Mont, and S. Shiu, "Privacy compliance and enforcement on european healthgrids: an approach through ontology," *Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences*, vol. 368, no. 1926, pp. 4057–4072, 2010.

[42] P. Ruotsalainen, B. Blobel, P. Nyknen, A. Seppl, and H. Sorvari, "Framework model and principles for trusted information sharing in pervasive health," 2011.

[43] M. Jafari, R. Safavi-Naini, C. Saunders, and N. P. Sheppard, "Using digital rights management for securing data in a medical research environment," in *Proceedings of the Tenth Annual ACM Workshop on Digital Rights Management*, 2010, pp. 55–60.

[44] A. Solanas, A. Martinez-Balleste, and J. Mateo-Sanz, "Distributed architecture with double-phase microaggregation for the private sharing of biomedical data in mobile health," *IEEE Transactions on Information Forensics and Security*, vol. 8, no. 6, pp. 901–910, June 2013.

[45] D. Weerasinghe and R. Muttukrishnan, "Secure trust delegation for sharing patient medical records in a mobile environment," in *7th International Conference on Wireless Communications, Networking and Mobile Computing*, Sept 2011, pp. 1–4.

[46] P. K. Katarzyna Pasierb, Tomasz Kajdanowicz, "Privacy-preserving data mining, sharing and publishing," 2013.

[47] R. Gajanayake, R. Iannella, and T. Sahama, "Sharing with care: An information accountability perspective," *IEEE Internet Computing*, no. 4, pp. 31–38, July 2011.

[48] T. Tashiro, S. Date, S. Takeda, I. Hasegawa, and S. Shimojo, "Practice and experience of building a medical application with permis-based access control mechanism," in *The Sixth IEEE International Conference on Computer and Information Technology*, Sept 2006, pp. 71–71.

[49] A. Gaignard and J. Montagnat, "A distributed security policy for neuroradiology data sharing," *Stud Health Technol Inform.*, vol. 147, pp. 257–262, 2009.

[50] T. Tashiro, S. Date, S. Takeda, I. Hasegawa, and S. Shimojo, "Architecture of authorization mechanism for medical data sharing on the grid," *Studies in health technology and informatics*, vol. 120, p. 358367, 2006.

[51] Y. feng Jiang, S. yue Zhang, Z. Huang, M. qing Liu, L. Yin, and J. ping Niu, "Access control for rural medical and health collaborative working platform," *The Journal of China Universities of Posts and Telecommunications*, no. 0, pp. 7 –10, 2013.

[52] S. Langella, S. Hastings, S. Oster, T. Pan, A. Sharma, J. Permar, D. Ervin, B. B. Cambazoglu, T. M. Kurç, and J. H. Saltz, "Sharing data and analytical resources securely in a biomedical research grid environment," *JAMIA*, vol. 15, no. 3, pp. 363–373, 2008.

[53] J. Stevovic, F. Casati, B. Farraj, J. Li, H. Motahari-Nezhad, and G. Armellin, "Compliance aware cross-organization medical record sharing," in *IFIP/IEEE International Symposium on Integrated Network Management*, May 2013, pp. 772–775.

[54] M. Kavitha and T. K. Anjana, "Password authentication scheme based on shape and text for secure sharing of phr using abe in cloud," 2013.

[55] A. Ferreira, G. Lenzini, C. Santos-Pereira, A. B. Augusto, and M. E. Correia, "Envisioning secure and usable access control for patients," in *IEEE 3rd International Conference on Serious Games and Applications for Health (SeGAH 2014)*, Rio de Janeiro, Brazil, May 2014.

[56] R. Nithiavathy, "Data integrity and data dynamics with secure storage service in cloud," in *International Conference on Pattern Recognition, Informatics and Mobile Engineering*, Feb 2013, pp. 125–130.

[57] M. A. C. Dekker, "Flexible Access Control for Dynamic Collaborative Environments," Ph.D. dissertation, CTIT PhD.-thesis series ISSN 1381-3617, Number 09-159, IPA Dissertation series, University of Twente, 2009.

A System of Model-Driven Security Design Patterns

Phu H. Nguyen
PhD Candidate, SnT, University of Luxembourg
4 rue Alphonse Weicker, L-2721 Luxembourg. phuhong.nguyen@uni.lu

***Abstract*—*Model-Driven Security* (MDS) emerged more than a decade ago for model-driven development of secure systems. However, a recent systematic review of MDS shows that most current MDS approaches have not extensively dealt with multiple security concerns but rather a specific one, e.g. authorization. Besides, security patterns which are based on domain-independent, time-proven security knowledge and expertise, can be considered as reusable security bricks upon which sound and secure systems can be built. But security patterns are not applied properly as they could be because developers have problems in selecting them and applying them in the right places, especially at the design phase. In this paper, we propose a MDS approach based on a *System of Security design Patterns* (SoSPa) in which security design patterns are collected, specified as reusable aspect models (RAMs) to form a coherent system of them that guides developers in systematically selecting and applying the right security design patterns for the job. Specifically, SoSPa consists of not only a catalog of security design patterns dealing with multiple security concerns, but also inter-pattern relations. The interrelationships specified at conceptual level using an extended feature model can then be refined at the detailed design level using RAMs.**

I. Problem and Motivation

Sound methodologies for secure systems development are seriously needed to deal with continuously evolving security threats and increasingly complex IT systems. Security must not be an afterthought but systematically engineered into the systems. Model-Driven Engineering (MDE) is considered by some researcher [2] as a solution to the handling of complex, evolving systems. As a specialization of MDE, Model-Driven Security (MDS) takes security-oriented models into focus from very beginning, and through out every stage of the development cycle.

Current state of the art of MDS [6] reveals that there is a lack of approaches dealing with multiple security concerns at the same time. Most current MDS approaches have not extensively dealt with multiple security concerns but rather solely one, e.g. authorization (especially, access control). Besides, security patterns which are based on domain-independent, time-proven security knowledge and expertise, can be considered as *reusable* security bricks upon which sound and secure systems can be built. According to Schumacher et al. [8], a *security pattern* describes a particular recurring security problem that arises in specific contexts and presents a well-proven generic scheme for its solution. But security patterns have not been applied as efficiently as they could be because developers have problems in selecting them and applying them. Indeed, security patterns could be applied at different levels of abstraction, e.g. architectural design rather than detailed design. Moreover, the levels of quality found in security patterns are varied, not equally well-defined like software design patterns [4]. Particularly, many security patterns are too abstract or general, without a well-defined, solution-oriented description. There is also a lack of coherent specification of interrelationships among security patterns, and with other quality attributes like performance, usability.

A system of security design patterns is a collection of patterns for designing secure systems, together with guidelines for their implementation, combination, and practical use in secure systems development. In this paper, we propose a MDS approach based on a *System of Security design Patterns* (SoSPa) in which security design patterns are collected, specified as reusable aspect models (RAM) [5] to form a coherent system of them. RAM is an aspect-oriented multi-view modeling approach with tool support for aspect-oriented design of complex systems. In RAM, any concern or functionality that is *reusable* can be modeled in an aspect model. The main contribution of this paper is a MDS framework to bridge the gap between the security patterns and their application. To support for the application of security patterns, a well-defined refinement process for security patterns is needed. In such a refinement process, security patterns can be defined firstly at an abstract level and then eventually can be extended towards more well-defined, solution-oriented security design patterns. The refinement process should also take into account the inter-pattern relation, and with other quality attributes like performance, usability, etc. The refinement process allows abstract security patterns to become more formal that enables a MDS framework for the productivity and quality in secure systems development.

II. Approach

Here we present our MDS approach based on a *System of Security design Patterns* (SoSPa) [7] which is generic and extensible to address multiple security concerns, including the interrelations among them.

Our SoSPa is inspired by [1] which shows an approach based on RAM for designing software with concern as the main unit of reuse. Here, we specifically target security with a *System of Security design Patterns* which can be specified using RAM. Roughly speaking, a developer could use SoSPa as an extensible library like a programmer would reuse generic classes in programming libraries. More than that, SoSPa and our MDS framework based on it provide means for systematically selecting the right security design patterns for the right job. Technically, the system of security design patterns consists of an extensible set of security concerns (e.g. authentication,

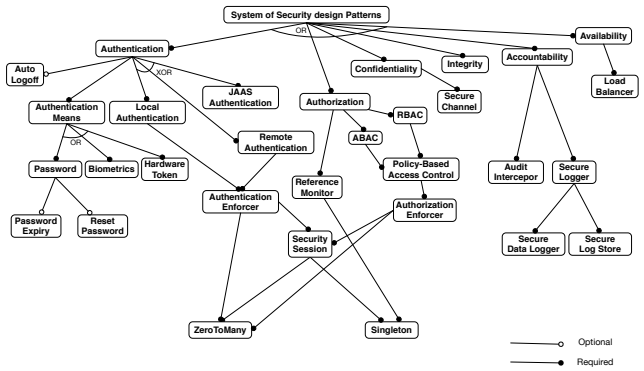


Fig. 1. SoSPa - A System of Security design Patterns

authorization, encryption, etc.) which can fulfill the security objectives (e.g., confidentiality, integrity, availability, privacy). Each security concern is composed of a set of aspect-oriented security design patterns that realizes the security concern. The meta-info about interrelationships among security design patterns are well specified within the system of them, i.e. by using an extended feature model (Fig. 1). For example, some security design patterns could complement one another, or exclude each other. Moreover, each security design pattern also contains other meta-info describing the side effects of its adoption on other non-functional quality criteria, e.g. performance, usability, etc. All these meta-info are useful for analysis of the trade-off among alternatives which leads to a thoughtful decision on systematically selecting the right security design patterns for the job. The process of selecting and composing security design patterns into the target system design consists of the following main steps:

1. Identify the security-critical assets of the target system with the priorities to have them.
2. For each asset, determine security concerns for the corresponding asset type. For each security concern, describe the context and the security problem that need to be solved.
3. Identify the possible attacks/threats for the security-critical assets by consulting well-known sources, e.g. the OWASP top 10 most critical web application security risks.
4. For each security concern, use the feature model of the security concern to select the most appropriate security design pattern, i.e., the pattern that best matches with the context and the security problem, most satisfies the interrelationships with the other already selected security design patterns, and maximizes the positive impact on relevant non-functional quality criterion like usability, performance, etc. This step derives the detailed design for the selected security pattern, including its *customization interface* and *usage interface*. The *customization interface* of a RAM model consists of so-called *mandatory instantiation parameters* that must be instantiated in order for the model to be used within a specific application. The *usage interface* of a RAM model is comprised of all the *public* model elements, e.g. *public* class properties like attributes and operations. More details about these two kinds of interface of a RAM model can be found in [1].
5. For each selected security pattern, use the customization interface of the generated design to adapt the generic design elements to the application-specific context. This step generates the mappings of the parameterized elements in the security design pattern with the target elements in the target system design. Any constraints between mappings of all the selected security design patterns need to be resolved.
6. Automatically weave all the selected security design patterns into the target system design. The mappings from previous step are the input for the weaving process in this step.
7. Analyze the woven secure system against the attack models obtained from step 3. Depending on each security pattern, the attack models can be used for formal verification of security properties in the woven model, or can be used for test cases generation in a security testing approach.

III. Conclusions and Future Work

In this paper, we have proposed a MDS approach based on a *System of Security design Patterns* (SoSPa) to guide the model-driven application of security patterns in secure systems development. In our approach, security design patterns are collected, specified as reusable aspect models to form a coherent system of them that allows developers to systematically select and apply the right security design patterns for the job. Not only security patterns but also the inter-pattern relations are specified in SoSPa.

We are validating our ideas by applying SoSPa to a case study, using patterns targeting three main security concerns, i.e. authentication, authorization, confidentiality, plus inter-pattern relations. Future work can be dedicated for extending the approach in [1] for specifying the constraints of security patterns with other quality attributes like performance, usability, etc. Moreover, the steps of risk analyses, and formal verification and validation of security properties still remain for future investigation.

References

[1] O. Alam, J. Kienzle, and G. Mussbacher. Concern-oriented software design. In *MODELS*. 2013.
[2] J. Bézivin. Model driven engineering: An emerging technical space. In *Generative and transformational techniques in software engineering*, pages 36–64. Springer, 2006.
[3] G. Georg, I. Ray, K. Anastasakis, B. Bordbar, M. Toahchoodee, and S. H. Houmb. An aspect-oriented methodology for designing secure applications. *Information and Software Technology (IST)*, 2009.
[4] T. Heyman, K. Yskout, R. Scandariato, and W. Joosen. An analysis of the security patterns landscape. In *Proceedings of SESS '07*, SESS '07, 2007.
[5] J. Kienzle, W. Al Abed, F. Fleurey, J.-M. Jzquel, and J. Klein. Aspect-oriented design with reusable aspect models. In *TAOSD VII*. 2010.
[6] P. H. Nguyen, J. Klein, M. Kramer, and Y. Le Traon. A Systematic Review of Model Driven Security. In *Proceedings of the 20th APSEC*, 2013.
[7] P. H. Nguyen, J. Klein, and Y. Le Traon. Model-driven security with a system of aspect-oriented security design patterns. In *Proceedings of the 2nd Workshop on View-Based, Aspect-Oriented and Orthographic Software Modelling*, page 51. ACM, 2014.
[8] M. Schumacher, E. Fernandez, D. Hybertson, and F. Buschmann. *Security Patterns: Integrating Security and Systems Engineering*. John Wiley & Sons, 2005.
[9] K. Yskout, T. Heyman, R. Scandariato, and W. Joosen. A system of security patterns, 2006.