POSTER–TRIPLEX: Verifying Data Minimisation in Communication Systems

Meilof Veeningen, Mayla Brusò, Jerry den Hartog, and Nicola Zannone
Department of Mathematics and Computer Science, Eindhoven University of Technology
PO Box 513, 5600 MB Eindhoven, The Netherlands
{m.veeningen,m.bruso,j.d.hartog,n.zannone}@tue.nl

ABSTRACT

Systems dealing with personal information are legally required to satisfy the principle of data minimisation. Privacy-enhancing protocols use cryptographic primitives to minimise the amount of personal information exposed by communication. However, the complexity of these primitives and their interplay makes it hard for non-cryptography experts to understand the privacy implications of their use. In this demo, we present TRIPLEX, a framework for the analysis of data minimisation in privacy-enhancing protocols.

Categories and Subject Descriptors

C.2.2 [**Computer-Communication Networks**]: Network Protocols—*Protocol verification*; K.4.1 [**Computers and Society**]: Public Policy Issues—*Privacy*

Keywords

Data minimisation; Coalition Graphs; Detectability; Linkability

1. INTRODUCTION

Communication systems such as identity management systems, social networks, and e-health systems deal with ever increasing amounts of personal information. EU privacy laws require such systems to satisfy the "data minimisation" principle. That is, systems have to be designed to ensure that actors within such systems collect and store only the minimal amount of personal information needed to fulfil their task. (Note that data minimisation concerns the knowledge of insiders when the system operates normally rather than the knowledge of attackers who attempt to disrupt its operation.) An important factor in achieving data minimisation is the use of *privacy-enhancing* communication protocols which employ cryptographic primitives to ensure minimal leakage of information. In particular, they ensure that participants learn as little as possible, even when they try to combine information from different sources to build profiles. Such protocols have been proposed in various domains, including toll pricing, e-voting, e-health and identity management.

CCS'13, Nov 04–08 2013, Berlin, Germany
ACM 978-1-4503-2477-9/13/11.
http://dx.doi.org/10.1145/2508859.2512514.

Understanding exactly what privacy guarantees different protocols offer is important, e.g., for system designers that want to use privacy-enhancing protocols, or for system administrators that want to select what system to use. However, it is typically not straightforward for non-cryptography experts to obtain such an understanding. One reason is that privacy-enhancing protocols typically combine (advanced) cryptographic primitives in subtle ways; also, typical scenarios involve multiple actors which may collude to build comprehensive user profiles. Although a number of privacy analyses of such protocols have been proposed [1, 4], they are typically performed manually in an informal and high-level (and thus, possibly subjective) way. Tools for (semi-)automated analyses using formal methods are increasingly applied [3, 5], but several issues inhibit broader practical use. First, they require considerable manual work for each particular property to be verified, often needing strong assumptions to make the computation feasible. Second, property definitions are usually specific to particular protocols, making it difficult to compare different protocols; finally, results are not presented intuitively, requiring substantial manual review.

In this paper, we present TRIPLEX, a tool-supported framework that provides high-level but precise formal analysis of data minimisation to non-cryptography experts. Using TRIPLEX, users can visually construct scenarios of different actors communicating using any kind of privacy-enhancing protocol. TRIPLEX automatically simulates these scenarios, and provides different analysis tools. These tools, geared towards non-cryptography-experts, allow users to analyse the knowledge actors learn by executing communication protocols as well as to verify protocol-independent privacy properties. We demonstrate TRIPLEX in an e-health scenario.

2. THE TRIPLEX FRAMEWORK

In this section, we present the TRIPLEX framework.

2.1 Framework Ingredients

The main idea behind the TRIPLEX framework is to analyse relevant privacy aspects of privacy-enhancing protocols in a specified scenario that may involve several actors and protocol instances (of different protocols). Intuitively, we annotate the messages that are exchanged in the scenario to express to whom information refers; we then use formal tools to determine who can detect and link together privacy-sensitive information. Our framework is based on the following main technical ingredients:

- *Three-Layer Model of Information* – We use the three-layer model of knowledge about personal information from [6]. This model combines abstract and concrete descriptions of information. Namely, communication protocols and their desired privacy properties are described abstractly, i.e., independently from the actual information transmitted. However,

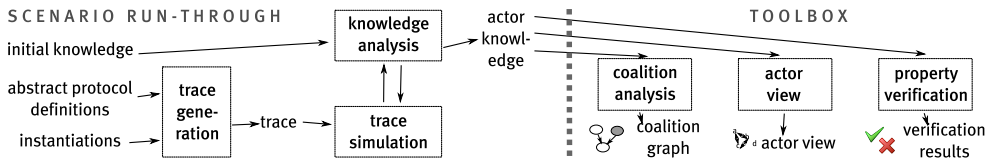


Figure 1: The TRIPLEX framework: the scenario run-through (left) and privacy analysis toolbox (right).

whether these properties are satisfied in a particular situation depends on what concrete information is used and on their contents. The model of [6] describes abstract and concrete information at three "layers", allowing protocols and privacy properties to be specified at a high level yet verified using the contents of particular pieces of information.

- *Dolev-Yao Model of Cryptography* – We employ a standard Dolev-Yao black-box model to capture the functionality of cryptographic primitives and operations used in communication protocols. In particular, we use the deductive system from [7], consisting of construction, elimination, and testing rules, to reason about messages in the three-layer model. TRIPLEX provides support for several standard (e.g., symmetric and asymmetric encryption, digital signatures, hashes) and advanced primitives (e.g., zero-knowledge proofs, anonymous credentials), and allows extension.

- *Coalition Graphs* – Actors may collude to build more comprehensive user profiles. Coalition graphs have been proposed in [8] as a way to visually represent the knowledge of personal information about a particular data subject in a scenario. They can be derived automatically from a three-layer representation of knowledge. Intuitively, nodes in a coalition graph represent profiles of personal information about a user that a coalition can make, whereas edges represent ways for a coalition to increase its knowledge by including additional actors. Coalition graphs can be used to analyse actors' knowledge as well as to visually compare different protocols, or to compare a protocol against a predefined "privacy-optimal" situation [8].

2.2 Framework Overview

The TRIPLEX framework is shown in Figure 1. The framework consists of a *scenario run-through* component which simulates a scenario, and a *toolbox* consisting of various tools with which actors' knowledge of personal information can be analysed.

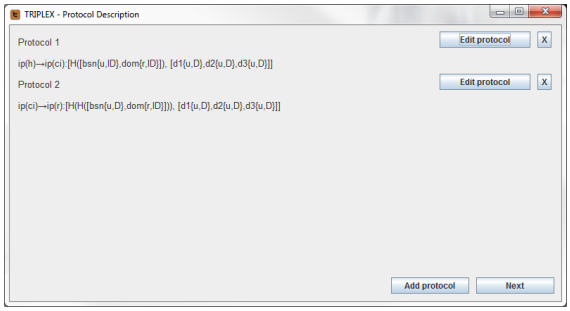
The scenario run-through simulates a user-specified scenario in which various actors exchange personal information using privacy-enhancing communication protocols. As input, this component requires a description of the scenario. This description consists of an abstract description of the protocols used (*abstract protocol definitions*), the characteristics of the information used in the protocol instances (*instantiations*), and actors' *initial knowledge*. From the abstract protocol definitions and instantiations, the component generates a *trace* representing the correct protocol execution in the three-layer model. The component then runs through the scenario by iteratively determining the knowledge of the actors at one point in time using the algorithm from [6], and using it to simulate the next step of the trace, which again changes their knowledge. Finally, the *actor knowledge* after the protocol execution is obtained. The scenario run-through component is implemented in Prolog. Optionally, knowledge analysis supports equational models of primitives by interfacing the KISS tool for static equivalence [2]. Although this extension increases the range of primitives that can be considered, it comes at the cost of decreased performance.

After the scenario has been simulated and the final actor knowledge is determined, the toolbox can be used to analyse this knowledge. The toolbox currently consists of three tools: *coalition analysis*, *actor view*, and *property verification*. The *coalition analysis* tool automatically computes and visualises the coalition graph describing the knowledge about one particular data subject held by the (coalitions of) actors within the system. This coalition graph can be manually checked for unexpected user profiles. In addition, the tool can visualise coalition graphs comparing knowledge to a predefined "optimal knowledge" [8], allowing users to directly identify what knowledge of personal information is avoidable. The *actor view* tool computes the knowledge that (coalitions of) actors have about the users within the system. The tool also takes into account exactly where that knowledge comes from (i.e., which particular protocol instance). Thus, this low-level view of actor's knowledge can help to explain the unexpected user profiles found using coalition analysis. Finally, the *property verification* tool enables the automatic verification of privacy properties in the scenario. Rather than considering the knowledge of all coalitions of actors as before, privacy properties allow a more focused analysis of one particular aspect of knowledge (e.g., whether a coalition of actors X and Y can combine their different views of data subject Z). Any property expressible in terms of characteristics of actor views can be verified, including detectability properties (does a particular actor/coalition learn a particular piece of information), linkability properties (can a particular actor/coalition combine different views of a data subject), and involvement properties (does an actor/coalition know that a particular other actor was involved in a transaction), or arbitrary combinations of these properties.

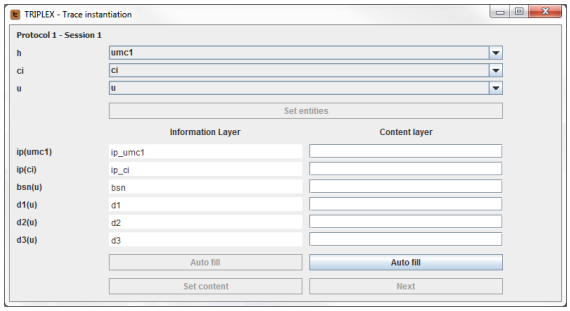
2.3 Demonstration

We demonstrate TRIPLEX by analysing data minimisation in Parelsnoer [8], a system for distributing patient data for medical research. In Parelsnoer, data from different hospitals is collected by a *central infrastructure* (CI). By applying cryptographic hashes, the hospitals prevent the CI from learning the patient identifiers, while the CI is still able to link patient data from different hospitals together. Upon request from a medical researcher, the CI compiles a dataset from the patient data for use in a particular research project, again using cryptographic hashes to ensure that data from different research projects cannot be linked to the patient or to each other. These two different steps (hospital to CI, CI to researcher) are modelled as two different abstract protocols. In particular, they are represented in TRIPLEX as sequences of messages (Figure 2a): we indicate the format of the messages and the type of information that they contain. For instance, the first message of the first protocol is a cryptographic hash of the patient's identifier, followed by a sequence of patient data items; the first message of the second protocol is a cryptographic hash of the BSN (i.e., the Dutch social security number) and a "domain" identifying the research project, again followed by a sequence of patient data items.

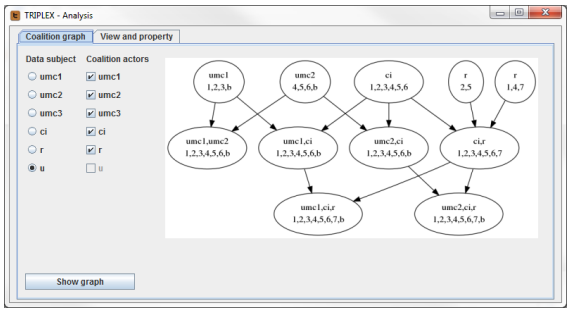
Next, we define the scenario in which we want to analyse the privacy consequences of using these protocols (Figure 2b). In our scenario, two hospitals *umc1*, *umc2* provide data about patient *p* to the CI *ci*; a third hospital *umc3* does not know the patient. Hence,



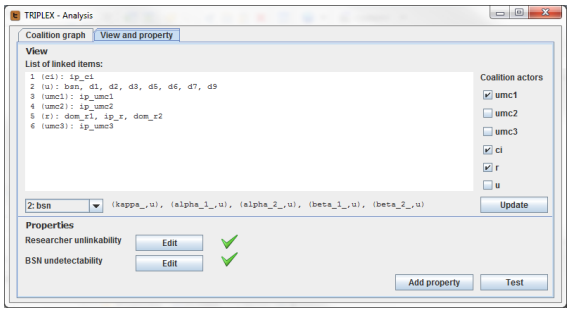
(a) Protocol descriptions



(b) Instantiating personal information



(c) Coalition analysis using coalition graph



(d) Coalition view and property verification

Figure 2: The TRIPLEX framework: the scenario run-through (left), and privacy analysis toolbox (right).

we have two instantiations of the "hospital to CI"-protocol: one involving *umc1* and *ci* as communication actors and *p* as data subject, and similarly, one involving *umc2*, *ci*, and *p*. Next, researcher *r* is involved in two different research projects, and he learns data about *p* in datasets for both projects. The distribution of the dataset is modelled by two instances of the "CI to researcher" protocol. Having defined the actors and instances, we specify what information is exchanged in the scenario. Here, *umc1* and *umc2* both store three pieces of patient data, which they all provide to the CI. The researcher obtains two disjoint datasets containing patient data from both hospitals, and gathers additional data as part of his research.

After TRIPLEX has simulated the scenario, the computed knowledge of (coalitions of) actors can be analysed in different ways. First, we look at the coalition graph that shows the knowledge of data about *p* of all actors *umc1*, *umc2*, *umc3*, *r*, and *ci*. Figure 2c shows that there are two coalitions of actors that know all information in the scenario, namely the CI, researcher, and either of the hospitals *umc1* and *umc2*. In particular, the knowledge of the coalition *umc1*,*ci*,*r* is composed of the patient database of *umc1*, the research data of the researcher, and the data distributed via the four protocol instances, in all of which *ci* was involved (Figure 2d). Furthermore, we define and verify two privacy properties: one focusing on the knowledge of the researcher, and one focusing on the knowledge of various actors on a particular piece of information. TRIPLEX reports that both properties are satisfied (Figure 2d).

3. CONCLUSIONS

We have presented TRIPLEX, a framework for the analysis of data minimisation in privacy-enhancing protocols. Based on a solid formal grounding, the framework gives an objective and precise overview of what personal information is learnt by (coalitions of) actors in a user-specified scenario. Our user interface allows non-cryptography experts to model protocols and visually analyse their privacy at a high level, while also giving expert users the option to model new cryptographic primitives and to directly access lower-level formal details. We have demonstrated TRIPLEX using a case study in the healthcare domain. The framework has also been successfully applied in the identity management domain [7].

Acknowledgements This work is funded by the Dutch Sentinel Mobile IDM project (#10522), EIT ICT Lab, and MEALS (#295261).

4. REFERENCES

[1] Identity Management Systems (IMS): Identification and Comparison Study. Independent Centre for Privacy Protection Schleswig-Holstein, 2003.

[2] KISS (Knowledge in Security ProtocolS). http://www.lsv.ens-cachan.fr/∼ciobaca/kiss/, 2009.

[3] B. Blanchet, M. Abadi, and C. Fournet. Automated Verification of Selected Equivalences for Security Protocols. *J. Log. Algebr. Program.*, 75(1):3–51, 2008.

[4] J. H. Hoepman, R. Joosten, and J. Siljee. Comparing Identity Management Frameworks in a Business Context. In *Proc. IFIP/FIDIS Summer School*, pages 184–196. Springer, 2008.

[5] C. Meadows. Formal Methods for Cryptographic Protocol Analysis: Emerging Issues and Trends. *IEEE Sel. Areas Commun.*, 21(1):44–54, 2003.

[6] M. Veeningen, B. de Weger, and N. Zannone. Formal Privacy Analysis of Communication Protocols for Identity Management. In *Proc. of Int. Conf. on Information Systems Security*, LNCS 7093, pages 235–249. Springer, 2011.

[7] M. Veeningen, B. de Weger, and N. Zannone. A Formal Privacy Analysis of Identity Management Systems. arXiv 1206.7111v1, ArXiv.org, 2012.

[8] M. Veeningen, B. de Weger, and N. Zannone. Formal Modelling of (De)Pseudonymisation: A Case Study in Health Care Privacy. In *Proc. of Workshop on Security and Trust Management*, LNCS 7783, pages 145–160. Springer, 2012.