Using a Security Requirements Engineering Methodology in Practice: the compliance with the Italian Data Protection Legislation

Fabio Massacci[a], Marco Prest[b] and Nicola Zannone[a]

[a]Dip. di Informatica e Telecomunicazioni, University of Trento, Italy

[b]Direzione Amministrativa IT, University of Trento, Italy

Extending Requirements Engineering modelling and formal analysis methodologies to cope with Security Requirements has been a major effort in the past decade. Yet, only few works describe complex case studies that show the ability of the informal and formal approaches to cope with the level complexity required by compliance with ISO-17799 security management requirements.

In this paper we present a comprehensive case study of the application of the Secure Tropos RE methodology for the compliance to the Italian legislation on Privacy and Data Protection by the University of Trento, leading to the definition and analysis of a ISO-17799-like security management scheme.

Keywords: Security requirements engineering, Information security management system; Standard for privacy protection.

1. Introduction

The last years have seen a major interest in the development of requirements engineering (RE) methodologies which are able to capture security requirements. This has been marked by some workshops (SREIS, SAPS, REHAS, et al.) and many papers and books [1–6].

Some works have focused on modelling security and privacy concepts within existing RE frameworks. For example Liu et al. [2] have used Tropos/i*, while Antòn et al. [1] have proposed a taxonomy of privacy requirements based on a goal oriented methodology. Others have modified the RE constructs to account for special constructs for privacy & security. The most notable proposal is Jürjens's UMLsec [6] where security tags are added to UML constructs. Sindre and Opdahl [4] define the concept of a misuse case, the inverse of a use case, which describes a function that the system should not allow. An analogous proposal has been put forward by van Lamsweerde et al. [5] that introduce the notion of anti-goals, i.e., goals of the attacker that can be refined. Giorgini et al. [3] present a framework extending Tropos in which security is considered during the whole process of requirements analysis, and trust and delegation relationships are used to model the interactions among actors involved in the system. Many of those proposals are backed up by a number of formal analysis tools. For sake of example Jurien's work [6] is based on the AutoFOCUS case tool, van Lamsweerde's approach is based on the KAOS, modal logic based, reasoning tool [5], and Giorgini et al. work is based on Datalog [3].

Yet, what seems missing is the proof-of-concept ability to support the enterprise in the definition of complex security policies as dictated by ISO security standards (e.g. ISO-17799 [7]) or complex national Data Protection Legislation. Indeed, it should be possible to use the RE methodology to derive the policy itself using its refinement mechanism and verify and validate the same policy using the analysis tools available with the framework. In contrast, many papers presents the methodology and supply some (toy) examples but only a handful describe complex case studies [8,9] which really copes with the complexity required by ISO-17799 compliance.

In this paper we present a major case study of the application of the Secure Tropos requirements engineering modelling and formal analysis

methodology [3,10] for the compliance to the Italian legislation on Privacy and Data Protection by the University of Trento. Due to lack of space, we focus on the key modelling aspects of the case study and refer to [3] for the introduction of the general formal framework based on Datalog.

For lack of space additional details are shown in the technical report. In the next section we briefly sketch the Italian and EU Data Protection Legislation and its requirements and the information about the University of Trento that is relevant to the law (§2). Then we present the Secure Tropos RE methodology (§3) and we dig into the details of the case study showing some examples of modelling actors (§4), modelling dependency and delegation (§5), and refining one's specification (§6). Finally we point out to a number of issues that have been discovered by the analysis (§7), discuss related case studies and conclude (§8).

2. The Italian Data Protection Legislation

Many countries have recently promulgated a new privacy legislation spurred by increased concerns over data protection. Table 1 gives a brief history of European and Italian legislation about protection of personal data and privacy.

The final EU and Italian legislation systematized the norms on privacy and data protection. It specified:

- the definitions of personal data, sensitive data, and data processing,
- the definitions of all entities involved in data processing, their roles and responsibilities (controller, processor, operator, subject),
- the obligations relating to public and private data controllers with specific reference to the legitimate purpose of data processing and the adoption of minimal precautionary security measures to minimize the risks on data.

The laws set some requirements forced the entire public administration to assess the security of their information systems and imposed the adoption of the implementation of minimal precautionary security measures as authentication and authorization system, antivirus, data backup and restore, and management and risk analysis. The requirements were close but not identical to the ISO standard 17799.

These measures had to be detailed into a "Documento Programmatico sulla Sicurezza" (DPS). Every organization was supposed to draw up, update yearly and obviously deploy a DPS.

The University has enforced the Data Protection Act through a Privacy Internal Regulation on January 14th, 2002 that transposed general regulations into its internal organization.

Williams [11] proposes a maturity model to establish rankings for security in an organization (Table 2). Matched against this scale, the University of Trento can be ranked between 3 and 4. In particular 4(a) is not yet enforced whereas 4(b) and 4(c) are (almost entirely) enforced.

An item-by-item comparison of the DPS and ISO-17799 is shown in the technical report.

3. Security-Aware Tropos

Here we use Security-Enhanced Tropos [3], a variant of Tropos [12], an agent-oriented software development methodology tailored to describe both the organization and the system. We have the concepts of actor, goal, soft goal, task, resource and social relationships for defining the obligations of actors to other actors. Actors have strategic goals and intentions within the system or the organization. A goal represents the strategic interests of an actor. A task specifies a particular course of action that produces a desired effect, and can be executed in order to satisfy a goal. A resource represents a physical or an informational entity. The relationships we have considered so far are functional dependency, ownership, provisioning, trust, and delegation of permission. A functional dependency between two actors means that the dependee will take responsibility for fulfilling the functional goal of a depender. The owner of a service has full authority concerning access and usage of his services, and he can also delegate this authority to other actors. Delegation marks a formal passage between the actors. In contrast, trust marks simply a social relationship that is not formalized by a

Table 1
Brief history of European and Italian data protection legislation

European Legislation
Directive 2002/58/EC on privacy and electronic communications.
Directive 2002/22/EC on universal service and users' rights.
Directive 2002/21/EC on a common regulatory framework for electronic communications networks and services.
Regulation No 45/2001 on the protection of individuals with regard to the processing of personal data.
Directive 2000/31/EC on electronic commerce.
Directive 1997/66/EC on the processing of personal data and the protection of privacy in the telecommunications sector.
Directive 1995/46/EC on the protection of individuals with regard to the processing of personal data.
Italian Legislation
Legislative Decree No 196 of 30 June 2003 Italian Personal Data Protection Code.
Directive of Innovation and Technologies Dep. of 16 January 2002 on computer and telecommunications security in Public Administration.
Legislative Decree No 467 of 28 December 2001 on corrective and additional provisions with regard to personal data protection.
Act No 325 of 3 November 2000 on the adoption of minimum security measures for personal data processing.
Legislative Decree No 281 of 30 July 1999 on personal data processing for historical, statistical and scientific research purposes.
Presidential Decree No 318 of 28 July 1999 Regulation on minimum security measures for personal data processing.
Act No 675 of 31 December 1996 on protection of individuals and other subjects with regard to personal data processing.

Table 2
Maturity of information risk management

Maturity Level	Description
0	**Non-Existent: management processes are not applied at all**
1	**Initial/Ad-Hoc: processes are ad-hoc and disorganized**
2	**Repeatable but intuitive: processes follows a regular pattern**
3	**Defined Process: processes are documented and communicated** (a) An organization-wide risk management policy defines when and how to conduct risk assessments. Risk assessment follows a defined process that is documented and available to all staff; (b) Security awareness exists and is promoted by management through formalized briefings. IT security procedures are defined and fit into a structure for security policies and procedures. Responsibilities for IT security are assigned, but not consistently enforced. An IT security plan exists, driving risk analysis and security solutions. IT security reporting is IT focused, rather than business focused. Ad-hoc intrusion testing is performed. (c) Management communicates consistently the need for continuous service. High-availability components and system redundancy are being applied piecemeal. An inventory of critical systems and components is rigorously maintained.
4	**Managed and Measurable: processes are monitored and measured** (a) The assessment of risk is a standard procedure and exceptions would be noticed by IT management. It is likely that IT risk management is a defined management function with senior level responsibility. Senior management and IT management have determined the levels of risk that the organization will tolerate and have standard measures for risk/return ratios; (b) Responsibilities for IT security are clearly assigned, managed and enforced. IT security risk and impact analysis is consistently performed. Security policies and practices are completed with specific security baselines. Security awareness briefings, user identification, authentication and authorization have become mandatory and standardized. Intrusion testing is standardized and leads to improvements. Cost/benefit analysis, is increasingly used. Security processes are coordinated with the overall organization security function and reporting is linked to business objectives; (c) Responsibilities and standards for continuous service are enforced. System redundancy practices, including use of high-availability components, are being consistently deployed.
5	**Optimized-best practices are followed and automated**

"contract"between the actors: such as a digital credential or a signed piece of paper attributing permission.

Various activities contribute to the acquisition of a first requirement model, to its refinement into subsequent models:

Actor modeling, which consists of identifying and analyzing both the actors of the environment and the system's actors and agents;

Dependency modeling, which consists of identifying actors which depend on one another for goal to be achieved, plans to be performed, and resources to be furnished, and actors which are able to provide goal, plans, and resources.

Trust modeling, which consists of identifying

actors which trust other actors for goal, plans, and resources, and actors which own goal, plans, and resources.

Delegation modeling, which consists of identifying actors which delegate to other actors the permission on goals, plans, and resources.

Goal refinement, which consists of refining requirements and eliciting new relations. This is standard in Goal-Oriented Methodologies [12].

A graphical representation of the model obtained following the first four modeling activities is given through three different kinds of *actor diagrams*: *functional dependency model*, *trust model*, and *trust management implementation*. In these diagrams, actors are represented as circles; goals, tasks and resources are respectively represented as ovals, hexagons and rectangles.

Once the stakeholders and their goals and social relations have been identified, the analysis tries to enrich the model with more details. Goal refinement aims to analyze any goals of each actor, and is conducted from the perspective of the actor itself by using AND/OR decomposition. A graphical representation of goal refinement activity is given through *goal diagrams*. The outcome of this phase is a set of social relations among actors, defined incrementally by performing goal refinement on each goal, until all goals have been refined. Goal refinement builds goal hierarchies where lower goals are more specific and are motivated by goals higher in the hierarchy.

4. Modelling Actors

The first activity in the early requirements phase is actors' modeling. In our example we can list some of them:

Data Controller determines the purposes and means of the processing of personal data. In the University, the data controller is identified with Chancellor (as the post-holder is also the legal representative of the University).

Data Processor monitors personal data processing on behalf of the controller. In the University, these are:

- Faculty Deans;
- Head of Department;
- Central Directorate Managers, and in particular with:
  - Chief Executive Officer (CEO);
  - Chief Information Officer (CIO).

Data Processing Operator is appointed by the data controller or processor to perform the operations related to the data processing or to manage and maintain the information systems and services. At University of Trento, these are:

- Personal Data Processing Operator;
- Database Security Operator;
- Network Security Operator.

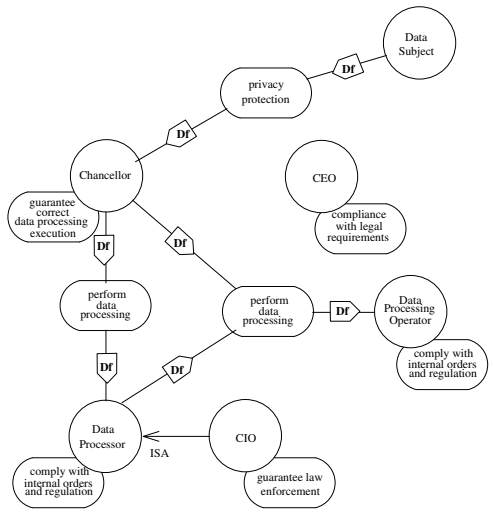
Data Subject is the natural or legal person to whom the personal data are related. In the Secure Tropos terminology, this is the legitimate owner of the data.

CERT is composed by:

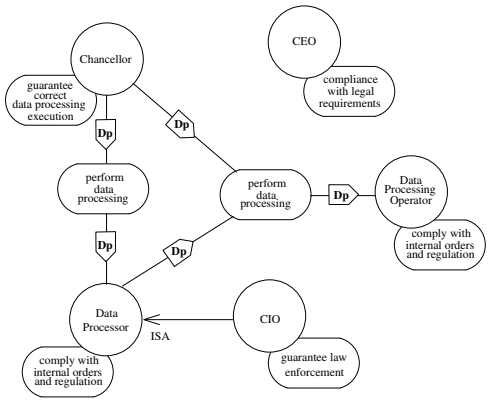
- the staff of ATI Network that manages the network infrastructure and services of the University;
- the Information Security Officer;
- the CIO.

To be more precise CERT includes a member in charge of security issues for every major ICT service center in the University.

In the underlying formal model based on datalog instances of actors are represented as constants satisfying atomic predicates for actors' types (e.g. being Chancellor) and binary predicates are used to link agents and goals.



(a) Functional Dependency Model



(b) Trust Management Implementation

Figure 1. Actor Diagrams

5. Dependencies and Delegation

The analysis proceeds introducing the functional dependencies and the delegation of permission between actors and the consequent integrated security and functional requirements. Figure 1(a) and Figure 1(b) show the functional dependency model and the trust management implementation. We use delegation of permission (**Dp**) to model the actual transfer of rights, and **Df** for functional dependency.

In the functional dependency model, *Chancellor* is associated with a single relevant goal: *guarantee correct data processing execution*, while *CEO* has an associated goal *compliance with legal requirements*. Along similar lines, *Data Processor* and *Data Processing Operator* want to *comply with internal orders and regulation*, while *CIO*, wants to *guarantee law enforcement*. Finally, the diagram includes some functional dependencies: *Data Subject* depends on *Chancellor* for *privacy protection* goal; *Chancellor* depends on *Data Processor* and *Data Processing Operator* to *perform data processing*; and, in turn, *Data Processor* depends on *Data Processing Operator* for it.

In the trust management implementation, *Chancellor* delegates permissions to *perform data processing* to *Data Processor* and *Data Processing Operator*. In turn, *Data Processor* delegates permissions to *perform data processing* to *Data Processing Operator*.

At this stage, the analysis already reveals a number of pitfalls in the actual document template provided by the minitry' agency. The most notable one is the absolute absence of functional dependencies between the Chancellor and the CEO, who is actually the one who runs the administration. Such functional dependency is present in the Universities statutes, but not here (an apparently unrelated document).

Another missing part in the trust management implementation is the delegation of permission from the data subject. This can be also automatically spotted with the techniques developed in [3]. Somehow paradoxically (for a document template enacted in fulfillment of a Data Protection Act) the process of acquisition of data (and the relative authorization) is neither mentioned nor forseen. In practice this gap is solved by the University by a blanket authorization: in all the paper or electronic data collection steps a signature is required to authorize the processing of data in compliance with the privacy legislation.

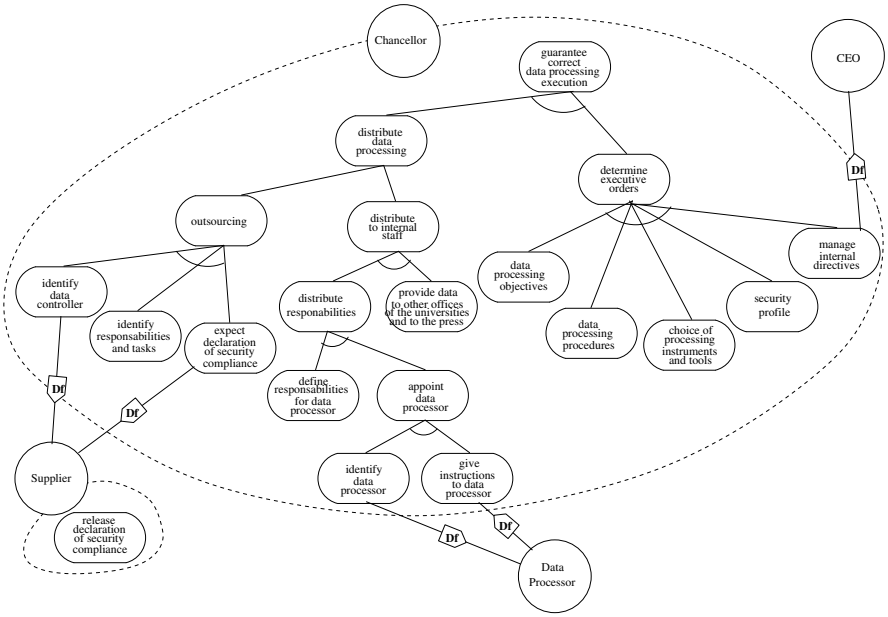


Figure 2. Functional Dependency Model for Chancellor

6. Goal Refinement

A first example of the goal refinement is given by the goal diagram depicted in Figure 2 for the *Chancellor*. The goal *guarantee correct data processing execution* is decomposed into *distribute data processing* and *determine executive orders*. We call this a "AND-decomposition". The goal *distribute data processing* is decomposed (OR-decomposition) into two subgoals: *outsourcing* and *distribute to internal staff*.

The security requirements of an organization outsourcing the management and control of all or some of its information system is addressed in a contract agreed between the parties. For example, the contract should address: how the legal requirements are to be met; what arrangements will be in place to ensure that all parties involved, including subcontractors, are aware of their security responsibilities; how the integrity and confidentiality of the organization's business assets are to be maintained and tested; etc. In a nutshell the contract should say that the goal *guarantee correct data processing execution* is also fulfilled by the service supplier. The contract should allow the security requirements and procedures to be expanded in a security management plan to be agreed between the two parties. Following these requirements, the goal *outsourcing* is AND-decomposed into *identify data controller*, *identify responsibilities and tasks*, and *expect declaration of security compliance*.

A second example, in Figure 3, shows the goal analysis for CIO, relative to the goal *guarantee law enforcement*. This goal is decomposed into *fulfill administrative and technical duties* and *manage security measures*. The goal *fulfill administrative and technical duties* is decomposed into three goals: *manage user access profile* for which *Data Processor* depends on *CIO*, *check activities' evolvement*, and *census data processing* for which *CIO* depends on *Data Processor*. The goal *manage user access profile* is decomposed into *create user access profile* and *guarantee authenticate connections*. The goal *create user access profile* is decomposed into *update authorization database*, *generate ID*, *generate and retrieve*

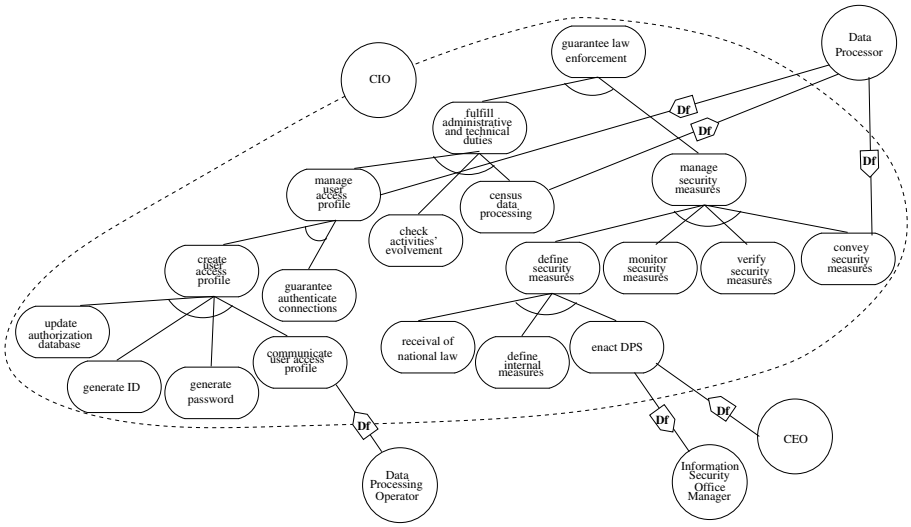


Figure 3. Functional Dependency Model for CIO

password,[1] and *communicate user access profile* for which *Data Processing Operator* depends on *CIO*. The goal *manage security measures* is decomposed into *define security measures*, *monitor security measures*, *verify security measures*, and *convey security measures* for which *Data Processor* depends on *CIO*. Essentially this map the formal requirements that a policy document should be approved by management, published and communicated, as appropriate, to all employees.

The goal diagram in Figure 4 shows the trust management implementation for *Chancellor* with respect to goal *guarantee correct data processing execution*. In particular, it points out that *Supplier* delegates a signed *declaration of security compliance* to *Chancellor* where *Supplier* engages in honoring and enforcing the undertaken responsibilities. This map the formal requirements that the University has security policies that requires adherence to several necessary precautions in order to maintain *privacy protection* in behalf of *Data Subject*. Further, *Chancellor* delegates *mail within instructions* to *Data Processor* and *executive orders list* to *CEO*.

[1]The procedure also includes some fuzzy steps on something that is a security anathema (helping users who forgot their password) but a fairly frequent problem.

Figure 5 shows the trust management implementation for *CIO*. The diagram displays that *Data Processor* delegates *data processing list* to *CIO* for census. Further, *CIO* delegates *ID*, *password* and *user access profile* to *Data Processing Operator*.

7. Adequacy and Analysis of the Model

The primitives suggested for Secure Tropos were sufficient to cope with the complexity of a real ISO-17799-like case study and the methodology allowed to pintpoint many issues.

For example, the first observation is that a trust model is not considered in the required procedures and documents. Trust relations are implicitly defined in the employment contract that actors draw up with the University. In absence of such model, some of the properties proposed in [3] cannot be verified since trust is at the base of such framework. Note also that, in according with the Code, data subjects own their personal data. In [3], we suggest to check if employees who are entitled to access to personal data, have previously gotten the permission from data subjects for them. In above models, this is not verified since there is not delegation from data subjects

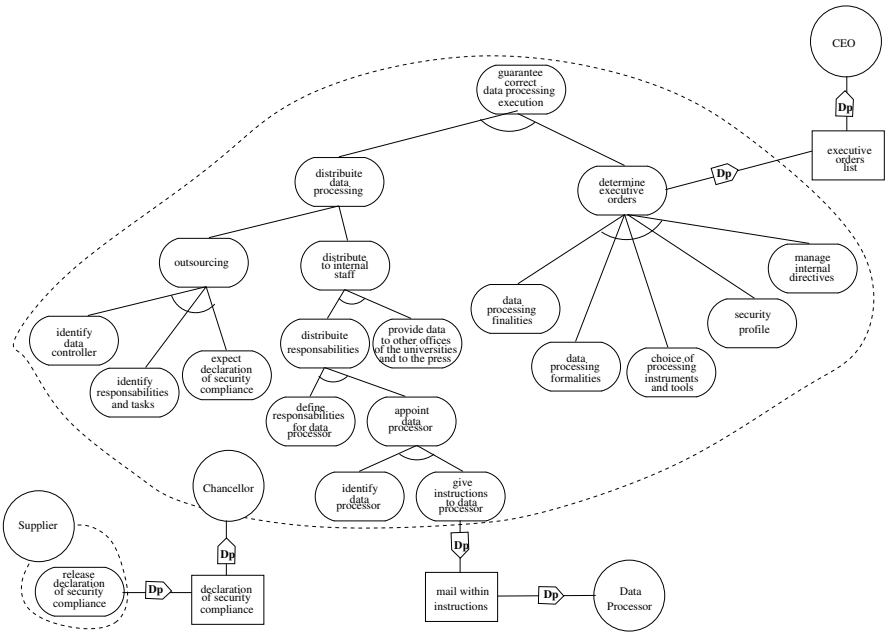


Figure 4. Trust Management Implementation for Chancellor

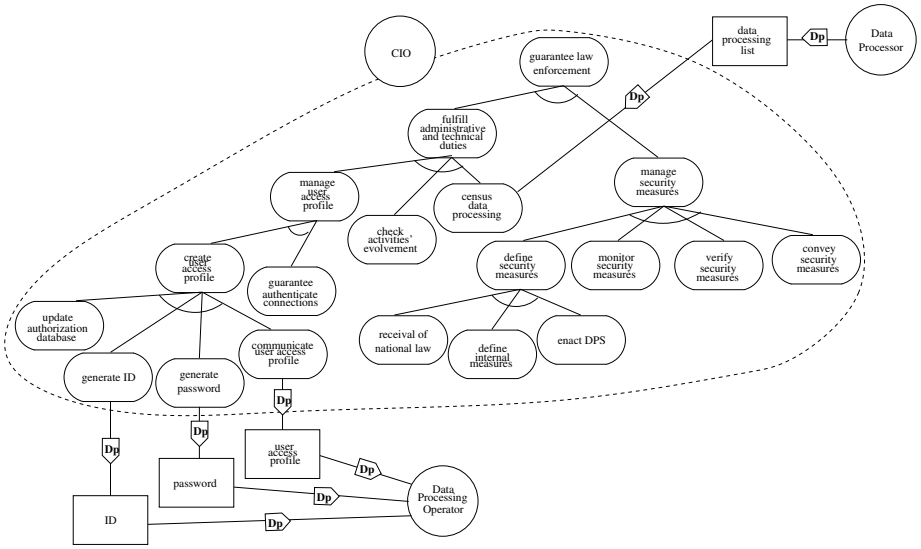


Figure 5. Trust Management Implementation for CIO

to employees for personal data. Essentially we only have a blanket authorization.

Further, DPS defines only objectives and responsibilities for the entities involved into the organization, but does not identify who is really able to provide services. For example, looking

at Figure 3 and 5, the CIO has the responsibilities to manage user access profile. In practice, he delegates the execution of this goal to an employee of the ICT Directorate that generates IDs and passwords, and then delegates them to data processing operators. Consequently, it is not possible capture requirements of availability unless an explicit model of the functional requirements is also given. For instance, we cannot verify whether data subjects delegate their personal data only to someone that is able to provide the requested service. This clashes with privacy principles and, specifically, with the notion of "limited collection": the collection of personal information should be limited to the minimum necessary for accomplishing the specified service.

Notice that this is not a problem of the University of Trento, but rather of the entire security assessment procedure in the state of the art: unless the ISO-17799 policy (or its equivalent DPS) is matched by a description of the functional goals of the organization it is not possible to conclude whether access is fair or respect least privileges principles. The same problem affects EPAL [13] and other privacy proposals in the literature.

The model has been further refined down to the the various offices and members of staff until it could be matched one-one with the actual DPS. These diagrams are not shown here and will appear in a companion technical report describing the entire study.

The most painful (and so far not formally analyzed part) is the treatment of manual non-ICT procedures. This difficulty steams from two main sources. The first one is that non-ICT procedures are often not completely formalized since there is no need for "programming" and "debugging" a human. This does not means that offices do not follow standard procedures but rather that these procedures are somehow "embedded" in the organization or the "office distributed knowledge". In absence of fully formalized functional procedures it is difficult to define the corresponding authorization and trust management procedures.

8. Related Case Studies and Conclusions

The last years have seen an increasing awareness that security and privacy play a key role in system development and deployment. This awareness has been matched by a number of research proposals on incorporating security and privacy considerations into the mainstream requirement and software engineering methodologies. Yet, only few papers describe complex case studies.

Becker et al. [8] use Cassandra to model and analyze an access control policy for a national electronic health record system. The background of this case study is the British National Health Service's current plan to develop an electronic data spine that will contain medical data for all patients in England. The proposed policies contain a total of 310 rules and define 58 parameterized roles.

In [1], Antòn et al. introduces a privacy goal taxonomy and reports the analysis of 23 Internet privacy policies for companies in three health care industries: pharmaceutical, health insurance and on-line drugstores. The identified goals are used to discover inner internal conflicts within privacy policies and conflicts with the corresponding websites and their manner of manage customers' personal data.

A study of the certification of information security management systems based on specifications promulgated by Taiwan's Ministry of Economic Affairs is proposed in [9]. In particular, this work shows the ability of Taiwan's information security management systems to meet the requirements proposed in international standards.

In this paper we have shown the Secure Tropos methodology at work on a real-life comprehensive case study encompassing on ISO-17799 security management policy. The proposed constructs and methodology were up the challenge and revealed a number of pitfalls, especially when the formal analysis techniques were applied.

Future work is in the full automated analysis of the policy at the level of individual staff members processing data.

REFERENCES

1. A. I. Antòn, J. B. Earp, A requirements taxonomy for reducing Web site privacy vulnerabilities, Requirements Eng. 9 (3) (2004) 169–185.
2. L. Liu, E. S. K. Yu, J. Mylopoulos, Security and Privacy Requirements Analysis within a Social Setting, in: Proc. of RE'03, 2003, pp. 151–161.
3. P. Giorgini, F. Massacci, J. Mylopoulous, N. Zannone, Requirements Engineering meets Trust Management: Model, Methodology, and Reasoning, in: Proc. of iTrust-04, LNCS 2995, 2004, pp. 176–190.
4. G. Sindre, A. L. Opdahl, Eliciting Security Requirements by Misuse Cases, in: Proc. of TOOLS Pacific 2000, 2000, pp. 120–131.
5. A. van Lamsweerde, S. Brohez, R. De Landtsheer, D. Janssens, From System Goals to Intruder Anti-Goals: Attack Generation and Resolution for Security Requirements Engineering, in: Proc. of RHAS'03, 2003, pp. 49–56.
6. J. Jürjens, Secure Systems Development with UML, Springer-Verlag, 2004.
7. ISO/IEC, Information technology – Code of practice for information security management, ISO/IEC 17799 (2000).
8. M. Y. Becker, P. Sewell, Cassandra: flexible trust management, applied to electronic health records, in: Proc. of CSFW'04, 2004, pp. 139–154.
9. A. R.-W. Fung, K.-J. Farn, A. C. Lin, Paper: a study on the certification of the information security management systems, Computer Standards and Interfaces 25 (5) (2003) 447–461.
10. P. Giorgini, F. Massacci, J. Mylopoulous, N. Zannone, Filling the gap between Requirements Engineering and Public Key/Trust Management Infrastructures, in: Proc. of EuroPKI'04, LNCS 3093, 2004, pp. 98–111.
11. P. Williams, Information Security Governance, Information Security Technical Report 6 (3) (2001) 60–70.
12. P. Bresciani, P. Giorgini, F. Giunchiglia, J. Mylopoulos, A. Perini, TROPOS: An Agent-Oriented Software Development Methodology, JAAMAS 8 (3) (2004) 203–236.
13. M. Backes, B. Pfitzmann, M. Schunter, A Toolkit for Managing Enterprise Privacy Policies, in: Proc. of ESORICS'03, LNCS 2808, 2003, pp. 162–180.