IMPLEMENTATION AND DESIGN OF SECURE BUSINESS PROCESS MODELS BASED ON ORGANISATIONAL GOALS

ADEEL PATRICK
(B-TECH, MBA)

Doctor of Philosophy
in
Business Administration

Abstract

Business processes are essential instruments used for the coordination of organisational activities in order to produce value in the form of products and services. Information security is an important non-functional characteristic of business processes due to the involvement of sensitive data exchanged between their participants. Therefore, potential security shortfalls can severely impact organisational reputation, customer trust and cause compliance issues. Nevertheless, despite its importance, security is often considered as a technical concern and treated as an afterthought during the design of information systems and the business processes which they support.

The consideration of security during the early design stages of information systems is highly beneficial. Goal-oriented security requirements engineering approaches can contribute to the early elicitation of system requirements at a high level of abstraction and capture the organisational context and rationale behind design choices. Aligning such requirements with process activities at the operational level augments the traceability between system models of different abstraction levels and leads to more robust and context-aware operationalisations of security. Therefore, there needs to be a well-defined and verifiable interconnection between a system's security requirements and its business process models.

This work introduces a framework for the design of secure business process models. It uses security-oriented goal models as its starting point to capture a socio-technical view of the system to-be and its security requirements during its early design stages. Concept mappings and model transformation rules are also introduced as a structured way of extracting business process skeletons from such goal models, in order to facilitate the alignment between the two different levels of abstraction. The extracted business process skeletons, are refined to complete business process models through the use of a set of security patterns, which standardise proven solutions to recurring security problems. Finally, the framework also offers security verification capabilities of the produced process models through the introduction of security-related attributes and model checking algorithms.

Evaluation of this work is performed: (i) through individual evaluation of its components via their application in real-life systems, (ii) a workshop-based modelling exercise where participants used and evaluated parts of the framework and (iii) a case study from the public administration domain where the overall

framework was applied in cooperation with stakeholders of the studied system. The evaluation indicated that the developed framework provides a structured approach which supports stakeholders in designing and evaluating secure business process models.

Contents

List of Figures

List of Tables

Chapter 2

Chapter 3

Chapter 4

IMPLEMENTATION AND DESIGN OF SECURE BUSINESS PROCESS MODELS BASED ON ORGANISATIONAL GOALS

Chapter 1

Introduction

Business processes are essential instruments used by organisations for the coordination of their activities in order to produce value in the form of products and services [1]. Therefore, they are an important asset, as they provide the blueprint to be followed in order to produce value for the organisation. As such, the de- sign of its business processes directly affects the way an organisation operates. Thus, the design of business processes is a critical aspect of organisational strategy and operations and is considered as an integral part of the business process management lifecycle. A number of modelling languages and techniques have been developed for the design of business processes, attempting to capture and represent elements of the contextual environment (e.g., actors, resources) under which a business process will operate.

Non-functional aspects are also critical to the quality and outcome of business processes. Security is one of them due to the potential impact of its shortcomings for organisations in terms of finances and reputation [2]. A recent global survey[1] interviewed over one thousand stakeholders of global organisations and discovered that less than 50% of EU organisations are aware of security-related regulations they are legally obligated to adhere to when conducting business, while a large percentage of them recognised that a security breach would result in direct financial loses, erosion of stakeholder value and loss of customer trust. An important takeaway of such reports is that information security cannot only be compromised due to the lack of technical controls but also due to the way that business is conducted. Thus, security shall not be considered as solely a low-level technical issue but it should be among the main concerns of the high-level organ-

[1] NTTSecurity - Business Security: Always a Journey, Never a Destination, 2013 Risk:Value Report.
Available at: http://it.nttdata.com/fileadmin/web_data/country/it/Global_Report_Risk-_Value_2013_A4_UEA_v6.pdf

isational strategy. In terms of business processes, security needs to be treated as an important process characteristic which needs to be considered during their early design stages [3]. To that end, specialised security-oriented extensions have been developed for the majority of the established process modelling languages. Nevertheless, capturing the rationale behind general and security-related design choices made during process design and aligning them to high-level strategic goals of the organisation, is outside of their scope [4].

The goals which an organisation aims to achieve by the execution of its business processes can provide highly relevant input during the process design stage. Goal-oriented requirements engineering (GORE) approaches use goals to capture the rationale behind design-time decisions. Therefore, when paired with process modelling approaches, they are a useful initial tool during the design of the business processes [5]. Specifically for the context of security, a number of security-oriented GORE approaches have been developed for the elicitation of security requirements which can later be integrated to process models to introduce security features aligned with the high-level strategy of the organisation.

However, holistic coverage of security is usually quite a complicated task for most of the existing approaches, which often specialise in either a specific category of security requirements (e.g. access control) or are tailored exclusively for risk management. In addition to that, approaches dealing with process security are usually equipped to deal with either the organisational (e.g., social interactions between users) or the technical perspective (e.g., implementation of security via services) of security. There is, therefore, a lack of a holistic, multi-perspective approach for designing secure business processes, aligned with organisational strategy. Other than the strategical alignment of security, flexibility is another desirable quality of business process designs. Due to the rapidly evolving environment in which organisations compete, continuous adjustment of their business processes is necessary. Keeping their processes up to date with such changes could be a challenge for organisations, since designing and implementing a secure process is a demanding task in terms of time and cost.

This work introduces a framework to guide the design of secure business process models derived from high-level, security-oriented goal models, which capture organisational goals and security requirements. To maintain a mapping between high level goals and security controls at the operational level, we transform goal models, created using the well-established Secure Tropos notation [6], as it provides concrete syntax able to capture both goal and security related concepts, to security-annotated BPMN 2.0 business process models [7] through the use of

intermediate hybrid process skeletons [8], [9]. The use of Secure Tropos as the starting point of the design process supported by the developed framework allows for a holistic security analysis, as it facilitates the elaboration of multiple perspectives of analysis. More specifically, the original and newly-introduced concepts of the Secure Tropos approach are able to capture the social perspective of the system to-be, through the modelling of system actors and their interactions and dependencies, and also facilitate the elicitation of security constraints, security-implementing mechanisms and risk-related aspects. Moreover, the transition from organisational level Secure Tropos goal models to BPMN 2.0 business process models, through the use of the hybrid reference process models, allows the further refinement of the security-related analysis at the operational level of abstraction. Finally, according the paradigm of design-time variability, the hybrid reference process model can generate a large number of similar, but also slightly different processes [10], according to contextual ad-hoc needs, through a structured decision support approach, also introduced as a component of the proposed framework.

The contributions of the proposed framework towards the state of the art are multi-faceted. As identified through a review of related works, presented in Chapter 2, even though approaches that combine goal-oriented requirements engineering and business process modelling exist, the analysis they support is limited. This work is the first research attempt which takes advantage of the multiple aspects of security analysis supported by Secure Tropos, extends them to provide risk-related analysis and decision support capabilities, and provides a structured way of transitioning to BPMN 2.0 business process models. Moreover, the intermediate hybrid reference process model, introduced by this work, is a novel artefact that bridges the gap between the organisational and operational level of security analysis and also promotes design-time flexibility, as the same hybrid reference process model can be instantiated into a multitude of similar but slightly different business process models, to accommodate situational system needs. Furthermore, this work also introduces a series of process-level security patterns to support the instantiation and refinement of the hybrid reference process model and a novel set of attribute-based security verification algorithms to ensure the adherence of the produced business process model to the initial set of security requirements. Each of the above contributions to the state of the art are achieved by the orchestrated use of the different components of the proposed framework, which will be presented in detail in Chapter 3.

Over the next section, basic concepts from the areas of business process man-

agement and security requirements engineering are introduced and defined in order to provide the reader with the necessary background information. Next, the aims, objectives and the research questions, which this work aims to tackle are presented in Section 1.2 and 1.3. Section 1.4 presents the research method that will be followed throughout this research project. Finally, the structure of the document is presented in Section 1.5 and the publications produced during the lifetime of this research project are presented in Section 1.6.

1.1 Background

1.1.1 Business Process Management

Several attempts at defining *Business Process Management* (BPM) have been identified in the literature of the area. The most established definition is provided by [1], stating:

> *"Business process management includes concepts, methods, and techniques to support the design, administration, configuration, enactment, and analysis of business processes."*

The domain of BPM is interdisciplinary as it borrows concepts from information technology and business management and applies them to design, analyse, automate and manage the business processes of an organisation [11]. A generic *BPM lifecycle* has been proposed to contextualise and provide order to this multitude of available actions related to business processes. A number of different views of this BPM lifecycle have been proposed in literature, from elaborate implementations (e.g., [1], [12], [13]) to more simplistic views (e.g., [11], [14], [15]). Regardless of the level of detail used to model the BPM lifecycle, there is a consensus regarding the sequence of the steps followed which are illustrated in Fig. 1.1. A brief overview of each main stage of the lifecycle will be provided next.

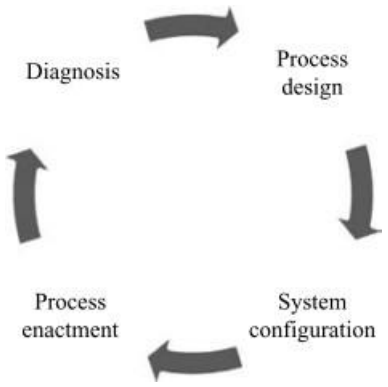

Figure 1.1: The BPM lifecycle [15]

During the initial stage, *Process Design*, the business processes of an organisation are mapped to process models using a variety of modelling methods. In order for the produced model to reflect an accurate representation of reality a number of contextual factors need to be taken into account during this stage (e.g., stakeholders, organisational goals etc.). Once an initial version of a process model has been created, it can be validated in order to verify that certain real-life properties and limitations of the process have been sufficiently modelled. Many iterations may be required at this stage in order to produce a complete model that can provide the framework for the execution of the process at the later stages of its lifecycle.

The next stage, *System configuration*, is concerned with configuring the infrastructure on which the designed process will be implemented. Such infrastructure can include a combination of physical IT systems and web-services explicitly configured as per the instructions provided by the process model.

During the *Process enactment* stage the designed business process is executed using the previously configured system. During the execution of the process different indicators can be defined in order to assess its performance and thus allow for runtime process monitoring. Additionally, it is common practice for process logs, which include information about the different process instances enacted, to be created and later be used for auditing purposes.

Finally, the *Diagnosis* stage offers the opportunity for the identification of errors and potential improvements to the execution of the business process. In this stage, using the generated process logs as input, a number of business activity monitoring and process mining techniques can be applied in order to assess different aspect of the process performance (e.g., execution time, bottlenecks) and identify whether and in what degree the executed process differs from its design.

The outcomes of these analyses can then be used for the redesign of the process, thus completing the circle and creating a feedback loop for next iterations of the process lifecycle.

The BPM lifecycle, with its discrete stages, provides a basis for the categorisation of different modelling standards, execution languages and software platforms related to the broader area of business process management. More specifically, as the practice of BPM continues to grow in popularity, an increasing number of software tools have been introduced in order to standardize and support the design, management and enactment business processes. Software systems aiming at supporting the design, implementation and evaluation of business processes are known as Business Process Management Systems (BPMS)[15], while systems supporting the automated enactment of the business process execution are known as Process Aware Information Systems (PAIS) [1].

The main role of such tools is to support and automate the application of different BPM standards during the lifecycle of a business process. In [15] a taxonomy of such standards is provided which distinguishes them according to their position in the BPM lifecycle and their similar characteristics, into the following groups:

- *Graphical standards*, expressing a business process in a diagrammatic way during its design stage. Popular standards within that group range from simple flowcharts and UML extensions (e.g., UML AD) to more semantically rich and rigidly defined notations such as BPMN 2.0 [7].

- *Execution standards*, facilitating the deployment and enactment of processes by translating the designs into markup process definition languages (e.g., BPEL, BPML) which are comprehensible by the process execution infrastructure.

- *Interchange standards*, used as an intermediate layer between graphical and execution standards. They are used to facilitate data exchanges between different design and execution languages and act as *"non-contextual translators between graphical standards and execution standards"* [15].

- *Diagnosis standards*, acting as diagnostic tools for the execution and post-execution analysis of process data for auditing, optimisation (e.g., identification of bottlenecks), performance evaluation and trend analysis of an organisation's processes. Such standards (e.g., BPQL) signify the most recent development in the field, extending its capabilities from simple work-

flow management techniques to a more holistic approach to business process management [15].

The focus of the research proposed by this work will be on the design of secure business processes. Therefore, from the classification previously presented, increased attention will be given to the design stage of the BPM lifecycle and the graphical standards for modelling business processes. For the elicitation of the security-related aspects and the modelling of the high level organisational context, which will be integrated into the designed processes, we will turn to goal-oriented security requirements engineering approaches.

1.1.2 Security Requirements Engineering

The elicitation and analysis of security requirements is an essential part in the requirements engineering process for the design of secure software systems. Security requirements engineering [16], promotes the adoption of a systematic process for identifying, analysing, and specifying the security requirements for a system to-be. The consideration of security during the early system development stages, rather than implementing security measures as an afterthought on an already designed system, can lead to more robust system designs that will not require costly readjustments during their lifecycle [17].

Another aspect to be considered during security requirements analysis is the socio-technical aspect of the system, which takes into account the complex, social interactions between the system's autonomous participants and software applications [18]. As a result of a socio-technical requirement analysis, which is not limited to only a technical consideration of the system to-be but also involves social entities and their high-level goals and constraints, new aspects of the system's design can be identified. Specifically in the context of security, threats resulting from social interactions of system actors can be identified during the early design stage and be mitigated by the system's design. Nevertheless, traditional requirements engineering approaches are not equipped to capture the wide range of concepts and system views required for such socio-technical requirements analysis. To overcome such challenge, the use of goal-oriented requirement engineering approaches is suggested by literature [19].

Goal-oriented requirements engineering (GORE) is a prominent model-based approach for the elicitation of functional (e.g., system functionalities) and non-functional (e.g., security) requirements in modern socio-technical systems. The basis of such approaches is the concept of goals, which is utilised to capture the

objectives that system stakeholders aim to achieve by using the system to-be. Through the use of goal decompositions by GORE approaches, an abstract goal can be dissected to simpler, more explicit sub-goals. In this manner, high-level business goals of system stakeholders can be broken down to specific low level objectives in a cohesive and organised manner. Therefore, GORE approaches in general, but also in the context of security, can capture the influence of the business context on a system's requirements and thus, enable the alignment between business and IT for organisations [20]. An extensive comparison of GORE approaches is presented in [21].

Since, in our research, goal-oriented security requirements engineering will be used for the elicitation of the security constraints, which will be then imposed on the designed business process models, it is worth briefly discussing some of the most prominent approaches in the area. The *i** modelling framework [22] is a prominent standard in the area of GORE and as a result a number of security-oriented GORE approaches have been developed based on it. Secure i* [23] and SI* [24] both use the well established notation of the i* framework to model actors, goals, resources and dependencies between them but also add concepts necessary for the analysis of security (e.g., threats, malicious actors, delegations, trust). *Tropos* [25] is another established software development methodology which has been the basis for the development of security-oriented extensions such STS-ml [26] and Secure Tropos [6] which introduce specialised concepts (e.g., security constraints, dependencies) and system modelling views to capture security requirements of modern multi-agent socio-technical systems. Secure Tropos is one of the basic components of our proposed framework, described in this work. Therefore, a discussion about this choice, as well as a comprehensive analysis of the concepts and modelling views of Secure Tropos, will be provided during the presentation of the basic components of our framework in Chapter 3. An extensive analysis and comparison of the rest of security-oriented requirements engineering approaches can be found in [27] and [28].

1.2 Aims and Objectives

The overall aim of this research project is to create a structured approach for the design of secure business process models which are aligned with the strategical objectives of the organisation. To achieve that such an approach should be able to: (i) support the analysis of both functional and non-functional (e.g., security) aspects of business process, (ii) facilitate decision making regarding security

choices and (iii) verify the security properties of the produced process designs. To provide further direction for this attempt, a number of objectives are specified below, the fulfilment of which will contribute to the achievement of the overarching project aim.

Obj.I: Create an approach that uses high-level, functional and non-functional organisational goals as input for the design of business processes.

Obj.II: Develop a structured way for producing business process designs able to operationalise the identified organisational goals.

Obj.III: Provide a new approach to support the selection of appropriate security configurations to be implemented at the business process level, according to situational needs and constraints.

Obj.IV: Provide a structured way for integrating predefined security configurations into business process models.

Obj.V: Develop an approach that enables the verification of the compliance of the security properties of a business process model to the security constraints identified at the organisational level.

1.3 Research Questions

The research questions presented below source from the overall aim and the individual objectives identified for this research project and aim to tackle the gaps identified in the literature with a novel and structured approach.

R.Q.1: How can information captured by organisational-level, security-oriented goal models be used as input for the creation of secure business process designs?

R.Q.2: How can the analysis and decision-making regarding security-related aspects of business process designs be supported?

R.Q.3: How can the adherence of a business process design to a series of security requirements be verified?

1.4 Research Methodology

The research method that was followed during this research project was based on the principles of *design science*. Design science represents the scientific study of designing and was introduced by H. Simon's 1969 publication of *"The Sciences of Artificial"*. Since then it has gained significant attention, especially in the field of information systems research, and is currently considered as an "equal companion" to natural and behavioural research [29], [30].

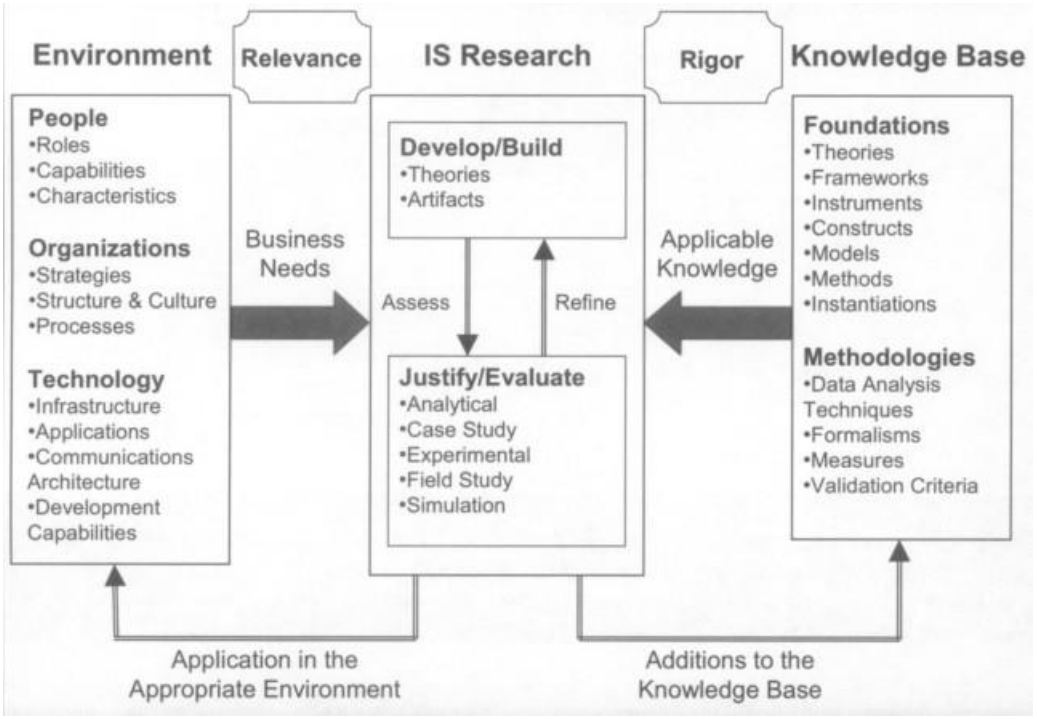

Figure 1.2: The Information Systems Research Framework [31]

According to the paradigm of design science, by the application of knowledge concerning tasks and situations, four types of artefacts can be created, namely constructs, models, methods and implementations, which are innovative and provide valuable solutions to problems identified in their environment [32]. The overall research framework for the development of information system artefacts, illustrated in Fig. 1.2, is centred around the "build and evaluate" iteration, which constitutes the core of the design science research approach. According to this framework, other than the iterative process for the artefact development, the contextual environment is used for the identification of the relevant problems to be satisfied, while the knowledge base is used as a source for relevant works and knowledge gaps. Once an artefact is built to perform a specific task, appropriate evaluation of its performance and contribution towards solving an identified

problem, shall be performed [30].

The basic steps followed by design science research contributions in the area of information systems, are defined by [30] as follows:

1. Identification and description of relevant organisational IT problem.
2. Demonstration of no existing solutions for the identified problem in the knowledge base of the area.
3. Development of a novel artefact (construct, model, method or instantiation).
4. Evaluation of the utility offered by the created artefact.
5. Articulation of added value provided by the artefact to the practice and knowledge-base.
6. Explanation of the practical implications of the developed solution.

In the context of our research project the developed artefact is a method, defined by [32] as a set of steps used to perform a task, aiming at the development of secure business processes. *Steps 1* and *2*, as defined above, were performed by reviewing the literature of the area of security in business process design, as presented in Chapter 2 of this document.

The development of the method, covering *Step 3* of the research framework, was the main activity of this research project. A number of discrete building blocks are required in order to create a method able to facilitate the development of secure business process designs, derived from high level organisational goal-models, as discussed at Chapter 3. Once such building blocks are solidified and a working method prototype has been tested as proof-of-concept, relevant computer-aided software engineering (CASE) tools to support the method's application, were identified, extended or developed from scratch.

As defined by *Step 4* of the research framework, an evaluation of the method's utility, efficacy and quality must be rigorously demonstrated in order for feedback to be provided back to the development phase, as part of the iterative *"build and evaluate"* loop [31]. There are several methods available for the evaluation of designed artefacts, with some examples mentioned in Fig. 1.2, out of which case studies are most commonly used in the field of information systems research [33].

The evaluation of this research project is presented in Chapter 4 and follows an iterative approach. First, each of the developed components of the proposed method was applied to real-life examples as a proof of concept. Several of the

publications originating from this research project (see Section 1.6) include applications of a single or a combination of components, to small scale real-life examples, in order for their functionality to be assessed in a qualitative manner and appropriate alterations to be made during the next iteration of their development. Additionally, components that have been developed from scratch were evaluated through workshop-based modelling exercises in order to assess their comprehensibility and ease-of-use, using the feedback of the workshop participants.

Later when a functional prototype of the whole method had been developed, a large-scale case study was performed for its evaluation. For this case study an organisation active in the development of security-critical systems was contacted and one of its e-governance information systems was selected as the main focus of the case study. The steps required for the design and execution of this case study followed the guidelines introduced by [34]. During its initial steps, quantitative metrics were identified to obtain a good indication of the effectiveness of the developed method. Such metrics will evaluate the conformance of the business process model to the initial goal model in terms of functional and security-related characteristics.

Moreover, qualitative evaluation approaches were explored during the case study design. More specifically, semi-structured interviews with the participating stakeholders of the organisation selected for the case study, provided us with insights regarding the perceived applicability and effectiveness of our method. Finally, another way to evaluate the contribution of the developed method was its ability to perform tasks that were previously not feasible by similar approaches. Such aspects were identified through the literature review (see Chapter 2) and aligned with our method's contribution in the concluding section of this work.

The outcome of the evaluation formed be the basis upon which the final conclusions were drawn, regarding the quality and effectiveness of our designed artefact. This provided the main input for completing *Steps 5* and *6* of the research framework, where the added value and practical implication of our method were identified, as discussed in Chapter 5.

1.5 Document Structure

The rest of the document is structured as follows, Chapter 2 presents a literature review which overviews related works in the area of business process security in order to identify overall research gaps and limitations. Chapter 3 presents the

framework developed as part of this research by first providing a general overview of its components and then presenting the theoretical background and application each individual component to a working example. Chapter 4 presents the different evaluation-related activities undertaken as part of this research project. Finally, Chapter 5 discusses the main contributions of this work and presents an overview of potential aspects that can be developed in future work. An overview of the contents of each chapter of this work along with their interconnections is provided in Fig. 1.3. The full material included in this research project, including all figures in full scale and resolution, are also available online[2].

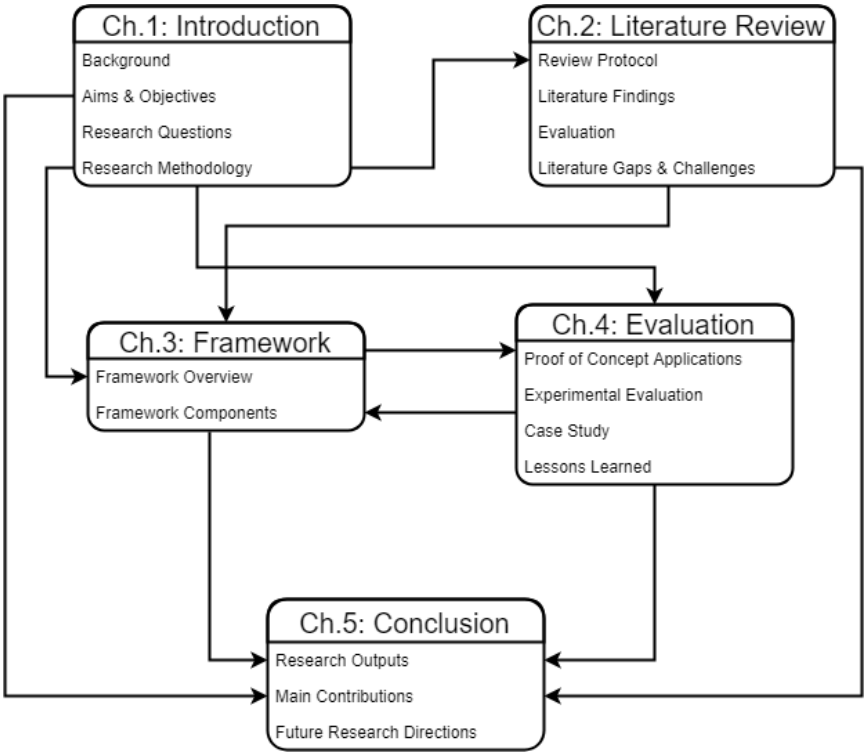

Figure 1.3: An overview of the thesis structure

1.6 Publications

The research leading to the development of the different components of the proposed framework has been presented and evaluated in a number of scientific publications. An overview of the framework components presented and evaluated in each of the publications listed below is provided in Section 4.1. As a result, parts of the text included in this document have previously appeared in the following

[2]Full thesis material also available at: http://www.sense-brighton.eu/our-team/argyropoulos/na-phd-project/

publications:

- Argyropoulos, N., Mouratidis, H., Fish, A.: *Towards the Derivation of Secure Business Process Designs*. In Proceeding of the 2nd International Workshop on Conceptual Modeling in Requirements and Business Analysis (MReBA 2015), pp. 248–258. Springer (2015) [8]
  Introduces contributions discussed in Sections 3.2, 3.4 and 3.5.

- Argyropoulos, N., Alcañiz, L. M., Mouratidis, H., Fish, A., Rosado, D. G., de Guzmán, I. G. R., Fernández-Medina, E.: *Eliciting Security Requirements for Business Processes of Legacy Systems*. In Proceedings of the 8th IFIP WG 8.1 working conference on the Practice of Enterprise Modelling (PoEM 2015), pp. 91–107. Springer (2015) [9]
  Introduces contributions discussed in Sections 3.2, 3.4 and 3.5.

- Mouratidis, H., Argyropoulos, N., Shei, S.: *Security Requirements Engineering for Cloud Computing: The Secure Tropos Approach*. In Karagiannis, D., Mayr, H.C., Mylopoulos, J. (Eds.), *Domain-Specific Conceptual Modeling*, (357–380). Springer (2016) [35]
  Introduces contributions discussed in Section 3.2.

- Argyropoulos, N., Kalloniatis, C., Mouratidis, H., Fish, A.: *Incorporating Privacy Patterns into Semi-Automatic Business Process Derivation*. In Proceedings of the 10th International Conference on Research Challenges in Information Science (RCIS 2016). (2016) [36]
  Introduces contributions discussed in Sections 3.5 and 5.3.

- Sprovieri, D., Argyropoulos, N., Mazo, R., Souveyet, C., Mouratidis, H., Fish, A.: *Security Alignment Analysis of Software Product Lines*. In Proceedings of the 4th International Conference on Enterprise Systems (ES 2016). (2016) [37]
  Introduces contributions discussed in Section 5.3.

- Argyropoulos, N., Shei, S., Kalloniatis, C., Mouratidis, H., Delaney, A., Fish, A., Gritzalis, S.: *A Semi-Automatic Approach for Eliciting Cloud Security and Privacy Requirements*. In Proceedings of the Hawaii International Conference on System Sciences (HICCS 2017) (2017) [38]
  Introduces contributions discussed in Sections 3.2, 3.4 and 3.5.

- Diamantopoulou, V., Argyropoulos, N., Kalloniatis, C., Gritzalis, S.: *Supporting the design of privacy-aware business processes via privacy process*

patterns. In Proceedings of the 11th International Conference on of Research Challenges in Information Science (RCIS), pp. 187–198. IEEE (2017) [39]
Introduces contributions discussed in Sections 3.5 and 5.3.

- Argyropoulos, N., Mouratidis, H., Fish, A.: *Supporting Secure Business Process Design via Security Process Patterns*. In Enterprise, Business-Process and Information Systems Modeling, pp. 19–33. Springer (2017) [40]
  Introduces contributions discussed in Section 3.5.

- Argyropoulos, N., Mouratidis, H., Fish, A.: *Attribute-Based Security Verification of Business Process Models*. In Proceedings of the 19th Conference on Business Informatics (CBI), pp. 43–52. IEEE (2017) [41]
  Introduces contributions discussed in Section 3.6.

- Pavlidis, M., Mouratidis, H., Panaousis, E., Argyropoulos, N.: *Selecting Security Mechanisms in SecureTropos*. In International Conference on Trust and Privacy in Digital Business, pp. 99–114. Springer (2017) [42]
  Introduces contributions discussed in Section 3.3.

- Argyropoulos, N., Angelopoulos, K., Mouratidis, H., Fish, A.: *Decision-Making in Security Requirements Engineering with Constrained Goal Models*. In Proceedings of the 1st International Workshop on SECurity and Privacy Requirements Engineering (SECPRE 2017). IEEE (2017) [43]
  Introduces contributions discussed in Section 3.3.

- Argyropoulos, N., Mouratidis, H., Fish, A.: *Enhancing Secure Business Process Design with Security Process Patterns*. Software and Systems Modeling (SoSyM) journal. Springer (2018) [Under Review] [44]
  Introduces contributions discussed in Section 3.5.

- Argyropoulos, N., Angelopoulos, K., Mouratidis, H., Fish, A.: *Risk-Aware Decision Support with Constrained Goal Models*. Information and Computer Security journal (ICS). Emerald Publishing (2018) [Accepted for Publication] [45]
  Introduces contributions discussed in Section 3.3.

Chapter 2

Literature Review

In this work a systematic literature review is performed according to the guidelines provided in [46]. The objective of this review is to synthesize the information collected by the literature in the area of secure business process modelling and identify current challenges, research gaps and future directions for researchers. According to the identified guidelines the first phase of the review consists of the planning, which includes the identification of the review protocol to be followed. Next the review is conducted by searching, filtering and selecting the relevant works and finally the collected data is synthesized and the report is created.

2.1 Review Protocol

In order to identify relevant works for this review, a number of selection criteria were established. Firstly, in order for an article to be considered relevant, it needed to be focused on both the overall area of security and business process modelling. Therefore, works focusing on business process modelling or information security in general were excluded since the structure of the keywords used made sure only works in the intersection of both areas appeared in the search results. Since the overall focus of this research is on the design of secure business processes, modelling is an essential aspect to be considered. Thus, the identified works had to be under the umbrella of model-driven engineering [47] and involve "model-driven" approaches to process design in order to be included in our review. To ensure a broad coverage of security related concepts in the context of business processes the identified works had to provide security and/or risk related analysis.

The search engine of Web of Science[1] was utilised for the identification of

[1] www.webofknowledge.com

relevant literature. This selection was mainly due to the wide variety of relevant journals indexed at Web of Science and its ability to select different filtering parameters and structure the keywords with logical operators and wildcard characters (e.g., AND, OR, *, "). The keywords used for our searches were *"business process security"*, *"workflow security"* and *"business process*" AND "security"*. Backwards snowballing techniques were also applied when relevant works were referenced by the identified literature. The only exclusion criterion applied to the search results was their language, which was limited to English only. No limitation on publication dates was enforced and as a result the identified literature's spans from 1998 to 2017. The initial number of records recovered by each of the keywords used are included in Tab. 2.1, in total 807 records were identified.

Keywords	*No. of records*
"workflow security"	22
"business process security"	16
"business process*" AND security	769
Total:	*807*

Table 2.1: Number of records by keyword search

The first stage of the selection of relevant works, according to the previously discussed criteria, was performed by checking the title and abstract of each of the identified works. During this stage each of the 807 search results was accessed, each title and abstract were read and if they were deemed as relevant to our review they were kept for further evaluation. As a result of this process, at the end of the first stage, 145 articles have been selected for further reading.

The second stage of the selection process included reading the whole body of the selected resources and deciding which should be included in the final review. For each resource a number of keywords were assigned, which later assisted in the categorisation of the selected literature in groups. As a result of the second round of the selection process 61 articles have been selected to be included in the final review, three of which were literature reviews or literature mapping studies while the rest presented novel contributions.

The above parameters led us to the exclusion of a number of different groups of articles identified during the literature review process. A body of works covering access control configurations for business processes has been excluded the review, even though it is usually considered as a sub-type of security control. The reasons for this exclusion are: i) such approaches are usually performed in a rule or role-based manner using formal languages, therefore not fitting within the "model-driven" scope of this review, ii) they are not considered adequate

as a standalone approach for the analysis of modern environments under which business processes can operate (e.g., cloud) [48] and iii) the level of process modelling abstraction which such approaches require is often significantly lower that the one used during the design stage of business processes, which is the focus of this work. Nevertheless, readers interested in access control configurations for business process or workflow systems can refer to [49] for a comprehensive review. Privacy was another security-adjacent aspect that appeared in a number of the identified works. Even though privacy tends to be grouped with other types of security requirements, it is a separate and multi-faceted concern. Privacy requirements engineering recognises a number of different types of privacy requirements which can often conflict with the security requirements of a system under design. Thus, an exhaustive search and inclusion of privacy-related works in this review would significantly increase its scale while also shifting the focus to other concerns (e.g., privacy analysis, conflicts between security and privacy) which are outside the scope of the current research project. To avoid that we did not extend the literature search criteria to include the term "privacy" in order to maintain the security-oriented focus of the review. Nevertheless, some of the works identified and discussed in the rest of the chapter also deal with privacy as another aspect of information security without specialising in it.

The exclusion of the groups of works discussed above increased the focus of this literature review towards the scope of the current research project at the cost of the overall coverage of the research area. This trade-off was necessary in order to identify a manageable set of literature with the highest possible relevance to the objectives of this work. Thus, the literature presented and analysed at the rest of this chapter provides an accurate snapshot of works focused on model-driven information security analysis for the design of business processes. Future research attempts can include a broader spectrum of works in order to identify a wider range of research gaps, using the outcomes of this review as a starting point.

2.2 Literature Findings

2.2.1 Security by Model Transformation

From Goal to Process Models

In order to successfully design business processes it is highly important to have an understanding of the organisational context within which such processes will

be enacted. Specifically, the goals that an organisation aims to achieve by the execution of such processes provide highly relevant input for the identification of the characteristics of a process. Since graphical process modelling standards alone are not fully equipped to capture the strategic rationale (e.g., high-level goals) which processes should achieve [4], it is preferable to perform these actions using goal-oriented languages and notation [15]. Goal-oriented requirements engineering (GORE) provides such a framework for capturing and analysing the intentions of stakeholders and translating them to system requirements [50].

GORE approaches elicit top-level organisational goals and through the use of goal-models they decompose them to a series of simpler, lower level sub-goals. A number of different organisational actors can be responsible for the achievement of these sub-goals using available resources (e.g., information, physical infrastructure). Nevertheless, while goal models can provide a high-level direction and rationale in the form of goals, they lack the ability to adequately identify the specifics of how these organisational goals will be reached. Therefore, it is recognised that GORE should be used more as an initial influence rather than a complete solution for the further development of organisational activities, such as process design [5]. As a result a number of approaches have been developed which use goal-models as the starting point for the elicitation and elaboration of process designs. In the rest of this section we will focus on approaches with a clear security orientation, which make use of such model transformations to integrate security features in business process models.

In [51], *SecureBPEL* is introduced as a process specification language emphasising in the security aspect of business processes, aiming to bridge the gap between the early requirement analysis and the development of secure workflows. This method is essentially an extension of the BPEL execution standard enriched with constructs from the Secure Tropos goal-oriented security requirements engineering framework. Such concepts are used to enforce delegation and trust requirements in web services used to support the designed business process, thereby extending the functionalities of traditional BPEL. SecureBPEL offers a way of deriving process skeletons based on requirements specified early in the development process, which can be then refined to produce secure workflows with minimal effort.

In [52], [53] the PriS framework is introduced for the incorporation of privacy requirements into business process designs. In order to achieve that, PriS initially models the systems requirements via goal models with privacy requirements as a special type of goal that impacts the achievement of other system goals. Next

the impact of the privacy goal to the organisational processes is identified and the processes are adjusted accordingly using a set of privacy-process patterns. Finally the implementation techniques best supporting these processes are selected according to the organisation's specific needs. A formal language (Formal PriS) is also introduced in order to precisely describe the concepts and support the activities introduced by the PriS framework. Overall PriS provides a coherent path from high-level organisational needs all the way to system configurations that satisfy them, further supported by a formal language which allows for precise transitions between the different levels of abstraction.

The work presented in [54] begins from legacy business process designs from which functional and security requirements are extracted and expressed via SI* organisational goal models. These requirements can then be refined at an organisational level and be transformed into BPMN specifications. As a result a new re-engineered process design emerges which can cover new requirements not operationalised by the initial legacy business process. This work also introduces the notion of goal equivalence, used to compare process models in terms of their ability to operationalise certain goals of the organisational goal-model. Finally some soundness and completeness properties are defined in order to verify that all the information captured in the organisational models is preserved in the final process model designs.

On a similar theme, [55] introduces the BP&SLA methodology for the identification of services to implement business process along with their related service level agreements (SLAs) that can guarantee the satisfaction of certain organisational requirements. To bridge the gap between abstractly defined organisational needs and executable business processes, goal-models are constructed during the initial phase of the method's application. Next, an intermediate structure, defined as business process hypergraph, is derived from the goal-model by automatically matching sub-processes with goals that they can achieve. Additionally some quality of service attributes can be defined for each sub-process, along with a trust level value which indicates its degree of satisfaction. Next a hierarchy of business processes is extracted where the sub-processes are grouped, ordered and connected with delegation and trust relationships. Finally, using constraint programming approaches, each node of the hierarchical business process hypergraph is matched with a service with SLAs that satisfy the organisational needs earlier expressed as quality of service attributes.

Another work focusing on the aspect of security during the design of business processes is presented at [56]. The *SecCo* (Security via Commitments) framework

is introduced for the elicitation of security requirements through the modelling and analysis of objectives, roles and social interactions of actors from an organizational perspective. The cornerstone of SecCo is the concept of social commitments between actors, based on which is the identification and expression of the security needs to be incorporated in the organisation's business processes. These security needs are extracted from an aggregation of goal-oriented models expressing the business view of the organisation and are transformed into social commitments between actors "promising" to fulfil these needs in the interactions they participate. Finally these commitments can be incorporated as textual annotations to high-level BPMN conversation diagrams.

The work of [57] extends the Formal Tropos requirements engineering approach to support security policies. The policy-extended Formal Tropos models consist of custom textual policies, manually introduced by system designers, expressed in the grammar proposed by this work. Once the sum of policies has been created a model transformation takes place, where through the use of the Atlas Transformation Language (ATL) the business requirement model is transformed into a business process specification expressed in Business Process Modelling Ontology (BPMO). The BPMO instance produced by such transformation can be used as input in graphical modelling environments to produce business process design skeletons. In contrast with the rest of the works discussed in this section, this approach does not use graphical goal models as a starting point but instead textually defines policies which can be used to express security constraints. Therefore, the main contribution of this work is the ability to produced rich requirements specifications during the early design stages via the policy-extended Formal Tropos notation and automatically transform them into accurate and compliant business process designs.

MDA-based Model Transformations

Model-Driven Architecture (MDA) is based on the idea of using models to perform software development and *"separating the specification of the operation of a system from the details of the way that system uses the capabilities of its platform"* [47]. The separation of concerns, around which the MDA approach is built, is supported by three distinct viewpoints from which the system under development can be considered. At the highest level of abstraction, the *computationally independent model* (CIM) of the system is created to capture the domain and overall environment within which the system will operate. It does not include details on the specifics of the system's structure but rather focuses on capturing

its requirements. At the next abstraction level, the *platform independent model* (PIM) of the system presents a technology-neutral viewpoint of the system's configuration, in order to allow a system representation that can be replicated in a number of different technological platforms. Finally, the *platform specific model* (PSM) represents the lowest level of abstraction by instantiating the specifications of a PIM to a particular type of technological platform. For transitioning between the different model of the same system model transformation techniques can be applied. The interoperability and reusability of the created models are the main advantages of this tiered approach to system modelling, introduced by MDA.

A method for transferring secure business process to cloud environments is presented in [58]. More specifically this work focuses on partitioning acentralised business process to multiple cloud providers assigning different parts of the process to a different provider depending on its security constraints. To achieve that each process activity is assigned with a "security level" depending on the security constraints imposed on it. Next, the activities are assigned to the cloud provider which can better cover their individual security needs. The separate sequences of activities that are now partitioned between different cloud providers are then synchronised in order to maintain the functionality and quality of service of the original process. Finally, the optimised and decentralised business process model is automatically transformed to BPEL in order to facilitate its deployment.

The M-BPSec framework [59], [60] aims to create secure business process specifications by transforming computationally independent models (CIMs) to platform independent models (PIMs) by the application of predefined transformation rules. At the CIM level, business analysts can express their security requirements at a high level of abstraction, on the business process model via a series of padlock symbols. The secure business process can either be modelled using UML activity diagrams (UML-AD) or BPMN. In the latter case a horizontal transformation can be applied to transition from a BPMN to a UML-AD secure business process diagram, the rules of which are specified in QVT, as presented in [61]. The vertical transition from a CIM secure business process (SBP) model to PIMs of UML class and use case diagrams is once again performed using transformation rules, expressed in QVT [62]. Such diagrams can capture security related information which is abstractly defined during the process specification and provide a higher level of detail which can assist the process implementation. Automated support for the modelling and transformations between the different components of the framework is provided by the BPSec-Tool.

In the same context of model transformation, the SECTET framework [63] is developed for the implementation of security in business process. The first step in the framework is the creation of a platform independent model (PIM) using a UML profile, called SECTET UML, to capture the initial business requirements. SECTET-PL, a domain-specific predicative language, is also introduced for the definition of security policies and is integrated with the UML modelling component of the framework. For the transition to a platform specific model (PSM) a series of transformation rules are defined in QVT. Using these rules XACML security policies can be generated from the requirements model.

The work presented in [64] aims to produce security service configurations beginning from graphical process models. At the CIM level a business process is modelled in BPMN and annotated with a security-oriented notation, introduced in [65]. Security policy configurations are extracted from the security annotated process model at the PIM level, after the process model has been verified by a model checker. Finally, a platform specific model (PSM) can be produced by transforming the security policy specifications to specific service configurations using XACML or WS-Security. Thus, this security-oriented framework can produce service-oriented target architectures by a series of transformations which begin from a BPMN process model.

In [66] an integrated approach for creating secure service compositions by modelling and enforcing secure workflows is introduced. At the CIM level a generic metamodel for secure object flows is introduced, including concepts that can be integrated to common modelling languages to extend their capabilities for describing security-related aspects. At the PIM level such concepts are applied to UML activity diagrams to allow the modelling of secure workflows. Finally at the PSM level the secure workflows earlier introduced are transformed into service specifications supporting various standards such as WS-BPEL, WSDL and WS-SecurityPolicy.

The work of [67] introduces BPA-Sec4Cloud, an approach for automating service-based security-aware business processes in cloud environments. During the design stage, an abstract business process model is constructed and annotated with high-level security requirements. This models is then further analysed to specify which of its activities are automated or manual and what data types need to be used to represent the information exchanges included in the process. Finally, the initial security requirements are further analysed by security experts to provide further details regarding to their level of criticality (i.e., *Low, Medium, High*) and potential countermeasures that can be used to satisfy them. Next,

completed, the activities of the process model are matched to web services which can be used to implement them, thus creating an *"enriched business process"*. The next phase of the approach translates this enriched business process, first to a platform independent (PIM) and subsequently to a platform specific (PSM) model. Finally, the PSM model is used as input for executable business process source code generation. The various steps of the approach are supported by the BPA-Sec4Cloud Tooling.

2.2.2 Security-annotated Business Process Models

During the design stage of the business process management lifecycle, the processes that an organisation utilises in order to achieve its goals are modelled. A number of techniques exist for the purposes of business process modelling, with graphical standards being the most intuitive and comprehensible amongst them. Using graphical standards, process designers are able to visualize the sequence of activities, which can range from sub-processes to simple tasks, the flow of information within the organisational structure, as well as events and decision points which may trigger discrete or concurrent sub-activites [68]. In the rest of this section we will first give an overview of the most widely used graphical process modelling standards followed by some of their security-oriented extensions for the annotation of business process models.

Graphical Process Modelling Standards

UML Activity Diagrams *(UML AD)* can be used to describe the behaviour of business processes during process modelling [69]. The UML framework, from which this standard sources, adopts the object-oriented approach to modelling and is characterized by intuitiveness and flexibility which has made it a popular choice in the overall area of system analysis and design. UML AD includes a wide range of standard UML concepts used to model the basic workflow elements such as actors, activities which can be further decomposed to sub-activities and modelled as states and message exchanges modelled as signals.

Despite their intuitiveness and ease-of-use, UML ADs offer limited capabilities for modelling organisational and resource related aspects of business processes, thus limiting the expressive ability of the produced designs regarding their interactions with their contextual environment. As a result they produce single-perspective models, unable to capture the multiple levels of abstraction necessary for illustrating and understanding modern business processes [15].

Event-Driven Process Chains *(EPC)* is another graphical standard for business process modelling characterised by intuitive and easily-comprehensible concepts and notation. EPC uses the concept of function to describe the activities of a business process, events to describe the conditions necessary for the transition from one activity to the next and logical connectors (i.e., AND, OR, XOR) to connect events and activities when necessary [70]. EPCs have a number of applications in industrial software platforms (e.g., SAP R/3), thus gaining popularity as a language for expressing business processes in practice [69].

Nevertheless and despite the popularity of this approach, issues with the definition of its syntax and semantics have been identified. As mentioned in [70], there is ambiguity in the definition of the language's concepts as well as an inability to check the completeness of the produced models, sourcing from the lack of standardisation. All these factors heavily affect the quality of the produced process designs as well as their transferability between different process modelling and execution platforms.

Business Process Modelling and Notation *(BPMN)* is currently considered the *"de-facto standard"* graphical modelling language for business processes [3], [71], [72]. Its latest version was introduced in 2011 by the Object Management Group (OMG) [7] and contains a wide range of semantics, which allow the expression of a series of relevant concepts (i.e., activities, events, complex workflows, conditional gateways etc.) in a well-defined and precise manner. It supports different levels of abstraction of process designs, ranging from private, internal business process models to collaborative conversation diagrams involving multiple organisations [69]. These characteristics allow BPMN process models to be easily mapped to execution code while also provide them with the necessary flexibility to support the analysis of business processes from multiple perspectives with varying levels of granularity [15].

Since BPMN was conceived and developed as a process-centric language, it has a clear advantage compared to object-oriented approaches (e.g., UML AD) when it comes to its adoption by business analysts. Moreover, BPMN has been proven superior to other graphical standards (e.g., EPC), when their ability to express real-life concepts was compared in [68]. Additionally it also provides the most complete approach towards expressing organisational structures and boundaries by utilising the intuitive pool and lane concepts [15]. Finally, BPMN has in place *"extension definition"* mechanisms that allow the introduction of new attributes to its meta-model in order to facilitate the definition of domain-specific extensions [73]. This feature, not found in any of the other modelling languages of the area,

ensures the integrity of its core elements and its semantic robustness even when constructs are extended to support new domains of interest.

Security-Related Extensions of Graphical Process Modelling Standards

In [74] the authors propose some extensions to the BPMN standard by expanding some of its existing elements (i.e., artefacts, data objects, groups and text annotations) in order to express security requirements such as integrity, privacy, non-repudiation and access control. These requirements are visually represented at business process diagrams with padlock symbols assigned on BPMN elements, each of which containing a capital letter to differentiate between different types of requirements. Similarly, the work of [75] extends the BPMN notation by introducing security-related notation to express security requirements on process models. A "security profile" is also introduced to express the attributes and constraints of each type of security requirement, analogous to the profiles introduced by UML.

The Sec-MoSC framework is another security-oriented BPMN extension introduced in [76]. Sec-MoSC aims to integrate security requirements with BPMN process models by introducing the concepts of NF-Attribute, NF-Statement and NF-Action. The NF-Attribute expresses the security requirements of a spe- cific process fragment, the NF-Statement quantifies that requirement (e.g., High, Medium,Low) while the NF-Action models mechanisms that can be implemented to satisfy such requirements. After the security annotated model is refined it can be automatically translated to BPEL execution code with security configurations sourcing from the parameters set at the process model level. The same authors have created the Sec-MoSC Tooling [77], a set of tools that offers support and automation during the implementation of the Sec-MoSC framework.

In [78] an extension to BPMN is proposed that allows the modelling of non-functional requirements (NFRs) such as security, performance and quality of service. In order to achieve that, the concepts of operating conditions and control cases are introduced as extensions to the existing BPMN notation. The operating condition is used for the modelling of constraints limiting a specific activity of the process while the control case captures business controls that should be put in place to mitigate the risk imposed on an activity by an operating condition. This set of concepts can be used to address both the rationale (*"why"*) and the possible configurations (*"what"*) aiming to address non-functional concerns of business process models, including but not limited to security.

In [79] an approach for the specification and expression of "security goals"

in business processes is introduced. Initially, the security goals of authorisation, authentication, integrity and confidentiality are expressed as constraints through security constraint models. Such models relate security goals to organisational entities, creating rules that restrict particular associations between these entities. These abstract security goal specifications are then introduced into the business process layer of the organisation's enterprise model, thus defining security in a high level of abstraction, communicable to non-technical stakeholders via annotations to BPMN process models. As these generic security related annotations at activities and message flows of the process diagrams do not affect the control- and data-flow characteristics of the models, they can be applied to other process modelling notations other than BPMN.

A language for textual security annotations of BPMN process models is introduced in [80], supported by a semantic annotator tool. Security constraints for business processes are represented using an ontology and a knowledge base holds previously defined correct annotations so guidance and suggestions can be provided to the modeller during the annotation of a process model.

The work presented in [81] introduces BPMN-sec, a BPMN extension focusing on the security aspect of business processes outsourced to the cloud. In BPMN-sec two main types of stakeholders are involved, namely a user-side and a cloud-side, each controlling different parts of the process. Initially the whole business process is modelled and developed at the user-side. With the application of an optimization algorithm, parts of it are later selected for migration at the cloud-side. In order to elaborate on the security of these sub-processes deployed at the cloud, BPMN-sec extends the meta-model of BPMN with security-related concepts defined as UML profiles. These profiles can represent several security requirements, such as privacy, availability, access control, non-repudiation and integrity, and can be associated with certain BPMN elements, such as pools, lanes, activities, data, and message flows.

The work of [82] introduces the foundation for an information security and assurance extension of BPMN by proposing concept alignments between the domain of security and process modelling. Building upon this foundation, [83] introduces SecureBPMN, a model-based approach for designing business process driven systems. The focus of SecureBPMN is on the expression of security requirements concerning "binding of duty" and "need-to-know". These requirements are expressed by meta-model extensions of BPMN that allow the specification of role-based access control, separation and binding of duty constraints and need-to-know principles in business process diagrams, through diagrammatic representations.

Specialised tool platforms are also extended, as part of this work, to accommodate the newly introduced expansion.

A similar attempt is described in [72], [84], [85] where SecBPMN2 is introduced as a BPMN security-oriented extension with additional annotation for the representation of security related concepts at business process models. Via a series of newly introduced security annotations, a number of aspects (e.g., accountability, authenticity, confidentiality, integrity, privacy, non-repudiation) can be represented and linked to existing BPMN elements. In addition to the annotations, the BPMN-Q query language is also extended to support the modelling of security policies. The security policies expressed through this extension, named SecBPMN2-Q, along with the security-annotated process model, can then be used as the input of an automated algorithm that verifies the existence of paths within the designed process that satisfy these policies. Thus this work contributes to the development of secure and expressive process models with verification capabilities.

The work introduced in [86]–[88] extends UML use-case diagrams to express security requirements. Security is expressed via textual annotations structured in a formal language (FML) in order to create secure system specifications. Finally, elaboration is provided on how such secure designs can be transformed to machine-readable code.

UML Activity Diagrams (ADs) have been the focal point of a number of security-related UML extensions. In [89] UML ADs are utilised to capture misuse cases. In such mal-activity diagrams malicious actors and their actions are modelled along with the process they negatively impact. New UML stereotypes and notation are introduced in [90], [91] as part of a security-oriented domain specific language. Activities in UML ADs can be linked with security requirements expressed by such stereotypes to capture security-related aspects of the process design. The work of Rodriguez et al. [92]–[95] introduces new notation in the form of padlock symbols to express security requirements in UML ADs. In addition to that, the UML metamodel is extended with security related datatypes and new stereotypes are defined. This domain specific extension of UML is used as an integral part of the M-BPSec framework, as previously discussed.

In [96] Event-Driven Process Chains (EPCs) are used as the basis for a security-oriented modelling extension. This work introduces a set of security symbols used to express security requirements which are introduced to EPC process models in order to secure data items and activities. The created security-annotated process model can then be automatically transformed into a series of

appropriate web services which can be used to realise its implementation. Petri-Nets are also extended to support security aspects in [97]. This work introduces IF-Net, a meta-model for the formal specification of business processes which allows the consideration of security-related aspects in control and data flows. The basic concept of IF-Net is the classification of system objects via labelling, in levels of incremental security with subjects only allowed to access specific levels according to their security clearance.

2.2.3 Risk Management at Business Process Models

A survey of works related to risk in the context of business process security is presented in [98]. As a result of the synthesis of the identified literature a roadmap for risk-aware business process management is created. According to this, new approaches in this area should produce models that combine business process and security concepts and can capture detection, recovery and countermeasures. They should also be able to integrate security and economic aspects during risk management while also be able to simulate the produced process designs in order to verify their completeness.

Focusing on the area of risk management of business processes the works in [99], [100] introduce the ATANA framework. This multi-step approach aims to assess the risks of business processes and introduce the appropriate safeguards for their mitigation. During the first step the business processes are modelled and their potential threats and vulnerabilities are identified by analysts using a number of available techniques (e.g., misuse cases). The deliverables of the first step are used as input for the workshop-based risk assessment which is performed during the next step. The main objective of that step is the composition of risks as asset/threat/vulnerability tuples, the definition of cost/benefit categories and the assignment of values to the identified risks and safeguards. To achieve this objective stakeholders from different domains of the organisation participate in workshop sessions performing risk assessment. Finally, the most efficient and effective safeguards are selected in order to be implemented, a decision which is based on the output of the workshop-based risk assessment.

The works in [101], [102] introduce OPBUS, a risk-aware business process modelling framework. The architecture of OPBUS is layered with the first layer revolving around process modelling, using BPMN with textual annotations capturing risk information. The same authors propose in [103] security pattern templates to facilitate the selection of risk treatment solutions which can be utilised at the modelling layer. The application layer maps the risk-related information

of the modelling layer to a constraint model. This constraint model is used as an input for the fault tolerance layer where constraint programming techniques are used for the retrieval of an optimal solution. The automation of security configuration selection is further elaborated by the authors in [104]. Finally at the service layer the optimal security configuration is implemented as a series of services.

In [105] a methodology for the analysis and evaluation of threat impact is presented. This methodology aims to produce a set of security requirements based on the identified threats, which will guarantee a systems security level. In order to achieve that the methodology begins by capturing the business processes using UML ADs. Next the process models are extended with the addition of potential threats, as threatening actions interjected into the normal activity flow. Next the produced model is translated into asset-flows in the form of executable specification written in the NuSMV input language. The desired security properties of the system are also encoded using formal languages understood by model checkers (e.g., Linear Temporal Logic (LTL) or Computational Tree Logic (CTL)). Finally, both the asset-flow and the encoded security goals are automatically analysed by a model checker which is able to identify potential violations of the defined security properties. Such violations are expressed as counterexamples which are potential process sequences that can compromise the security of assets. Such counterexamples can, thus, be used as the input for a new iteration of the risk management process.

The focus of [106] is on the alignment and integration of risk management (RM) elements in business process modelling, in order to facilitate decision making based on the risk assessment of the cloud-based process under development. The main stakeholders required for this are the cloud consumer, the cloud provider and the cloud broker, the latter being an emerging role acting as an intermediary between the other two. According to this work the cloud broker matches the consumer's process to the cloud provider better equipped to fulfil its security needs, in order for a risk-aware business process to be constructed and securely deployed to the cloud. Once a suitable cloud provider is selected and final adjustments are made to the process and the infrastructure supporting it, the identified risks of the process are evaluated for their effective treatment. If the risks are treated at a level deemed satisfactory by the cloud consumer ("risk acceptance") then the business process is ready to be deployed to the cloud.

In [107] an extension to BPMN is introduced aimed at risk handling. This work aims to improve the specification of risks at BPMN processes which by

then was performed through textual annotations (e.g., "error events"), therefore lacking in clarity and precision. In order to improve this aspect, they introduce a new modelling element called "Risk Factor" which categorises identified risks in terms of risk type and quantifies their likelihood and impact in a five point scale. Each risk type is also represented at the process model via distinct notation. Additionally, the concept of "Risk Handler" is introduced, representing a risk mitigation method for handling the identified risks of a business process (i.e., reduce, retain, avoid, transfer, exploit or ignore risk).

Another attempt in the area of risk management in the context of business process modelling is presented in [108]. As the authors claim, this work does not attempt to introduce yet another extension, but rather semantically align the well-established, security related concepts of the ISSRM (Information Systems Security Risk Management) domain model with the already existing notation of the latest version of BPMN (v2.0). This alignment attempt aims to explore how security concerns can be annotated, and security requirements defined by business activities modelled by BPMN, and how can BPMN, through the illustration of potential risks, facilitate the reasoning about the defined security requirements. A mapping between ISSRM concepts (e.g., asset, threat, risk, impact) and the BPMN constructs used to express them, is attempted through a running example of a business process modelled in BPMN, where a number of potential security risks (e.g., confidentiality, integrity etc.) have been identified and appropriate countermeasures have been added. The potential risks and the appropriate security requirements are identified at the process level by matching process fragments with security-risk patterns used to capture common security requirements (e.g., confidentiality, integrity, availability). Such patterns have been defined and classified in [109]. In [110] the same authors introduce SREBP, a holistic method to manage security risks in business process models by combining the ISSRM and BPMN concept alignments, the defined security-risk patterns and the process model security annotation approach. In [111], [112] SREBP is enriched with the application of the enterprise model frame, which is based on the ArchiMate modeling language in order to directly relate enterprise architecture elements to specific BPMN elements.

A similar attempt is presented in [113] where the authors explore how threats can be described in business process models by using the capabilities offered by the latest version of BPMN (v2.0). The need for such research was motivated by the fact that the new capabilities offered by the latest version of BPMN have not received much attention concerning possible security or risk related extensions.

According to this work threats can be modelled by special types of events which may result in a deviation from the standard flow of the business process. Error and escalation types of events, already existing constructs of BPMN, can be used for such purposes in collaboration diagrams. For higher abstraction types of BPMN diagrams (i.e, conversation and choreography diagrams) it is more practical to represent threats in the form of textual annotations. As observed by this work, BPMN already has a wide range of constructs, so no extension is necessary for the representation of threats. Nevertheless, the expression of threats via events in collaboration diagrams can increase their complexity thereby decreasing their degree of comprehension. Additionally, this approach for threat representation, focuses only on the potential effects of threats at the workflow level of the process and does not deal with the calculation of their impact or possibility, which is left to traditional risk assessment frameworks.

Finally, the work of [114] presents a technique to model threat patterns which can be used for threat identification in business process models. The technique is based on the transformation of normal scenarios, captured by UML sequence diagrams, to negative scenarios where a threat can be realised by a mis-actor using a threat pattern rule. These patterns are captured by the creation of UML threat profiles based on information collected by different international standards (e.g., Common Criteria).

2.3 Evaluation

An important aspect of the analysis, supported by the works identified through this literature review, is the extent of the coverage they provide. The cover- age of the supported analysis can be evaluated in two ways, namely coverage of security- and risk-related aspects and coverage of different abstraction levels (organisational, operational and implementation level).

The first analysis criterion is the coverage provided for security- and risk-related aspects, in more detail:

- the *security* analysis aspect, covers the elicitation of security requirements (e.g., confidentiality, integrity, availability) at the organisational level, security policies or security-annotated activities at the process level and security related services at the level of the implementation. Privacy concerns are also included in this category as they are often grouped together with security related aspects in literature.

- the *risk* analysis aspect is concerned with the identification of threats and the analysis of risks introduced by them at the organisational and operational levels, as well as risk mitigating solutions at the implementation level.

Concerning the different levels of abstraction where analysis is supported, we differentiate between:

- the *organisational level*, where concepts such as goals, actors and resources can be captured using goals models,
- the *operational level*, where sequences of activities performed by different actors can be captured by means of business process modelling,
- the *implementation level*, where the components of process models are matched or assigned to services or other execution level artefacts (e.g., code).

All works identified through this literature review have been categorised according to the above criteria as presented in Fig. 2.1. Each circle represents an abstraction level and so works placed within the intersection of two or more circles provide support at multiple abstraction levels. Moreover, works appearing in black lettering support security-related analysis, underlined works support risk-based analysis, and works appearing in blue and underlined lettering support both aspects of analysis.

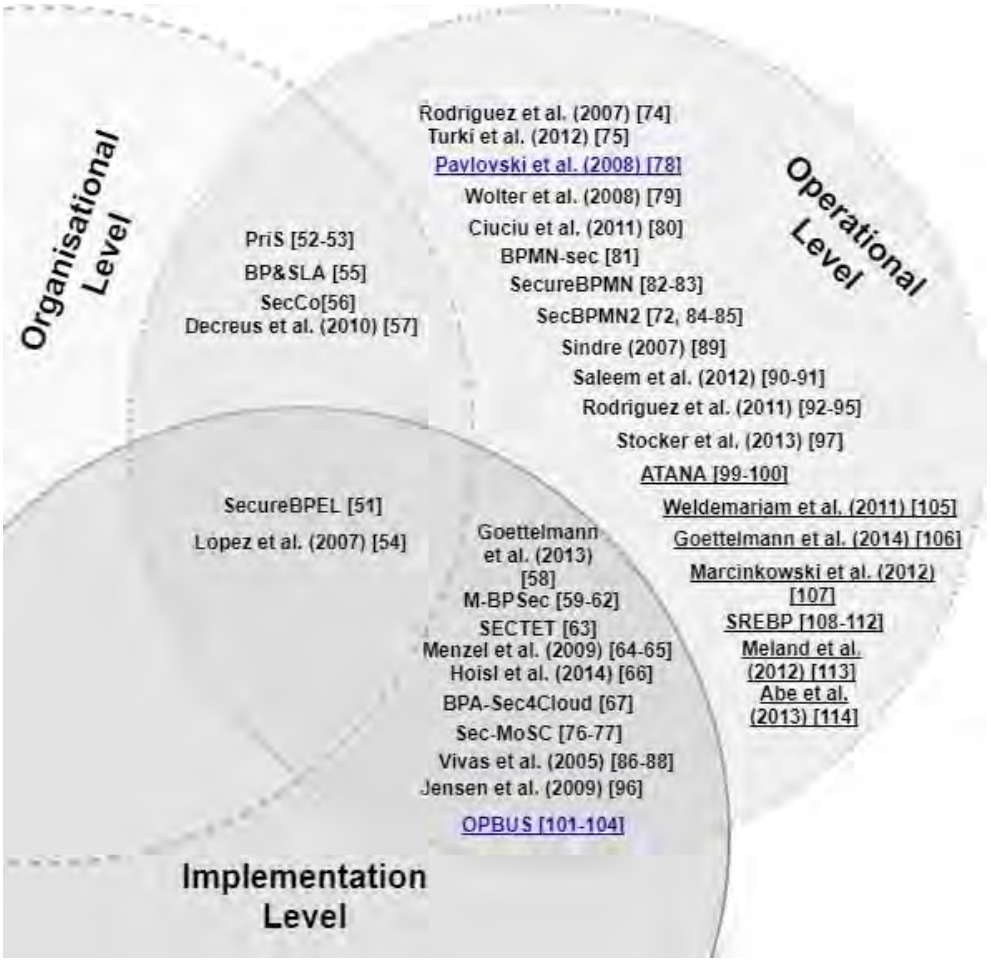

Figure 2.1: Security- and risk-related analysis support by level of abstraction

It is apparent from Fig. 2.1 that only a small number of the identified approaches ([51], [54]) are able to support system analysis throughout all the different levels of abstraction. The majority of the identified works focus on the operational level as they solely offer support for analysis of business process models. Another useful insight from the taxonomy presented at Fig. 2.1 is that the transition from organisational level models (e.g., goal models, UML diagrams) to the operational level (i.e., business process models) is much less represented in the literature of the area when compared to the transition from process models to implementation level artefacts (e.g., service compositions).

Regarding the coverage of the different concerns grouped under the umbrella of information security, Fig. 2.1 reveals that only a small amount of identified works ([78],[101]–[104], [115]) are able to holistically consider all different aspects of analysis (i.e., security and, risk). Instead most approaches specialise in one type of analysis, with security analysis being the most represented in the identified work.

	Requirement Elicitation	Threat Modelling	Countermeasure Elicitation
[51]	✓		
[52], [53]	✓		✓
[54]	✓		
[55]	✓		
[56]	✓		
[57]	✓		

Table 2.2: Requirements analysis support of organisational level approaches

Table 2.2 summarises the type of analysis provided by the identified approaches which offer organisational level modelling capabilities. Since the organisational level captures the highest level of abstraction, the works listed in the table provide a goal modelling component used to elicit security requirements. For the requirements elicitation process, concepts such as goals, constraints and policies can be used to identify high level security requirements which can later be incorporated into the produced process models at the operational level. While the elicitation of security requirements is the main purpose of the approaches listed at Tab. 2.2, literature suggests that it is also highly important and beneficial for them to be able to also incorporate concepts able to capture risk related aspects and mechanisms or countermeasures which deal with the identified security requirements [116]. Therefore, we have included *"Threat Modelling"* and *"Countermeasure Elicitation"* as criteria in our evaluation of works with an organisational modelling component, as presented in Tab. 2.2.

Furthermore, Tab. 2.2 indicates that none of the identified approaches is flexible enough at the organisational level to provide coverage for the combination of threat modelling and countermeasure elicitation. More importantly the support for modelling risk related aspects is absent from all the identified approaches while the elicitation of countermeasures is only included as part of the PriS framework [52], [53] which specialises in the aspect of privacy and offers a series of suggested privacy enhancing technologies matched to specific types of privacy requirements.

To evaluate the operational level modelling support of the identified approaches, an overview of which is provided in Tab. 2.3, a number of criteria have been introduced. The mapping of process activities to elements of the organisational level requirements model (i.e., goals) is considered a valuable practice as it augments the traceability of changes between system models of different abstraction levels [57]. Additionally, since process models are not equipped to

adequately capture the rationale behind design choices, mapping their components to requirement models helps provide justification. As a result, the ability to map process elements to organisational level artefacts is on of the criteria used for the evaluation of the approaches included in Tab. 2.3.

Another important criterion is the introduction of new sets of notation at the process model level in order to visually communicate security and/or risk related concepts. Modelling such concepts into business process models in the form of notation facilitates model comprehension by stakeholders of different domains and fosters collaboration [3]. Additionally, it is also beneficial that new sets of notation are expressive enough so they can fully capture all the different aspects of analysis (i.e., security- and risk-concepts). Otherwise, more than one approaches, complementary to each other, may need to be applied at the same process model, thus introducing considerable overhead and complexity.

As illustrated in Tab. 2.3, none of the identified process modelling approaches satisfies all the criteria previously discussed. In terms of traceability between requirements and business process models, frameworks with an organisational modelling component (see Tab. 2.2) can provide concept mappings between goals and process level activities. On the other hand, most of the identified works do not perform requirements elicitation at the organisational level and therefore are limited to simple security and risk-related annotation of process models.

Regarding the process annotation capabilities of the approaches identified in this review, only the works of [78] and [108]–[112] introduce annotations capable of capturing both security and risk related concerns. The vast majority of the identified works focus mainly on security related annotation, either introducing new symbols to mark security constrained process activities (e.g., padlock symbols) or use textual annotations to denote security concerns. A smaller set of works introduce similar types of annotations but focused specifically on risk modelling. Therefore, a number of specialised approaches exist which support the analysis of individual aspects of security and risk but only a small number of works is able to cover both aspects.

	Mapping to Org. Goals	Security/Privacy Annotation	Threat/Risk Annotation
[51]	✓		
[52], [53]	✓		
[54]	✓		
[55]	✓		
[56]	✓	✓	
[57]	✓		
[59]–[62]		✓	
[64], [65]		✓	
[67]		✓	
[74]		✓	
[75]		✓	
[76], [77]		✓	
[78]		✓	✓
[79]		✓	
[80]		✓	
[81]		✓	
[82], [83]		✓	
[72], [84], [85]		✓	
[92]–[94]		✓	
[96]		✓	
[97]		✓	
[101]–[104]			✓
[105]			✓
[107]			✓
[108]–[112]		✓	✓
[113]			✓

Table 2.3: Process security modelling support of identified approaches

Since our analysis focuses on model-driven approaches in the context of business process security, an important factor that needs to be considered is the representational support provided by the identified works. The successful representation of business processes via business process models requires a set of explicit steps to be followed for the creation of such models, notation capable of capturing the main concepts necessary for their analysis (i.e., security and risk) and a platform that supports all the above and facilitates model development. By combining the guidance provided by rules and the expressiveness provided by domain-specific notation, with the ease-of-use offered by support tools (e.g., design platforms), the design process can be streamlined and large parts of it can be automated. The importance of an automated approach for the derivation of business processes based on the overall business goals of an organisation, has been identified as an important direction for future research in the area of business process modelling, as it enhances the usability and reduces the amount of effort required [116]. Therefore, to evaluate the representational support of the identified approaches, three evaluation criteria have been introduced in Tab. 2.4 to represent the need for design steps, additional notation and tool support.

Table 2.4 indicates that only a limited number of the identified approaches satisfies all three aspects of representational support. Most works introduce new concepts or notation to capture security and risk-related aspects in process models but only a few also develop modelling tools capable of supporting the creation of process models with the newly introduced notation. The same can be observed for the existence of specific sets of rules or steps to support the creation of process models, as the identified works usually introduce sets of notation but, either do not specify specific steps to be followed or leave the design process to the discretion of the involved stakeholders.

	Design Steps/ Rules	Additional Concepts/ Notation	Tool Support
[51]			✓
[52], [53]	✓		
[54]	✓		
[55]	✓		✓
[56]	✓		
[57]	✓		✓
[58]	✓		✓
[59]–[62]	✓	✓	✓
[63]			✓
[64], [65]		✓	✓
[66]	✓		✓
[67]		✓	✓
[74]		✓	
[75]	✓	✓	✓
[76], [77]	✓	✓	✓
[78]		✓	
[79]		✓	
[80]	✓	✓	
[81]		✓	
[82], [83]		✓	✓
[72], [84], [85]		✓	✓
[89]		✓	
[90], [91]		✓	
[92]–[94]	✓	✓	
[96]		✓	✓
[97]		✓	
[101]–[104]		✓	✓
[105]		✓	
[106]	✓		
[107]		✓	
[108]–[112]		✓	

Table 2.4: Representational support of identified approaches

2.4 Research Gaps and Challenges

From the evaluation of the works identified via the literature review performed in this chapter the following research gaps and challenges in the area of secure business process model design can be identified:

1. *Ch.1: Need for holistic security analysis.* Information security encompasses a multitude of aspects which can be categorised under confidentiality, integrity, availability, authenticity, accountability and non-repudiation [117]. Nevertheless, there are other relevant aspects related to information security that need to be reflected at the process level such as authentication, authorisation as well as privacy-related aspects [118]. Similarly, the inclusion of risk-related aspects further enhances the analysis of secure business process, as they allow to capture potential threats, evaluate their impact and propose mitigating configurations. Therefore, all aspects of security and risk need to be taken into account in order to holistically analyse security during the design of business processes. While our review of the related literature identified a variety of attempts to address individual aspects of security and risk, works that support the holistic analysis of such areas are in short supply.

2. *Ch.2: Support for analysis at multiple levels of abstraction.* As previously discussed, different levels of abstraction are able to capture different aspects of security analysis. In order to understand and represent user requirements in terms of enabling organisational strategy to encompass business needs, we need to be able to describe the context of the system. The goals which an organisation aims to achieve by the execution of its business processes can provide highly relevant input during the systems design phase. Goal-oriented requirements engineering (GORE) approaches use goals to capture the rationale behind design-time decisions at the organisational level of abstraction. Therefore, when paired with process modelling approaches, they are a useful initial tool for the design of the business processes [5]. Next, at the operational level of abstraction, business process models are capable of capturing a great level of detail in regards to the flow of activities and information and resource exchanges between the participating stakeholders. At the operational level security implementing technologies, introduced in response to security constraints identified at the higher level of abstraction via goal models, can now be mapped onto specific process activities, thus facilitating the analysis of security at the process level. Finally, at

the lowest level of system analysis abstraction, the implementation level, process activities can be matched to specific IT systems and services which are capable of implementing their functionality. Nevertheless, the identification of processes and their matching to services, which should ideally take place early during the system's development, is a very important and yet challenging and not well-understood activity [119]. Therefore, this propagation of security analysis through the different levels of abstraction, from high level organisational strategy to low level services and security implementing technologies, allows for a seamless transition from abstract security requirements to specific security configurations. It is, consequently, a noteworthy approach for the design of secure business process and as such it should be further studied by researchers of the area.

3. *Ch.3: Ability to identify threats and elicit countermeasures during requirements analysis.* For a more comprehensive analysis of the different aspects of security at the organisational level it is important that, besides the elicitation of security requirements, threats are identified and countermeasures are elicited. It is considered highly beneficial to incorporate such aspects of analysis at the requirements level [116], since their early identification and inclusions at the analysis provides a more comprehensive view of the systems security. Nevertheless, the evaluation of current approaches for the design of secure business process which include a goal-oriented security requirements component as a starting point, revealed very limited adoption of threat identification and countermeasure elicitation. Thus, future attempts in this research area should consider extending their analysis capabilities at the requirements level to accommodate such aspects, as involving them early during the analysis allows for a more accurate representation of security for the system to-be.

4. *Ch.4: Decision support capabilities throughout the design process.* As already discussed, the importance of connecting operational level elements with high-level goals bolsters the alignment between strategy and operations. In the context of security, linking specific process components with security constraints, introduced at the organisational level, allows the provision of rationale for design choices at the business process model level. Therefore, it is preferable for approaches in the area of secure business process modelling to not only provide the necessary notation to annotate all aspects of security (requirements, threats, mechanisms, countermeasures

etc.), but also link the design choices to specific goals to provide reasoning. Once security-constraint parts of the process have been identified, annotated and mapped to specific organisational goals and/or security requirements, decisions regarding the inclusion of security-implementing process activities need to be taken at the business process model. Therefore, decision support should also be facilitated at the operational level of analysis, allowing reasoning about security configurations based on properties of the business process model (e.g., complexity of the workflow, cost of implementation).

5. *Ch.5: Well-definedness and automation of design process.* The design of business process models can be a demanding and time consuming endeavour, especially as the scale of the modelled systems grows. The considerable amount of effort required for such a process can be significantly reduced if a well-defined series of steps and/or rules guiding the design of secure business process models exists. Another aspect which adds to the complexity of the design process is the different security-oriented notations introduced on top of the standards notation of graphical process modelling standards. Ad-hoc sets of notation with no explicit definitions introduced by most of the current approaches to security-oriented business process modelling, often overwhelm the stakeholders as they require effort and domain-specific knowledge to be fully comprehended [3], [15]. Therefore, intuitive and explicitly defined security related notation can greatly improve the quality and readability of the produced models and further reduce the effort required during the design stages. Finally, automated tool support for the construction and analysis of business process models improves the applicability of approaches for secure business process modelling [116] as it can easily facilitate the application of well defined design steps, analysis rules and explicit notation. Thus, the focus of future attempts in the area of model-driven business process modelling should be the creation of well-defined approaches, supported by software tools in order to improve the modelling experience.

Therefore, the output of the research project presented in this work is motivated by the research gaps and challenges identified through the analysis of the literature of the area. The developed framework for the design of secure business processes, presented in Chapter 3, builds on existing approaches and modelling languages and introduces new components and artefacts in order to address the identified challenges.

Chapter 3

Proposed Framework

The framework presented in this work is developed to assist in the creation secure business process designs sourcing from high level stakeholder requirements. More specifically, the final output resulting from the application of this framework will be a business process model which will contain both functional and security implementing activities. Throughout the application of each of the framework's components, a variety of stakeholders are to be involved, each providing different types of input and executing relevant modelling and analysis activities.

During the initial stages of the framework application, the input of high-level organisational stakeholders (e.g., upper management, consultants) is required for the identification of the top-level strategic goals the system under development should accomplish. Such system objectives are captured via goal models, which constitute the main initial artefact around which the analysis supported by this framework is structured. In addition to the above mentioned stakeholders, information security analysts are also involved in the initial stages of the framework's application in order to identify the system's main security-related objectives using goal-oriented security requirements engineering. Through the propagation of such analysis facilitated by security-oriented goal models, the security constraints, threats and security implementing activities of the system to-be are identified early during its development lifecycle and connected to its strategic objectives, making security an important cornerstone around which the business process supporting system to-be will be developed.

Next, the goal model, capturing participating actors, their goals, tasks, resources and security concerns is utilised as a means of automatically producing a business process skeleton via a set of model transformation rules provided by the proposed framework. Business process analysts and designers can utilise the automatically generated business process skeleton and refine it into a complete

and secure business process model. During this step some final design choices are made by business process designers in collaborations with security engineers regarding the operationalisation of the systems security, guided by the framework's comprehensive decision support component. The created business process model can also be verified by the same stakeholders, in regards to the satisfaction of the initially identified security requirements, using the framework's verification component.

Framework Contributions	Research challenge see Section 2.4
(i) Support for the elicitation and operationalition of all aspects of security requirements.	[Ch.1, Ch.3]
(ii) Alignment between high-level goals and process-level configurations.	[Ch.2]
(iii) Seamless transition between different abstraction-level models via explicit mappings and model transformation rules.	[Ch2, Ch.5]
(iv) Support for stakeholder input during decision making both at the organisational and operational level.	[Ch.3, Ch.4]
(v) An adaptable approach to process model instantiation, where a number of similar but slightly different process designs can be derived from the same reference model, according to the specific situational needs of each implementation.	[Ch.4, Ch.5]
(vi) A set of preconfigured security-implementing process fragments that guide the operationalisation of security at the business process level in a structured manner.	[Ch.5]
(vii) Business process security verification capabilities via a structured, attribute-based approach, to identify potential security shortcomings of the produced business process model.	[Ch.5]
(viii) Software tool support to assist and automate the application of the framework's components.	[Ch.5]

Table 3.1: Framework contributions mapped to identified research challenges

According to our findings from the literature review, presented in Section 2.4, a number of research challenges have been identified in the area of secure business

process design. The developed framework aims to work towards tackling the identified research challenges by contributions presented in Tab. 3.

The above contributions highlight the information security orientation of the framework that will be presented in this chapter. Nevertheless, there are adjacent concepts that are often considered along with information security, such as privacy and trust. The modularity provided by the components of the proposed framework could allow the consideration of privacy and trust. To achieve that, the components could be extended to include concepts that could allow the elaboration of such aspects without significantly altering the overall functionality of the rest of the framework. Nonetheless, potential conflicts between security and privacy or trust would also need to be identified, analysed and resolved. Since that would significantly increase the scope of this work and add considerable overhead to the framework's application it has not been considered in the context of this project. Therefore, information security shall be the central concern of the proposed framework but the potential for its further extension to cover security-adjacent aspects is recognised and its implications to the quality and completeness of the framework's outcome are discussed in the Conclusion chapter (see Chapter 5).

The rest of this chapter focuses on presenting the different building blocks of the proposed framework. First a general overview of the framework's components and activities will be presented in Section 3.1. Next each component will be individually introduced and discussed. A running example will also be used throughout the presentation of each component to provide a proof-of-concept of the application of the proposed framework to a real life system.

3.1 Framework Overview

The main building blocks of each component and its interconnection with the rest of the proposed framework are presented in Fig. 3.1. The blue nodes represent the main modelling artefacts produced throughout the frameworks application. The grey nodes represent the building blocks utilised by each component to support the creation of each modelling artefact. Furthermore, a high level overview of the sequence of activities performed by each component during the framework's application, is presented in Fig. 3.2. A more detailed breakdown of each component's activities, inputs and outputs will be individually presented at each component's corresponding section within this chapter.

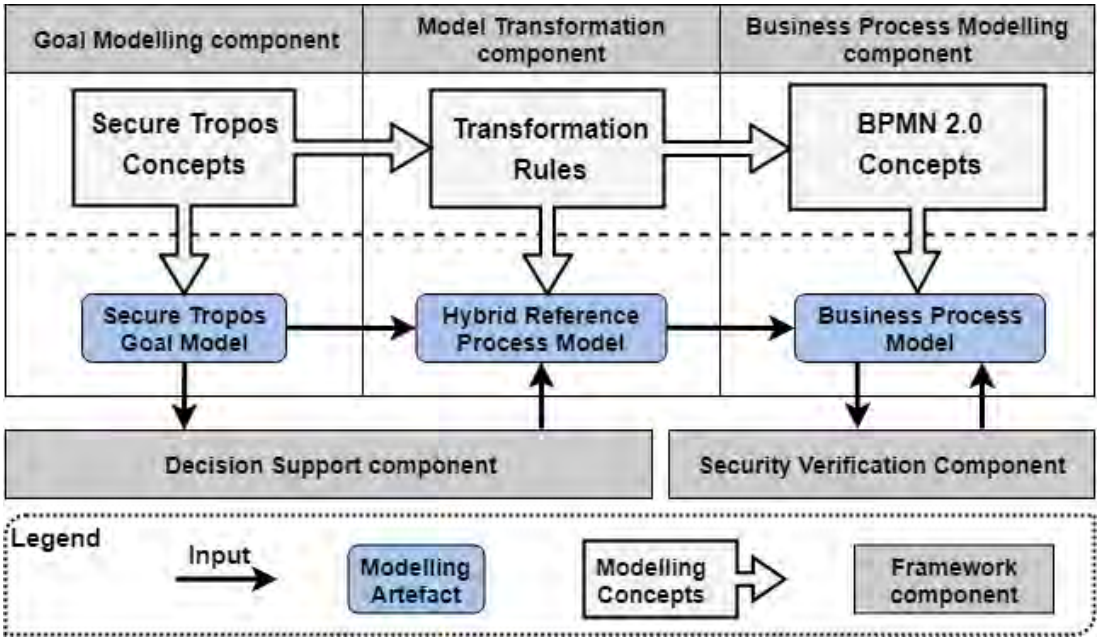

Figure 3.1: Components of proposed framework

The *Goal Modelling component* is concerned with capturing the organisational structure, strategy, and security concerns at a high level of abstraction via the use of goal models by high-level organisational stakeholders (e.g., management, consultants). At the same time it provides input, in the form of non-functional system characteristics and potential security implementing technologies by security experts, to support the decision making during the later stages of business process design.

The *Decision Support component* provides a structured approach to system designers and security experts for deciding the security composition of the system to-be. Through this component, security, risk and non-functional aspects of the system can be quantitatively defined and evaluated. Satisfiability solvers are then utilised for the identification of system compositions which best fit the identified parameters. Based on the output of this component, the security implementation of the system to-be can be identified and later be operationalised by the produced business process model.

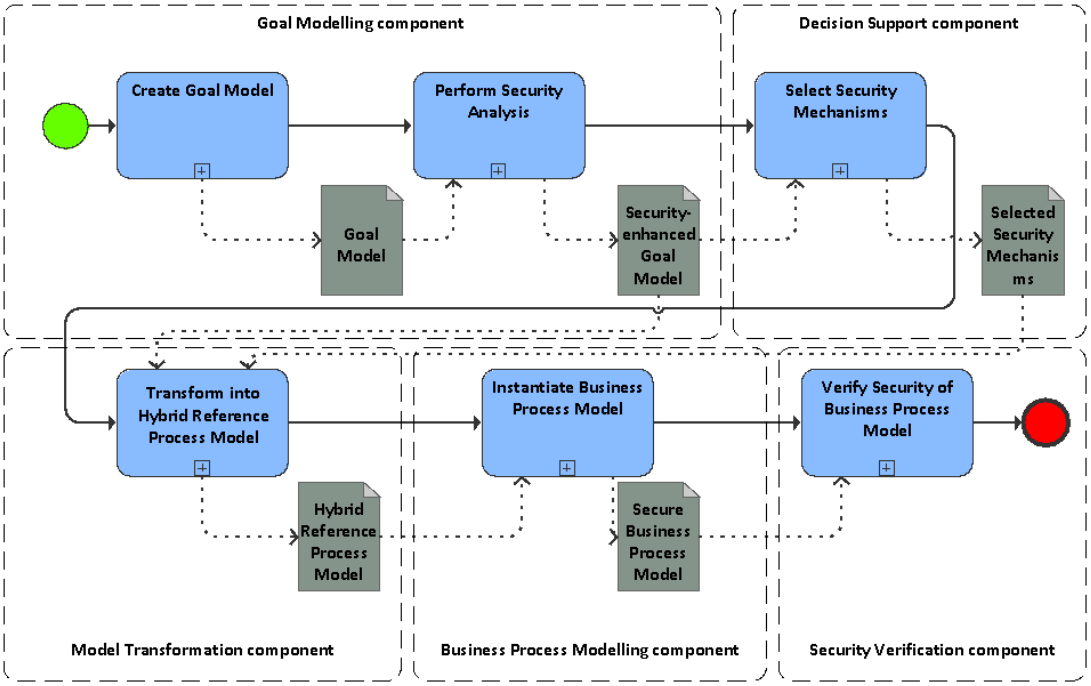

Figure 3.2: Proposed framework overview

The *Model Transformation component* is utilised for translating the organisational level concerns captured by the goal modelling component, to the operational level of abstraction. Therefore, this component links goal and business process modelling concepts and uses this mapping to extract transformation rules. These rules are then used to produce the hybrid reference process model from the goal model. The hybrid reference process model uses both goal and business process modelling concepts to create a process skeleton that encompasses the information captured by the goal model diagram.

The hybrid reference process model is, therefore, the main artefact used by process designers for the definition of the framework's final deliverable, the secure business process model by the *Business Process Modelling component*. This component contains a library of process patterns, which are used to operationalise the different security-implementing mechanisms identified at the goal modelling level and selected using the *Decision Support component*. The business process model skeleton, automatically created and captured by the hybrid reference business process model, is manually refined to a complete BPMN 2.0 collaboration diagram.

Finally, the *Security Verification component* evaluates the degree of satisfaction of the system's security requirements by the created business process model.

To achieve that a number of relevant security related attributes are introduced at the business process level which can be evaluated by security checking algorithms to identify potential violations of the system's security constraints.

In addition to the contribution of each component to the overall functionality of the framework, most of the components can also be used independently of each other to achieve smaller specifiable goals. The publications produced during the development of the framework, listed in Section 1.6, include cases of individual component applications. More specifically, in [8] the model transformation component is used a standalone artefact for transforming Secure Tropos goal models to business process skeletons. In [40], [44] the business process modelling component is used as a structured approach towards the security instantiation of business process models using security patterns. In [43], [45] the decision support component is introduced as a means of optimising the security configuration selection of information systems. In [41] the security verification component is utilised in order to verify the security properties of existing business process models.

Furthermore, combinations of framework components have be used in conjunction with other approaches to a variety of areas of interest. For instance, in [9], parts of the framework have been used for eliciting security requirements for legacy business processes. The framework has also been utilised for the creation of business processes for software product lines in [37], and the design of secure cloud-based information systems in [38].

3.2 Goal Modelling component

Secure Tropos [6] is a security-oriented extension of Tropos [25], a goal-oriented requirements engineering method. The main motivation behind the creation of Secure Tropos was the lack of a methodology to support the capturing, analysis and reasoning of security requirements from the early stages of the development process. As such, Secure Tropos, combines concepts from requirements engineering for representing general concepts and security engineering for representing security-oriented concepts, which are presented in detail in [35].

The creation of security-oriented goal models for the elicitation of requirements, threats and implementation mechanism alternatives for the system to-be is the starting point of the framework proposed by this work. The ability of Secure Tropos to capture and analyse such concepts in an explicit and structured manner is the main reason for its selection as the method of choice for performing the organisational level modelling required by our framework. More

specifically, the advantages of Secure Tropos, compared to other security-oriented GORE approaches are:

i. its ability to perform social analysis during the early requirements stage by capturing actors, their goals, resources and interdependencies,

ii. the simultaneous consideration of security along with the other requirements of the system-to-be, via the provision of a number of different modelling views, each capturing different aspects of the system's design (e.g., organisational view, security requirements view, security attacks view).

iii. the support for not only the requirements but also the design stages of the development lifecycle, through the mapping of abstract security constraints and threats to specific implementation mechanism alternatives.

3.2.1 Goal Modelling Concepts

The subset of Secure Tropos concepts, as introduced in [35], used for the organisational level analysis included in our proposed framework are listed below.

- A *Goal* represents a condition in the world that an actor would like to achieve [120]. In other words, goals represent the strategic interests of actors. In Tropos, the concept of a hard-goal (simply goal hereafter) is differentiated from the concept of soft-goal. A *Soft Goal* is used to capture non-functional requirements of the system, and unlike a (hard) goal, it does not have clear criteria for deciding whether it is satisfied or not and therefore it is subject to interpretation [120] (e.g., the system should be scalable).

- An *Actor* represents an entity that has intentionality and strategic goals within the multiagent system or within its organisational setting [120]. An actor can be human, a system, or an organisation.

- A *Resource* presents a physical or informational entity that one of the actors requires [25]. The main concern when dealing with resources is whether the resource is available and who is responsible for its delivery.

- A *Plan* represents, at an abstract level, a way of doing something [25]. The fulfilment of a plan can be a means for satisfying a goal, or for contributing towards the satisfying of a soft goal. In Tropos different (alternative) plans, that actors might employ to achieve their goals, are modelled. Therefore developers can reason about the different ways that actors can achieve their goals and decide the best possible implementation.

- A *Dependency* between two actors represents that one actor depends on the other to attain some goal, execute a task, or deliver a resource [120]. The depending actor is called the depender and the actor who is depended upon is called the dependee. The type of the dependency describes the nature of an agreement (called dependum) between dependee and depender. Goal dependencies represent delegation of responsibility for fulfilling a goal. Soft-goal dependencies are similar to goal dependencies, but their fulfilment cannot be defined precisely whereas task dependencies are used in situations where the dependee is required to perform a given activity. Resource dependencies require the dependee to provide a resource to the depender. By depending on the dependee for the dependum, the depender is able to achieve goals that it is otherwise unable to achieve on their own, or not as easily or not as well [120]. On the other hand, the depender becomes vulnerable, since if the dependee fails to deliver the dependum, the depender is affected in their aim to achieve their goals.

- A *Security Constraint* is the main concept introduced by Secure Tropos. Security Constraints are used, in the Secure Tropos methodology, to represent security requirements [17]. A Security Constraint is a specialisation of the concept of constraint. In the context of software engineering, a constraint is usually defined as a restriction that can influence the analysis and design of a software system under development by restricting some alternative design solutions, by conflicting with some of the requirements of the system, or by refining some of the system's objectives. In other words, constraints can represent a set of restrictions that do not permit specific actions to be taken or prevent certain objectives from being achieved. Often constraints are integrated in the specification of existing textual descriptions. However, this approach can often lead to misunderstandings and an unclear definition of a constraint and its role in the development process. Consequently, this results in errors in the very early development stages that propagate to the later stages of the development process causing many problems when discovered; if they are discovered. Therefore, in the Secure Tropos modelling language security constraints are defined as a separate concept. To this end, the concept of security constraint has been defined within the context of Secure Tropos as: *A security condition imposed to an actor that restricts achievement of an actor's goals, execution of plans or availability of resources*. Security constraints are outside the control of an actor. This means that, in contrast to goals, security constraints are not conditions

that an actor wishes to introduce but rather is forced to adhere to. Security constraints can also be grouped according to the security objective the achievement of which they contribute towards. Security objectives are broader descriptions of security principles or rules such as confidentiality, integrity, availability, authentication and authorisation.

- A *Threat* represents circumstances that have the potential to cause loss; or a problem that can put in danger the security features of the system [121]. Threats can be operationalised by different attack methods, each exploiting a number of system vulnerabilities.

- *Security Mechanisms* represent standard security methods for helping towards the satisfaction of the security objectives [17]. Some of these methods are able to prevent security attacks, whereas others are able only to detect security breaches. It must be noted that further analysis of some security mechanisms is required to allow developers to identify possible security implementations at a technical level.

One of the modelling views introduced by the Secure Tropos approach is the security requirements view, which provides a detailed analysis of the organisational view of the system under design. This view depicts node-link diagrams enclosed in circular containers representing system actors, with different types of nodes and connections to model both organisational and security related elements.

Figure 3.3: Legend of Secure Tropos concepts

Another modelling view of Secure Tropos utilised by this framework is the security attacks view, which provides further analysis of the threats identified at the security requirements view. A unique security attacks view is created for each of the identified threats which further illustrates how an attacker can harm the system at hand via the manifestation of the threat. More specifically, a series of

Attack Methods are identified for each threat, which represent the ways an attacker can utilise to harm the system (e.g., social engineering attack method for an information leak threat). Each attack method is linked to one or more system *Vulnerabilities*, which capture weakness of the designed system that each attack can exploit (e.g., unpatched equipment, insecure communication protocols). The identified vulnerabilities are linked to specific system components (i.e., goals, plans, resources) which can be directly compromised by each vulnerability. Additionally, each of the security mechanisms proposed at the security requirements view can be connected to a vulnerability to indicate whether it can protect the system against it. The Secure Tropos framework provides CASE tool support which accommodates both the creation of the described modelling views and the automated model analysis which is able to identify potential constraints and vulnerabilities for which countermeasures, in the form of security mechanisms, have not been identified. A legend of all the Secure Tropos concepts described in this section is presented in Fig.3.3. The relationships between the concepts included in the security requirements and security attacks view are captured at the partial Secure Tropos metamodel illustrated in Fig.3.4.

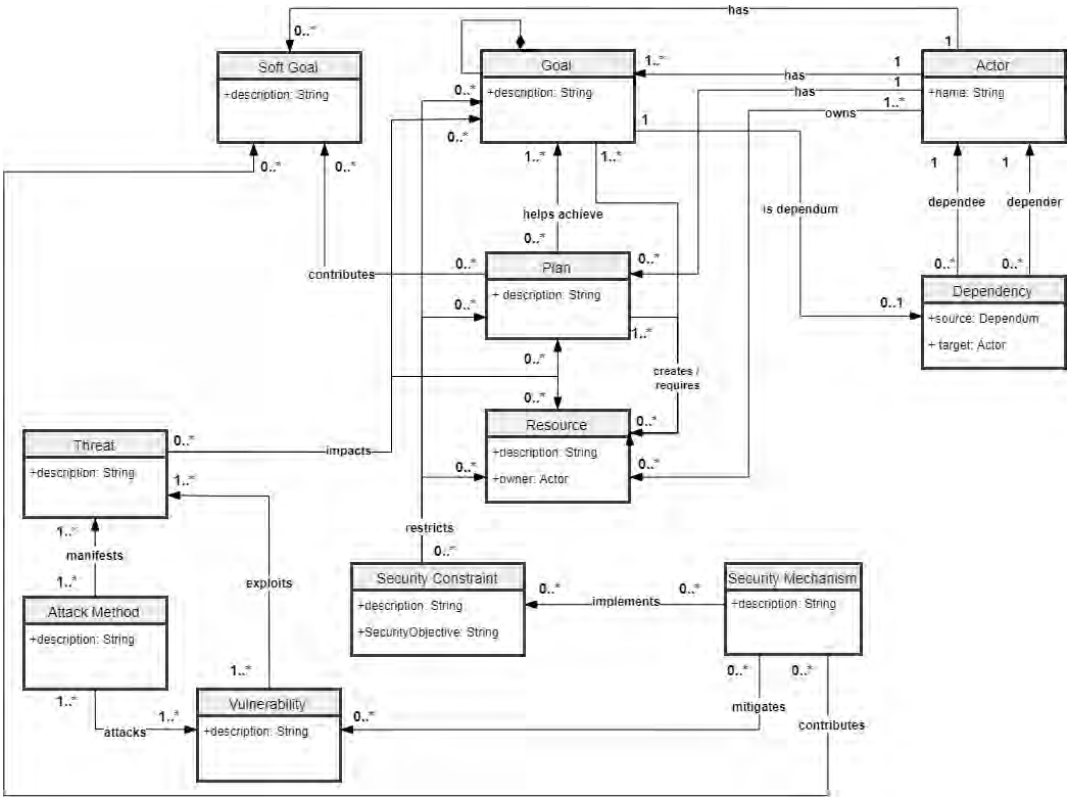

Figure 3.4: Partial metamodel of relevant Secure Tropos concepts

3.2.2 Goal Modelling Component Application

The sequence of activities performed as part of the Goal Modelling component application along with relevant inputs and outputs, are summarised in Fig. 3.5.

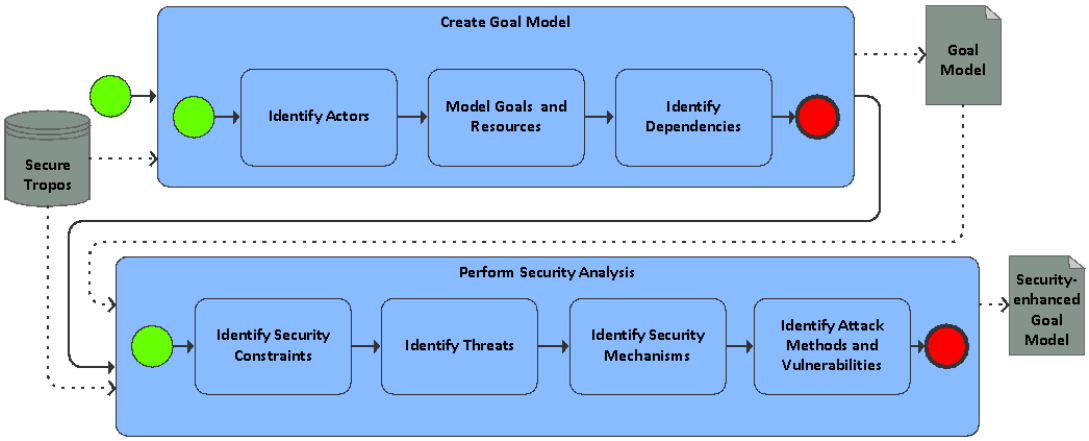

Figure 3.5: Activities for the application of the Goal Modelling component

An example of a security requirements view diagram is presented in Fig. 3.6. It illustrates the security requirements view diagram of an electronic prescription system, which will be used as a running example throughout this chapter to illustrate the application of the different components of our framework. The purpose of this system is to facilitate the creation and archiving of electronic prescription created by medical practitioners and used by patients to receive medication. The entities interacting within that system, namely the *"E-prescription system"*, the *"Medical Practitioner"* and the *"Patient"* are represented as actors, each of which has a set of goals that they are aiming to achieve by interacting with each other through dependency relationships. Their goals are decomposed to sub-goals and in some cases plans which represent simple activities each actor has to perform (e.g., *"Store new prescriptions"*).

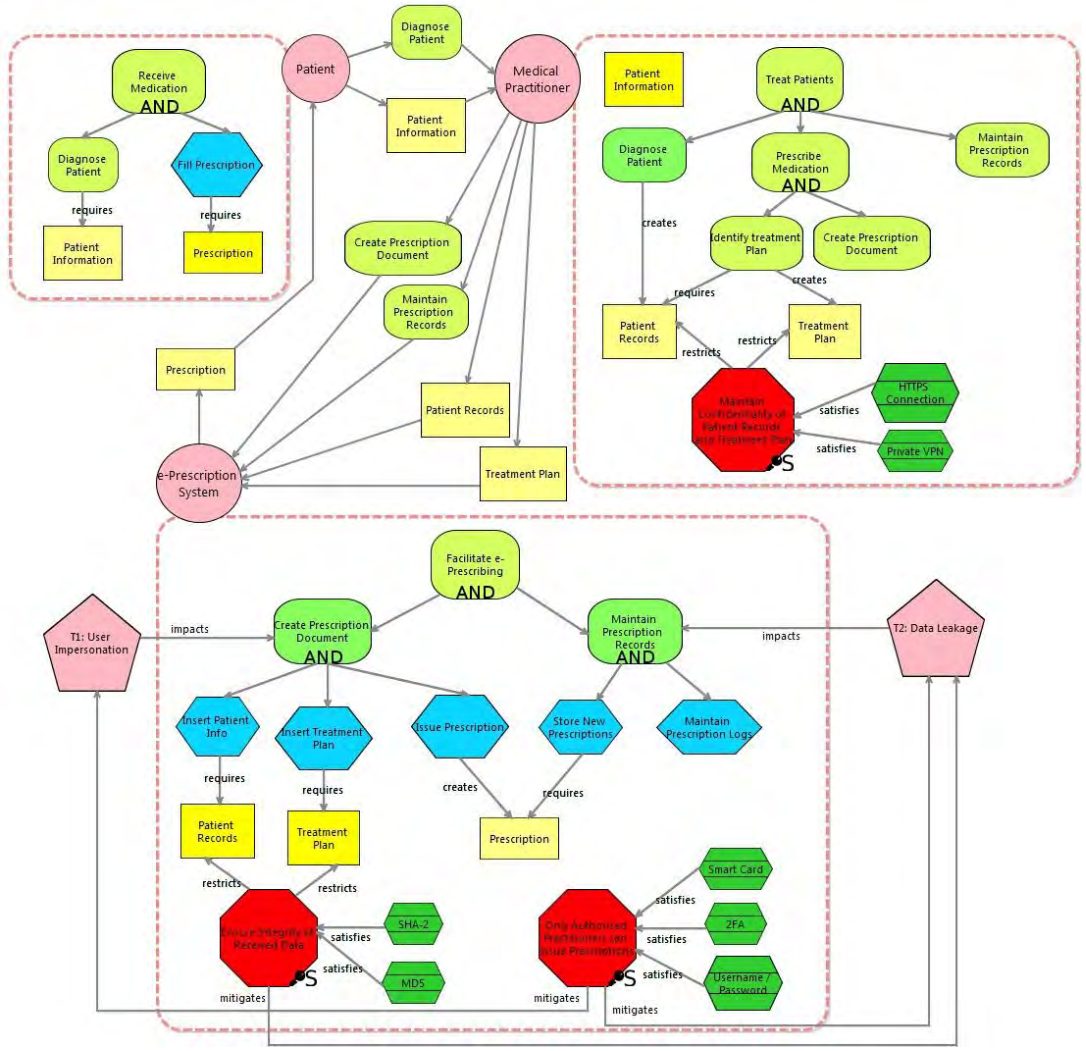

Figure 3.6: Security Requirements view model of e-Prescription system

The Patient's top-level goal is to *"Receive Medication"* and in order to achieve that depends on the Medical Practitioner through the goal *"Diagnose Patient"* and on the E-Prescription System for receiving the *"Prescription"* document. Similarly, the Medical Practitioner depends on the E-Prescription system for creating and storing prescription documents, modelled through goal *"Create new Prescription"* and *"Maintain Prescription Records"* and resource (*"Treatment Plan"* and *"Patient Records"*) and dependencies. Soft goals can also be identified to capture non-functional concerns for the system under design, for instance the soft goal *"Efficiency of Prescription creation"* aims to ensure that a new prescription document can be created by the least amount of actions possible by a medical practitioner.

Next, once the main actors, goals, resources and dependencies have been identified, the security requirements of the modelled system are to be identified. More specifically, security concerns are created and connected to goals and plans in order to restrict their functionality (e.g., *"Only authorised practitioners can create prescriptions"* categorised as an Authentication constraint). Threats are also identified and connected to entities they can impact. For instance, the *"User Impersonation"* threat in our model can impact the *"Create Prescription Document"* goal performed by the E-Prescription system. To achieve the systems security objectives and mitigate identified threats, a number of alternatives of security implementing mechanisms are introduced. For example the security Authentication-related constraint described above, can be satisfied by the implementation of either *"2-Factor Authentication"*, *"Smart Cards"* or *"Username and Password"*.

To further elaborate on the security aspects of the modelled system, Secure Tropos supports the creation of a Security Attacks view for each of the identified system threats. In our example the Security Attacks views for the *"User Impersonation"* and *"Data Leakage"* threats are presented in Figs. 3.7, 3.8. In those models, for each threat a number of Security Attacks are identified (e.g., *"Phising"* and *"Keylogging"* for the User Impersonation threat) and connected to system vulnerabilities they can exploit (e.g., "Compromised User Account"). The previously identified security mechanisms can then be connected to one or more vulnerabilities they can (fully or partially) protect against. Therefore, security and system analysts can have a better overview of potentially unprotected system vulnerabilities and reiterate their security analysis to propose better alternatives in terms of security mechanisms.

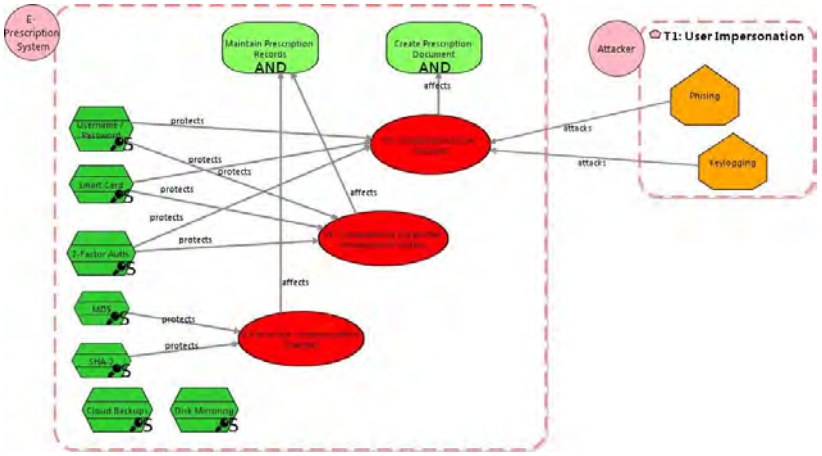

Figure 3.7: Security Attacks view of the User Impersonation threat

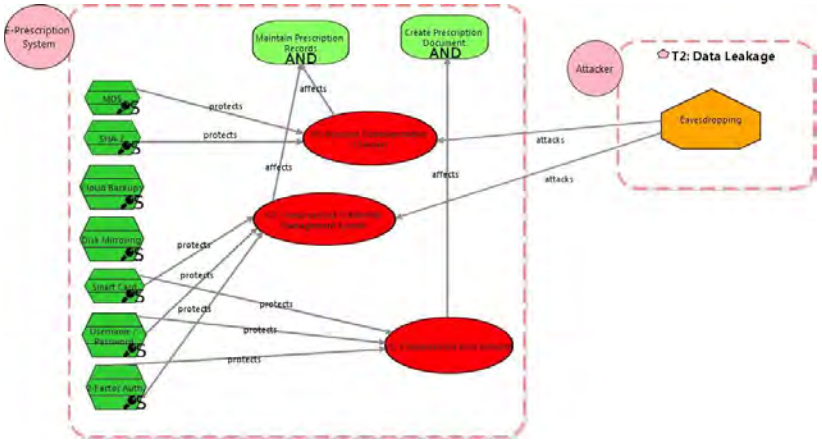

Figure 3.8: Security Attacks view of the Data Leakage threat

The Secure Tropos models created by the Goal Modelling component of the framework for the e-Prescription system will form the basis for the analysis provided by the Decision Support component, presented in Section 3.3. The relationships captured in those models provide valuable information regarding both the structure and the security coverage of the modelled system. The Decision Support component quantifies those relationships and, through an optimisation process, identifies the security mechanism combination best fitting the system's functional and non-functional needs.

3.3 Decision Support component

Before the transformation of the Secure Tropos goal model of the system to a BPMN business process model can take place, decisions have to be made regarding its security composition. More specifically, a combination of security mechanisms has to be selected from the different alternatives that have been previously introduced. The Decision Support component is introduced in this section, in order to support a structured and quantitative decision making process regarding the selection of security mechanisms best fitting the system's functional and non-functional goals.

Using the Decision Support component, different combinations of security mechanisms for each security-constraint goal, plan or resource can be selected according to the specific needs of the system at hand. The selection criteria influencing the final decision can be defined by the system stakeholders and designers and can capture a variety of security (e.g., risk reduction, constraint coverage) and non-functional aspects (e.g., cost, performance) of the system. To capture such aspects, a number of additional attributes are introduced to existing Secure Tropos concepts and constraint goal models (CGMs) are utilised to select the optimal configurations.

3.3.1 Risk-oriented Extension of Secure Tropos

Secure Tropos introduces a conceptual basis which facilitates security trade-off modelling and analysis [24]. An inherent limitation of all Tropos based approaches is their lack of precise semantics for the quantitative evaluation of system behaviours, including security and risk coverage [122]. Additionally, concepts necessary for the risk analysis process (e.g., risk) are missing. Attempts to align it with risk-related concepts have been developed [123], but they lack the ability to quantitatively perform risk assessment and support a fine-grained security trade-off analysis. To that end, we extend Secure Tropos with a number of concepts and attributes, as presented in Fig. 3.9 in bold lettering.

Risk Related Attributes

The concept of *Risk* is introduced into the existing Secure Tropos metamodel and connected to the concept of *Threat*, since any threat introduces a certain amount of risk through its associated *Vulnerabilities*. Each vulnerability represent a potential weakness that can be exploited by a threat and compromise the system's security.

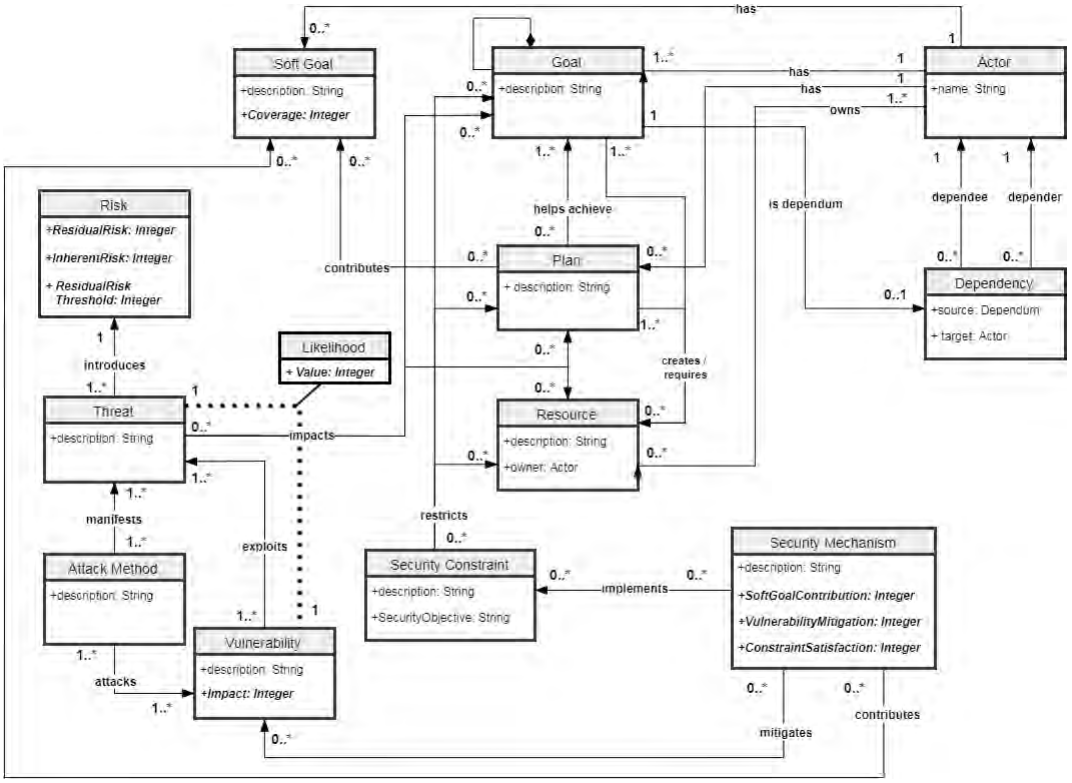

Figure 3.9: Metamodel of Risk-Oriented Secure Tropos Extension

The impact of each vulnerability is captured by the attribute *Impact* which can be evaluated using a number of different techniques. A common approach is estimating the impact of vulnerabilities using CVSS (Common Vulnerabilities Scoring System) [124] and/or historical data. A semi-quantitative scale is often used for value assignment of a vulnerabilities impact using discrete values (e.g., [10, 50,100] to represent low, medium, high impact) [125]. However, in this work we estimate the impact of a vulnerability as the relative impact with respect to that of all other vulnerabilities of the system. In other words, the higher the value of the impact the more important a vulnerability is. Therefore, to estimate the impact of each vulnerability we apply Analytic Hierarchy process (AHP) [126], [127], a common prioritisation approach in software engineering [128], [129].

The probability of a vulnerability being exploited for the manifestation of a security attack is captured by the *Likelihood* attribute. Similar to the estimation of a vulnerability's impact, likelihood in our work quantifies how much more probable is the exploitation of a vulnerability by a certain threat compared to another one. Therefore, likelihood represents a different prioritisation of vulnerabilities with respect to their probability of being exploited and is also estimated

using AHP. In contrast to its impact value, which is unique for its vulnerability, the likelihood value depends on the combination of a threat-vulnerability pairing, as the same vulnerability can be exploited by more than one threat but with a different likelihood.

The initial amount of risk introduced by a threat is an aggregation of the risk introduced by each of the vulnerabilities exploited by the threat and is captured by the *InherentRisk* attribute of the *Risk* concept. The amount of risk remaining after risk treatment is applied is captured by the *ResidualRisk* attribute. Additionally, the attribute *ResidualRiskThreshold* captures the maximum accepted amount of residual risk for each threat by the system stakeholders.

The concept of the *Security Mechanism*, which Secure Tropos uses to model technologies utilised to implement the system's security objectives, is extended with a number of attributes. These attributes will allow us to evaluate the contribution of each security mechanism towards the achievement of each of the system's soft-goals (*SoftGoalContribution*) and the mitigation of each identified vulnerability (*VulnerabilitytMitigation*).

Finally the *Coverage* attribute has been added to the *Soft Goal* concept to capture the total coverage provided to each by the selected sets of security mechanisms.

Risk Calculation

The newly introduced concept of Risk and additional attributes to the existing Secure Tropos concepts facilitate the definition of functions which can be used to guide the risk-based adaptation process. More specifically:

Definition 1 *Let $V_1, \ldots, V_n$ denote the vulnerabilities of the system, and let $L_i, I_i \in \mathbb{R}$, with $0 \leq L_i, I_i \leq 1$, denote the Likelihood of V_i being manifested and its Impact, respectively. Let $V_i \in \{0,1\}$ indicate the exploitation of vulnerability V_i by a threat $V_i = 1$, or not $V_i = 0$.*

The ***Inherent Risk****, R_I, introduced by a threat is defined by:*

$$R_I = \sum_{i=1}^{n} (L_i \times I_i \times V_i). \tag{3.1}$$

Definition 2 *Let $m_i \in \mathbb{N}$ be the number of security mechanisms mitigating vulnerability V_i, and let $M_{ji} \in \mathbb{R}$, with $0 \leq M_{ji} \leq 1$, denote the Vulnerability Mitigation of the j-th security mechanism towards a vulnerability V_i. The* ***Mitigated***

Risk *of a threat,* R_M *, is defined by:*

$$R_M = \sum_{i=1}^{n} L_i \times \overline{\quad} \times \prod_{j=1}^{m_i} \frac{M_{ji}}{m_i} \quad . \tag{3.2}$$

The residual risk of each threat is the remainder of its inherent risk when the mitigated risk is redacted.

Definition 3 *The* ***Residual Risk*** *of a threat,* R_R *is defined as:*

$$R_R = R_I - R^M \overset{(3.1),(3.2)}{=} \sum_{i=1}^{n} (L_i \times \overline{\quad} \times V_i) \left(1 - \prod_{j=1}^{m_i} \frac{M_{ji}}{m_i}\right) \quad . \tag{3.3}$$

Constraint Goal Models

Goal models often present high variability, expressed by multiple alternative solutions to fulfil one or more goals. One of the tasks of GORE is to decide which of these alternatives should be implemented or not in the system-to-be. Given the nature of goal models, each goal represents a predicate that relates with other predicates through AND/OR relationships. Therefore such relationships between goals can be used to construct first order logic formulas.

In order to elaborate on complex aspects of system designs, captured by goal models, additional attributes can be assigned to different components of the models. As previously discussed, in this work we introduce a number of attributes to quantitatively capture aspects of risk, security coverage and non-functional goals. Thus, each alternative solution in terms of security mechanism leads to a goal model with different total values for each of the variables captured by the newly-introduced attributes. Hence, goal reasoning in our approach means finding a solution to a maximum satisfiability (MAX-SAT) problem.

To solve such problems we turn our attention to the field of satisfiability and optimisation modulo theories (SMT/OMT). There, the combination of the different variables are captured by formulas associated with linear equations that must be optimised by any solution found for the satisfiability problem. The integration of SMT/OMT with goal models has been implemented be Constrained Goal Models (CGMs) [130]. Such goal models allow the definition of a) multiple variables associated with the modelled goals and b) linear equations composed by these variables that should be optimised. Therefore, along with the satisfiability problem that is native to goal models, a multi-objective optimisation problem should be solved in parallel. This is done with the use of a scalable external

reasoner, OptiMathSAT [131], which is invoked to find optimal solutions over CGMs.

The use of such a reasoner allows for flexibility to the optimisation process as system designers and stakeholders can decide both which variables capture critical aspects of the system and should, therefore, be included in the formulas, and the priority of each of the selected variables in the optimisation process. As a result, the application of the reasoner can produce a number of system configurations depending on the selected variables and their prioritisation. This allows for constructing a number of scenarios during the decision support step of the approach, each of which produces a different system configuration in terms of selected security mechanisms. Each of the resulting configurations can be used to produce a different business process instance by following the rest of the framework's steps.

3.3.2 Decision Support Process

The aim of the Decision Support component is to support the selection of the system's security implementation. The input required is a Secure Tropos goal model where a multitude of security mechanisms and threats have been identified, as a result of the system's security analysis via the application of the Goal Modelling component. The output is a combination of such mechanisms that best satisfy the system properties defined by its stakeholders. The steps followed to perform the decision support process are as follows:

Step 1 *Optimisation Variables Selection*: The variables capturing relevant system aspects, based upon which an optimisation process will be performed, are selected by the system stakeholders. Since the optimisation process introduced in this work is security-oriented, the selection focuses on the Residual Risk variable for each of the identified system threats, as defined in Formula 3.3. The coverage provided by each security mechanism towards the satisfaction of each security constraint is another relevant security-related aspect and is, therefore, used as another optimisation variable. Other than the security and risk-related variables, a number of non-functional goals may be relevant in the decision making process. Therefore, variables reflecting such system aspects (e.g., cost, performance) should be defined as system soft-goals, towards which each of the proposed security mechanisms contribute.

Step 2 *Value Assignment* : The selected variables, expressed as attributes of com-

ponents of the system's goal model have to be instantiated. The instantiation process includes assigning values for security constraint coverage, in a scale of zero (0) to one (1), for each proposed security mechanism, according to estimations provided by security experts. In a similar manner, the soft-goal coverage values are instantiated, in a scale of zero(0) to one hundred (100), to indicate the contribution of each proposed mechanism toward the achievement of the identified system soft-goals.

For the instantiation of the risk-related variables, the formulas introduced in Section 3.3.1 have to be evaluated. First,the calculation of the Inherent Risk (see Formula 3.1) for each of the system's threats is performed by instantiating the Likelihood and Impact values of each threat's vulnerabilities using AHP. Next, the Risk Mitigation (see Formula 3.2) provided by each of the proposed security mechanisms is instantiated according to the estimations of security experts.

Step 3 *Variable Prioritisation:* Once all relevant variables have been assigned with numerical values, the optimisation process has to be defined. Such a process, supported by the OptiMathSAT satisfiability solver, allows the definition of both hard and soft cap values for each variable. This means that the system stakeholders can optionally assign a specific value which a variable cannot exceed (hard cap) (e.g., $S_{Conf} > 75\%$), a min/max optimisation direction (soft cap) (e.g., $Performance -> MAX$, $ResidualRisk(R_R) -> MIN$) or a combination of both. The solver also facilitates the prioritisation of variable satisfiability, therefore each of the variables can be assigned a priority in the satisfiability problem. As a result, a variable with a higher priority will be optimised before a variable with a lower priority. OptiMathSAT also allows complex constraints to be defined as functions of the selected variables (e.g., $TotalResidualRisk = 0.5 * R_{R(T1)} + 0.3 * R_{R(T2)} + 0.2 * R_{R(T3)}$) and prioritised in the same way as the rest of the variables.

Step 4 *Security Implementation Generation:* Once all variables have been selected, instantiated and (optionally) prioritised, the satisfiability solver can now generate a combination of security mechanisms that optimally satisfies the defined optimisation problem. It can be the case that the problem cannot be solved, therefore, it may be required that Step 3 is repeated and different priorities and/or caps are defined. Nevertheless, if the optimisation problem can be solved a combination of the selected mechanisms is provided by the solver along with the overall values of the variables produced by the solution

(e.g., Total Cost, Total Risk Mitigation).

Step 5 *Security Implementation Selection:* The decision support process usually involves the definition of multiple optimisation scenarios during Step 3, in order to represent different optimisation priorities of the system's stakeholders (e.g., lower cost, highest risk mitigation). During this final step and once combinations of security mechanisms that satisfy each of the defined scenarios has been generated, the system's stakeholders select the mechanism combination that will be implemented in the system to-be.

3.3.3 Decision Support Component Application

The steps for the application of the Decision Support component are overviews in Fig. 3.10 and applied to the example e-Prescription system to support the stakeholders in the definition of its security composition.

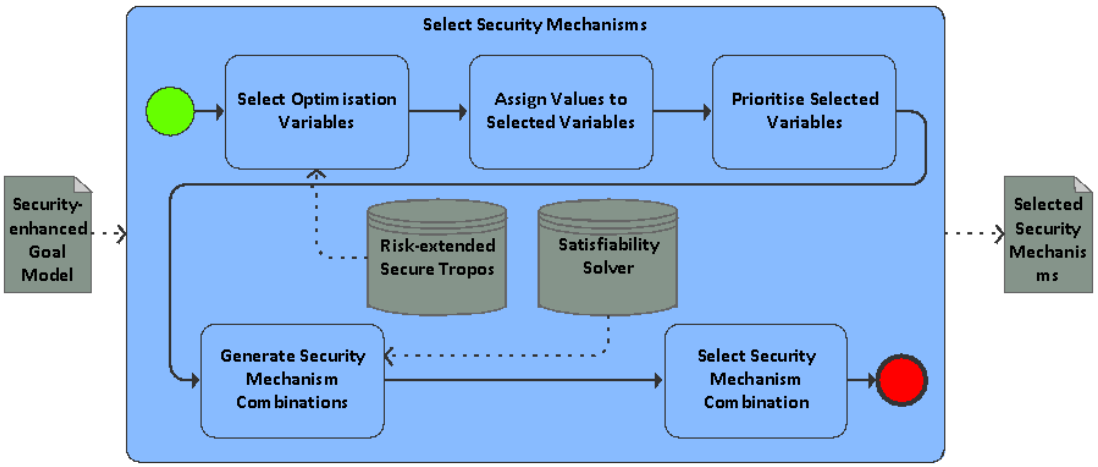

Figure 3.10: Activities for the application of the Decision Support component

The first step of the decision support process, the Security Analysis, has already been performed by the Goal Modelling component and resulted in the Security Requirements and Security Attacks models presented in Figs. 3.6, 3.7 and 3.8. The next step involves the selection of the variables along which the optimisation process will take place. Since two threats have been identified during the security analysis of the e-Prescription system, the residual risk of each of such threats forms the first set of optimisation variables (i.e., $R_{R(T1)}$ and $R_{R(T2)}$). Another set of variables captures the satisfaction of each identified security constraint by each of the proposed security mechanisms (i.e., S_{Auth}, S_{Int}, S_{Conf}). Furthermore, the soft-goals identified at the Security Requirements model of the system

identify non-functional system aspects which the stakeholders consider an important part of the system's design. Thus, the variables of *Cost* and *Efficiency* are also introduced as aspects of the optimisation process.

Since all variables, around which the decision making process is built, have been identified, the next step requires their value assignment. For the calculation of the residual risk values, as indicated by Formula 3.3, we first need to calculate the individual Impact and Likelihood values for each vulnerability of each threat using AHP, in order to capture a quantitative ranking of each vulnerability. The pairwise ranking approach of AHP allows security experts to assign Impact values by comparing all three of the identified vulnerabilities. Similarly, the Likelihood values are calculated by ranking each threat-vulnerability pairing (i.e., T1-V1, T1-V2, T2-V3 as modelled in Figs.3.7 and 3.8), as the same vulnerability can be exploited by more than one threat but with a different likelihood. The impact and likelihood values for each threat, instantiated as a proof-of-concept for the specific example, are used to calculate the inherent risk for each threat, as presented in Tab. 3.2.

Threat	Vulnerability	Impact	Likelihood	Inherent Risk
T1	V1	0.25	0.75	0.3125
	V2	0.5	0.25	
T2	V3	0.25	1	0.25

Table 3.2: Threat - Vulnerability value assignment for the e-Prescription system

The security mechanisms proposed in Fig. 3.6 also require the value assignment of their attributes which capture the mitigation percentage of each vulnerability (M_V) and their contribution towards the satisfaction of each security constraint $S_{Constr.}$ and soft-goal. Security experts and system analysts need to assign such values to each of the proposed security mechanisms. For the example e-prescription system such values are assigned as shown in Tab. 3.3.

Security Mechanism	M_{V1}	M_{V2}	M_{V3}	S_{Int}	S_{Auth}	S_{Conf}	*Cost*	*Effic.*
MD5	0	0	0.25	0.4	0	0	15	80
SHA2	0	0	0.70	0.75	0	0	20	80
SmartCard	0.4	0.5	0	0	0.75	0	75	70
2FA	0.6	0.7	0	0	0.9	0	70	30
User/Pass	0.2	0.3	0	0	0.6	0	30	50
HTTPS	0	0	0	0	0	0.6	10	80
Private VPN	0	0	0	0	0	0.8	40	50

Table 3.3: Security mechanism value assignment for the e-Prescription system

The next step in the decision support process requires the stakeholders to prioritise the variables involved in the prioritisation. For the example e-Prescription system three different scenarios have been defined, each of which involved different priorities and caps for the identified variables. Each scenario was provided as an input to the OptiMathSAT satisfiability solver which produced a different security mechanism combination to satisfy each scenario's parameters. The scenarios created for this example are the following:

- *Scenario 1*: This scenario represents a system composition where the top priority of the stakeholders is the minimisation of the residual risks of the two identified threats. The next priority is the maximisation of the security constraint satisfaction followed by the minimisation of costs and the maximisation of the system's efficiency. No hard cap limits were set for any of the variables.

- *Scenario 2*: This scenario represents a system composition where the top priority is the minimisation of costs, followed by the maximisation of efficiency, the maximisation of constraint satisfaction and finally the minimisation of residual risks. Once again, no hard cap limits were set for any variable.

- *Scenario 3*: In this scenario, hard caps have been set for both the residual risks of the two identified threats and the for the satisfaction of each security constraint. More specifically, each residual risk must be less than 50% of the initial (inherent) risk and each security constraint must be at least 50% satisfied. The cost has been set to be minimised and the efficiency to be maximised.

Variable	Scenario 1	Scenario 2	Scenario 3
$R_{R(T1)}$	min[1]	min[6]	< 50%
$R_{R(T2)}$	min[2]	min[7]	< 50%
S_{Int}	max[3]	max[3]	> 50%
S_{Auth}	max[4]	max[4]	> 50%
S_{Conf}	max[5]	max[5]	> 50%
$Cost$	min[6]	min[1]	min
$Effic.$	max[7]	max[2]	max

Table 3.4: Variable values and thresholds per adaptation scenario

An overview of the priorities and caps of each variable for each of the three scenarios is provided in Tab. 3.4. The security mechanism combinations that

satisfy the initial conditions for each scenario, as identified by the optimisation solver, are presented in Tab. 3.5. System stakeholders should, at this point, be able to select the security mechanism combination resulting from the scenario best representing their needs. The selected mechanisms will be later used to instantiate the business process model during the application of the process modelling component of the framework. For the purposes of this example we will select the security mechanisms combinations resulting from Scenario 3.

Scenario 1	SHA-2	2-Factor Auth.	Private VPN
Scenario 2	MD5	User/Pass	HTTPS connection
Scenario 3	SHA-2	2-Factor Auth.	HTTPS connection

Table 3.5: Resulting system configurations per scenario

Therefore, the role of the Decision Support component is to guide the selection of the security countermeasures that will be implemented in the system to-be. To achieve that it quantifies the contribution of each of the proposed security mechanisms towards the satisfaction of a number of different system properties such as risk mitigation, security constraints satisfaction and non-functional aspects (e.g., performance, cost). The prioritisation of the satisfaction of such system properties creates a number of optimisation scenarios, each of which can be satisfied by a different combination of security mechanisms. Therefore, the system's stakeholders can make an informed decision regarding the system's security composition, by selecting the optimisation scenario best representing their needs.

3.4 Model Transformation component

The components of the proposed framework introduced thus far facilitate the elaboration and analysis of functional and non-functional aspects of the system to-be at a high level of abstraction (i.e., organisational level). Due to this high abstraction level it is easier for non-technical stakeholders (e.g., management, business analysts) to be involved in defining the objectives, high-level requirements and constraints of the system to-be and capture and refine them using goal-oriented requirements engineering approaches. In order to transfer such elements of the organisational structure to the operational level at which business processes operate, a linkage between the two levels of abstraction needs to be created. This linkage is a crucial step for the creation of operational level artefacts (i.e., business process models) as it provides a blueprint for business process designers who are able to built business processes which are aligned with organisational level artefacts of the system (e.g., goals, requirements, constraints).

To achieve that, during the model transformation phase, we introduce an intermediate model called *hybrid reference process model*. This model includes concepts from both goal and process models (*hybrid*) and captures all the security-related information elicited from the Goal Modelling and Decision Support components of the framework. The model produced as a result of the application of the Model Transformation component can be later instantiated into a number of similar but slightly different business process models (*reference model*), according to the specific security needs of each instance.

The process related concepts (i.e., lanes, activities, data objects) included in the hybrid reference process model are transformed from their corresponding goal model concepts (i.e., actors, goals, plans, resources) and also inherit the Secure Tropos concepts capturing security-related analysis (i.e., constraints, objectives, mechanisms, threats). By capturing such connections between goal and process model level concepts via the hybrid reference process model we can trace changes at the high-level requirements of an organisation to specific parts of its business processes and vice-versa.

3.4.1 Concept Mappings and Model Transformation Steps

To identify conceptual similarities between goal and process modelling concepts and create explicit transformation rules we use the meta-models and concepts definitions provided by Secure Tropos [6] and BPMN 2.0 [7]. More specifically, a *lane* in BPMN 2.0 is described as a container for organising and categorising

activities [7], usually performed by a specific entity (e.g., process participant, information system). Since an *Actor* is also used as a container for goals and plans to be achieved by an entity in the context of goal models, we can transform the actors included in the goal model to lanes of the same name in the hybrid reference process model, as described in Fig.3.11. Therefore, information regarding the participants and stakeholders of the system, originally captured in the goal model can be transferred to the business process via this concept mapping.

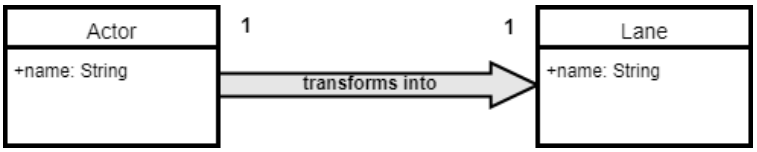

Figure 3.11: Actor to lane concept relationship

In a similar manner we can map the goals of each actors and the plans used to achieve them, as included in the metamodel to process activities. An *Activity*, according to the definition of BPMN 2.0, is a generic container for work performed by an entity [7] and can take two distinct forms, a *Sub-Process* and a *Task*. The difference between sub-processes and tasks is that the former can be broken down into a finer level of detail while the latter captures atomic activities that cannot be further decomposed. Similarly in goal models, goals are used as containers for capturing the intentions of system actors and can be further decomposed to a finer level of detail, while plans express atomic actions that need to be performed for the achievement of a goal. Thus, as illustrated in Fig. 3.12, by transforming goals to sub-processes and plans to tasks in the hybrid reference process model, we can transfer information regarding the intentions of each actor and use them to generate the main activities to be included at the business process level.

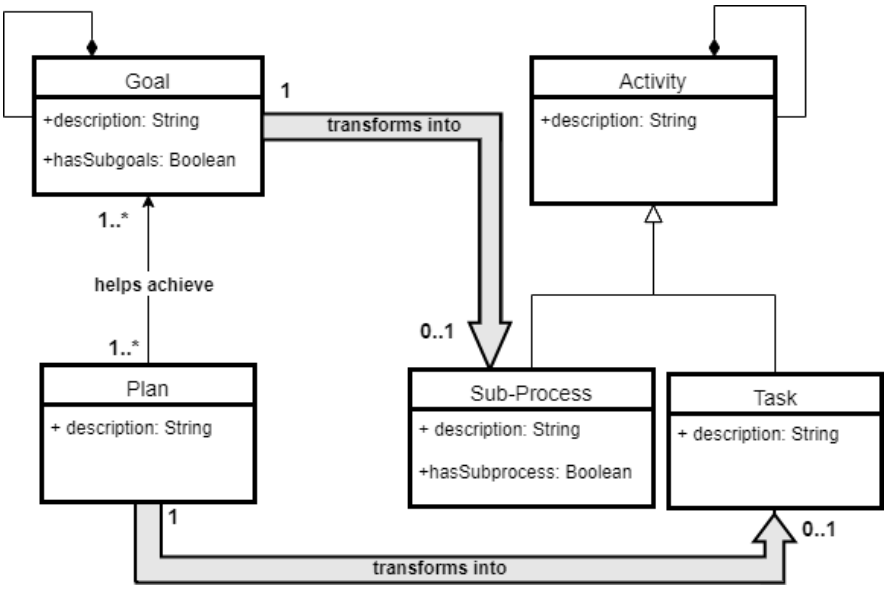

Figure 3.12: Goal and plan to activity concept relationships

The exchange of information assets in physical or digital form is one of the fundamental components of a business process. For this purpose the concept of *Data Objects* is included in BPMN 2.0 and defined as entities providing information about what activities require to be performed and/or what they produce [7]. Similarly, at the goal model level resources are used to capture information entities which are required for or created from the fulfilment of a goal or the performance of a plan. Therefore, due to the conceptual similarities between the two concepts, the resources included in the goal model can be transformed to data objects at the hybrid reference process model, as shown in Fig. 3.13. This way information captured at the goal model regarding such assets can be transferred to the business process model.

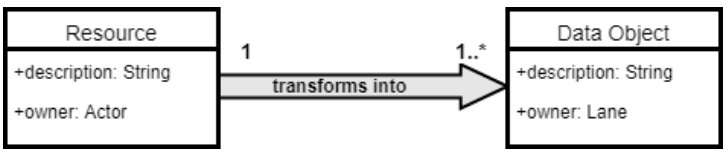

Figure 3.13: Resource to data objects concept relationships

As mentioned earlier, apart from the business process model concepts, the hybrid reference process model inherits a number of concepts from the Secure Tropos goal model. More specifically, concepts used to capture security aspects (i.e., security constraints, security mechanisms, threats), connected with goals, plans and resources of the goal model are transferred to the hybrid reference

process model and connected to the corresponding activities and data objects. An overview of the concepts and relationships included in the hybrid reference process model are provided at the metamodel, illustrated in Fig.3.14, where the concepts inherited by Secure Tropos are included in the dashed-line container.

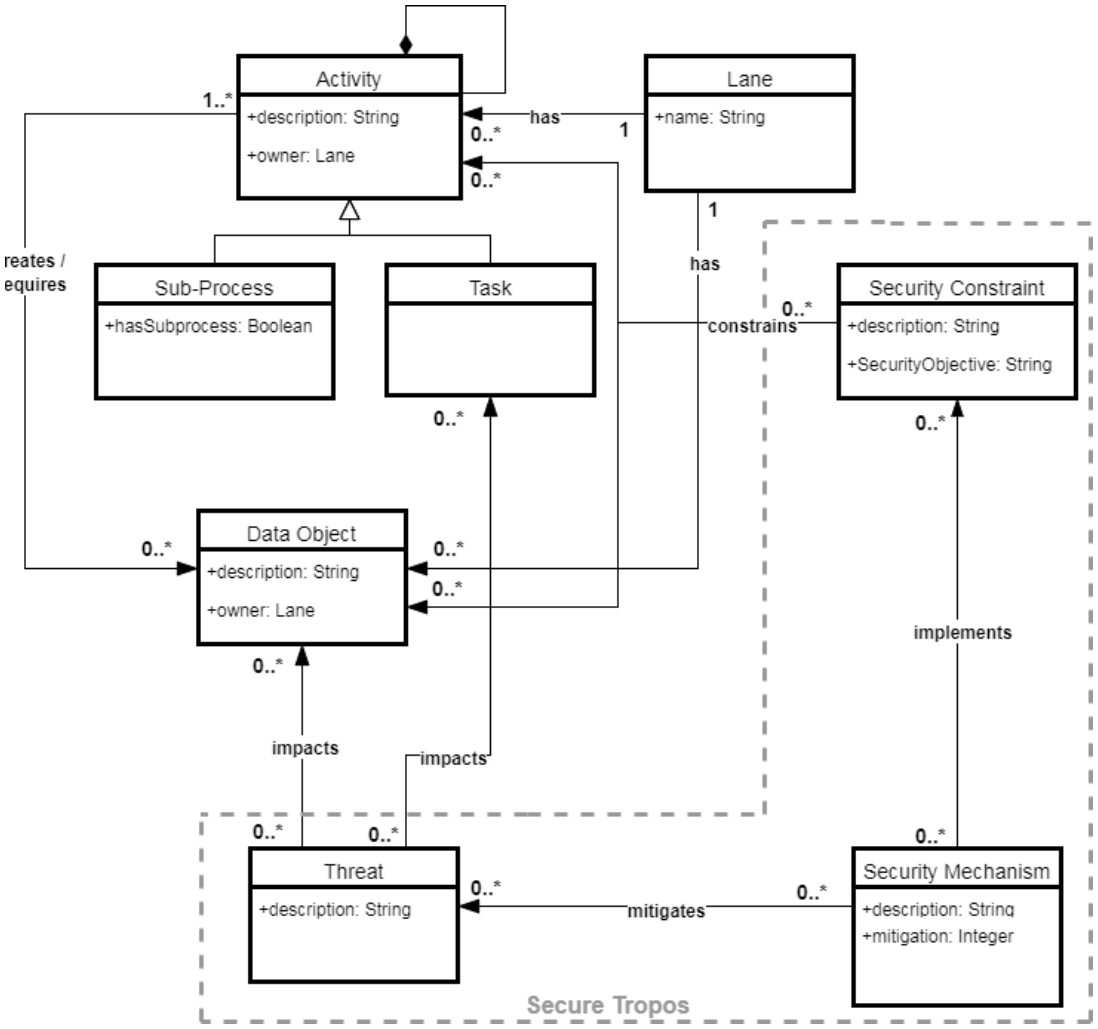

Figure 3.14: Metamodel of the hybrid reference process model

A series of transformation steps have been defined in Tab. 3.6 for guiding the process of creating a hybrid reference process model starting from a security oriented goal model. The mappings between concepts of Secure Tropos and BPMN 2.0 introduced above, are the basis upon which each of the transformation steps is built. Each of the transformation steps are to be applied iteratively for each of the components included in the security requirements view of the Secure Tropos goal model created by the application of the previous components of this framework.

Step 1	$\forall(ac)$ (actor) of the goal model: $\exists(l(ac))$ (lane) in the hybrid model.
Step 2	$\forall(g)$ leaf-level (goal) of the goal model: $\exists(sp(g))$ (sub-process) in the hybrid model. $\forall(p)$ leaf-level (plan) of each goal (g) the goal model: $\exists(t(p))$ (task) within ($sp(g)$) in the hybrid model.
Step 3	$\forall(r)$ (resource) of the goal model: $\exists(d(r))$ (data object) in the hybrid model.
Step 4	$\forall(c)$ (security constraint), $\forall(m)$ (security mechanism) and $\forall(t)$ (threat) connected to a goal (g), plan (p) or resource (r) of the goal model: *Transfer* it to the hybrid model. *Connect* it to the corresponding activities ($sp(g)$ || $t(p)$) or data objects ($d(r)$).

Table 3.6: Steps for the goal-to-hybrid reference process model transformation

3.4.2 Model Transformation Component Application

The application of the transformation steps of Tab. 3.6 to the e-Prescription system's goal model produces the hybrid reference process model illustrated at Fig.3.15. More specifically, the actors introduced during the organisational level analysis of the system (i.e., *Patient, Medical Practitioner* and *E-Prescription System*) are transformed into business process lanes according to *Step 1* of the transformation rules. Next, according to *Step 2*, activities, in the form of sub-processes and tasks, are created and placed in the corresponding lanes, originating from the leaf-level goals and plans of each system actor. Goals participating in dependency relationships are to be placed as sub-processess only within the lane representing the dependee actor, in order to avoid duplicate activities appearing in multiple lanes. During *Step 3*, the relevant resources (e.g., *PatientInformation* and *Prescription*), previously introduced at the goal model, are now data objects at the hybrid reference process model connected as inputs or outputs to the activities that create or require them. For instance, since the *"Prescription"* resource is created by the plan *"Issue Prescription"* at the goal model, a data resource with the same name is the output of the corresponding task at the hybrid reference process model. In contrast to goals, resources participating in dependency relationships in the goal model, create data objects in both the lanes representing the depedee and depender actors.

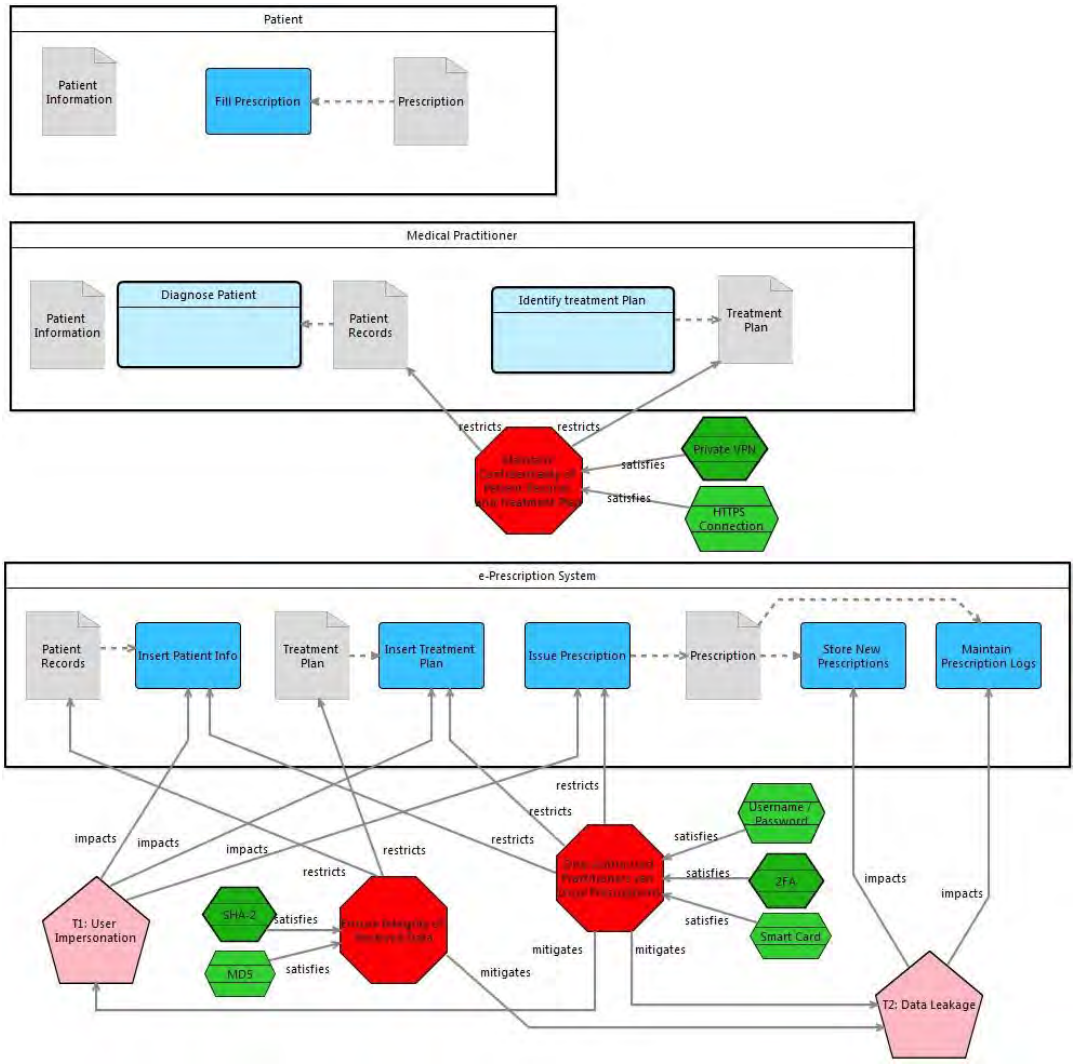

Figure 3.15: Hybrid reference process model of the e-Prescription system

After all the concept transformations have been completed, a basic process skeleton capturing the main participants and activities of the system has been created. To also capture the security related aspects of the system on this process skeleton we apply *Step 4* of the transformation rules. According to that step, the constraints connected to a goal, plan or resource of the goal model are transferred in the hybrid reference process model and connected to the corresponding sub-process or task. In case of a constraint placed at a non leaf-level goal at the goal model, connections are created to all activities stemming from that non leaf-level goal at the hybrid reference process model. For instance the constraint *"Only authorised practitioners can issue prescriptions"* originally connected to the goal *"Create Prescription Document"* at the goal model presented in Fig. 3.6, will be

connected to all three of the activities created from the leaf-level nodes of that goal (i.e., *"Insert Patient Info"*, *"Insert Treatment Plan"* and *"Issue Prescription"*) at the hybrid reference process model. The same process is followed for transferring the threats identified at the goal model to the corresponding activities and data objects in the hybrid reference process model level. The security mechanisms identified for the satisfaction of each of the constraints are also transferred and connected to the corresponding constraint. To maintain the maximum amount of information at the hybrid reference process model level, all proposed mechanisms identified at the goal model level are transferred. The mechanisms selected as a result of the Decision Support component application are distinguished by their bold border, while the rest mechanisms are included in case of future system redesigns, which may lead to the selection of alternate security configurations. The resulting hybrid reference process model for the e-Prescription system is illustrated in Fig. 3.15

3.5 Business Process Modelling component

The business process modelling component uses the hybrid reference process model as input in order to produce secure business process designs. For each security-constraint activity or resource of the hybrid reference process model, a security mechanism has been selected to be implemented using the Decision Support component, as presented in Section 3.3. The Business Process Modelling component handles the operationalisation of the selected implementation mechanisms and their integration within the final business process model. To provide a structured approach towards security operationalisation for process designers, the Business Process Modelling component introduces a set of security design patterns in the form of process fragments. Such patterns are instantiated and integrated to the process skeleton, captured by the hybrid reference process model, which is then manually refined to create a complete BPMN business process model.

3.5.1 Business Process Design Patterns

For the operationalisation of security implementing mechanisms in the business process model we introduce a series of business process design patterns. A pattern, in the context of software development, is a reusable package which incorporates expert knowledge and represents a recurring structure, activity, behaviour or design [132]. Specifically for the area of information security, a common obstacle in the design of secure information systems is the disconnect between security experts and the system developers [133]. Since the main concern of system developers is functionality, security is underprioritised and implemented in an ad-hoc manner during the later development stages. Security patterns are often utilised as a way to overcome such issues, as they are able to provide to non-experts standardised and proven solutions to common security-related issues [134]. Patterns can encapsulate security expertise and standardise proven solutions to recurring problems [133], which can facilitate a systematic and structured approach towards the operationalisation of security by non-experts [135]. A security pattern is a well-understood solution to a recurring information security problem and can be categorised in structural patterns, which incorporate designs that can be implemented in the final product and procedural patterns, which represent high level directions for improving the process of development of security-critical software systems [133].

During the requirements and analysis phases of the system development life-

cycle, the majority of the proposed design pattern focus on security attacks while patterns for implementing countermeasures are few [132]. Therefore, as part of this work a number of structural process design patterns are introduced, aiming to model the implementation of countermeasures for the main types of security requirements (e.g., confidentiality, integrity, availability) at a business process level of abstraction. Such patterns are at a mid-level of abstraction and are, therefore, generic enough to be implementation-agnostic but able to specify a basic sequence of activities and interactions between process participants which lead to the satisfaction of the system's security requirements.

The basic structure of each of the proposed patterns is captured using BPMN collaboration diagrams [7] and includes the activities required for the operationalisation of a security implementing technology. Definitions from international standards [117], [136] for each type of security requirement (i.e., authentication, authorisation, confidentiality, integrity, availability) were utilised to identify the basic functionality that each pattern should describe. Furthermore, literature sources (i.e., [64], [137]) were utilised to identify how such functionality can be expressed in the context of a business process model.

The security-implementing activities included in each pattern are annotated with a padlock symbol at their top left corner to visually communicate their security-oriented nature. Corresponding activities exist at the user's lane describing any required interaction with the system's security implementing activities (e.g., input of user credentials). The security-constrained activity or data object, which created the need for the implementation of security, is marked with a bold black border in order to be easily distinguishable from other activities or objects. A series of message exchanges between the two lanes are also included to capture the communication between the user and system side during the interaction with the various mechanisms and for communicating the success or failure of the operation (e.g., *"Access Granted"*). Finally relevant start and end events along with gateways that split the process flow are also modelled within each pattern. An overview of the BPMN 2.0 concepts utilised for the construction of the patterns is presented in Fig. 3.16

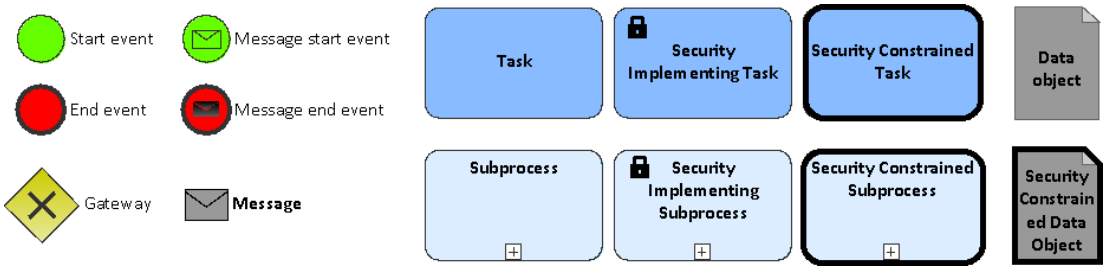

Figure 3.16: Overview of BPMN 2.0 elements used in patterns

The activities contained within each pattern are not dependent on the implementation of a specific mechanism but rather on the type of the security requirement at hand. Therefore, the pattern operationalising a specific type of security requirement (e.g., authentication) can be instantiated by a number of different mechanisms (e.g., smartcard, biometrics, username/password). It is also the case that one pattern can be reused within another pattern. For instance, the pattern for Authentication is reused within the Authorisation pattern since its functionality is required for the completion of the authorisation process.

The instantiation and contextualisation of each pattern for its introduction to a specific business process model is a semi ad-hoc process performed by the process designer, guided by a set of steps. More specifically:

1. An activity or data object with an attached security constraint is selected from the hybrid reference business process model.

2. The type of security constraint (e.g., confidentiality, integrity) restricting the selected activity or data object is identified from the hybrid reference process model and the corresponding security process pattern is selected to be further instantiated.

3. The security mechanism(s) attached to the selected security constraint at the hybrid reference process model is used to instantiate the security-implementing activities included in the security pattern. For instance, a security-implementing activity such as *"Request Authentication Details"* which is present in the non-instantiated Authentication pattern is altered by the process designers into a more explicit declaration (e.g., *"Request 2-Factor Authentication Details"*) to reflect the implementation of a specific security mechanism, which has been selected by the stakeholders via the application of the Decision Support component.

4. The activity or data object selected from the hybrid reference process model during Step 1, is used to instantiate the security-constraint activity or data object field of the selected security process pattern, visually represented with a bold black outline.

5. The instantiated security pattern is manually connected to the rest of the business process by the process designer. More specifically, the control flow, gateways and events contained within the pattern have to be connected with the control flow of rest of the business process model according to the syntax rules of BPMN 2.0. The position of the pattern with the business process model is relative to the position of the security-constraint activity or data object. For instance, the pattern for Authentication is placed before the execution of an authentication-constraint activity, while the pattern for integrity is placed after the creation or transmission of an integrity-constraint data object.

While the above steps provide the process designers with a set of predefined steps for the instantiation and integration of the security patterns within a business process model, there are still design choices that have to be made depending on the context of the business process at hand. More specifically, the appropriate connection of an instantiated pattern within the control flow of a business process model can require some fine-tuning under certain conditions. For instance, if a constraint activity is located within a looping control flow, or a number of constraint activities are present in succession, then a pattern has to be correctly placed so unnecessary repetition is avoided. Such cases of complex control flows prevent the complete automation of the security pattern instantiation and thus require the intervention of a process designer who can adjust the process according to the context of the model at hand. Nonetheless, the security process patterns presented in the rest of this section, along with the steps discussed above, provide a structured way for process designers to integrate security during the design of business processes.

Regarding the different types of security requirements, patterns are created for operationalising confidentiality, integrity and availability countermeasures. Requirements such are authentication and authorisation are often also grouped under security, therefore the authentication and authorisation patterns are integral parts of the rest of the security design patterns presented below.

Authentication

Authentication, in the context of a business process, entails the verification of a credential of a subject using security mechanisms [64]. Therefore, a process participant is required to have a verified identity before performing a specific activity or accessing a resource. To realize the authentication requirement, as illustrated in Fig. 3.17, every time a user submits a request to the system for accessing an authentication-constrained resource or activity, the system should check that request and ask for the user's authentication data. Once the user submits the authentication data in the appropriate form (e.g., username/password, biometric data) the system should check its validity and, if valid, allow the user to access to the constraint resource or activity.

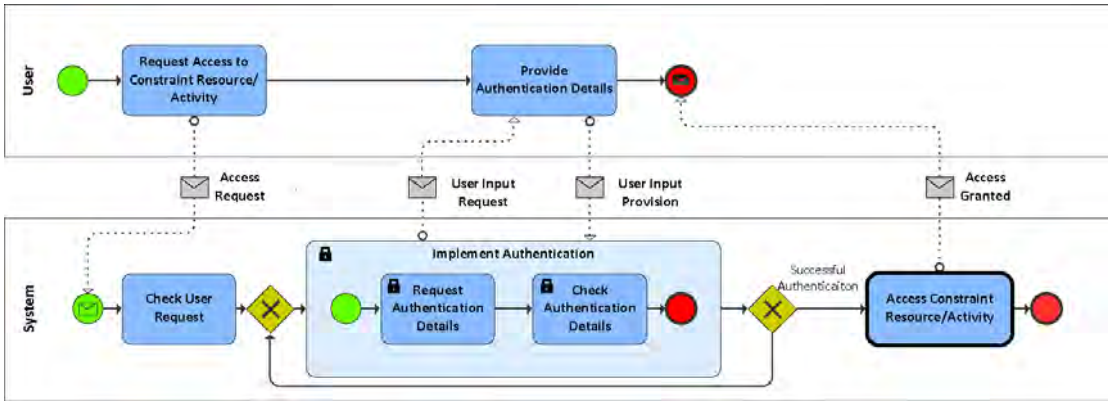

Figure 3.17: Authentication pattern

Authorisation

Authorisation, in terms of a business process model, requires the restriction of access to assets based on certain business or security requirements of an entity [117]. Therefore, only process participants with the appropriate permissions can access a resource or perform an activity that is authorisation-constrained. As shown in Fig. 3.18, to realise the authorisation requirement, first a user requests access to authorisation-constrained activities or resources and the authentication process takes place in order for the user's identity to become known to the system. After the successful authentication, the role and/or the permissions attached to the user's account are checked and, if appropriate, the user gains access to the constraint activity or data object.

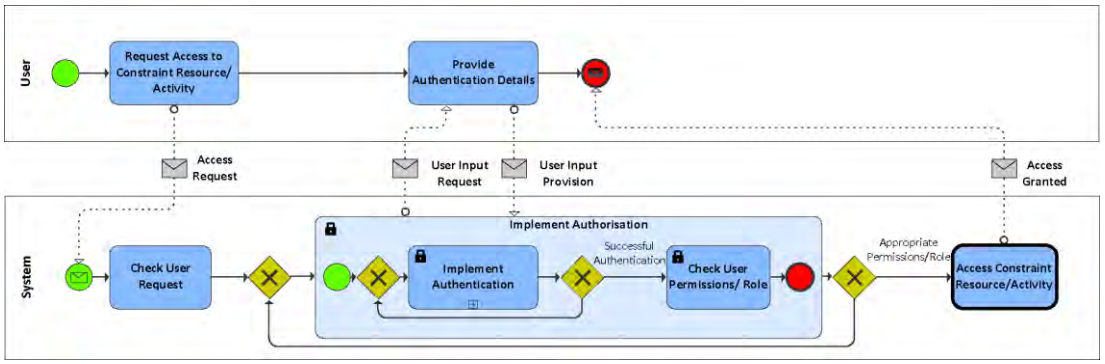

Figure 3.18: Authorisation pattern

Confidentiality

Confidentiality, in terms of business process models, is a property of a data object and involves the identification of authorised entities that can access it[137]. As shown in Fig. 3.19, to achieve confidentiality in a business process, if the user is not already authorised, the authorisation process takes place as previously described. Next, a secure communication channel is created between the user and the system through which the confidentiality-constrained data object can be transferred.

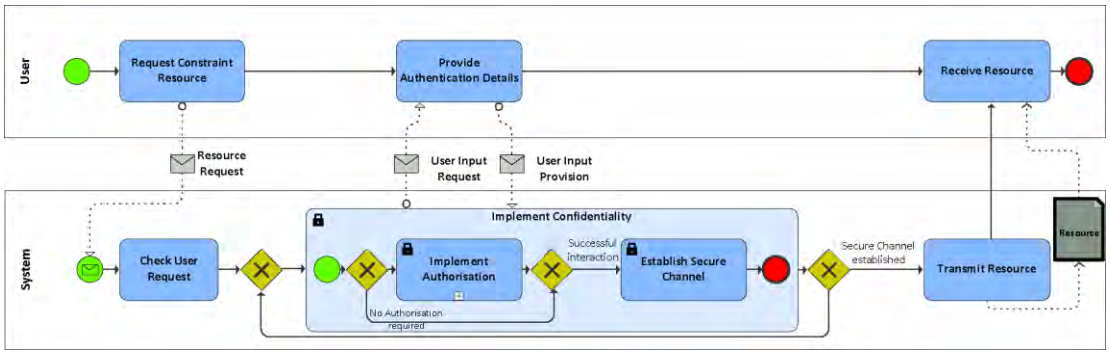

Figure 3.19: Confidentiality pattern

Integrity

Integrity is concerned with ensuring that information is protected from improper modifications so as to avoid intentional or accidental unauthorised changes to system data [136]. As illustrated in Fig. 3.20, to achieve integrity, after an integrity-constrained data object has been transferred to the system, the system's copy of the resource needs to be compared to the original by data validation techniques.

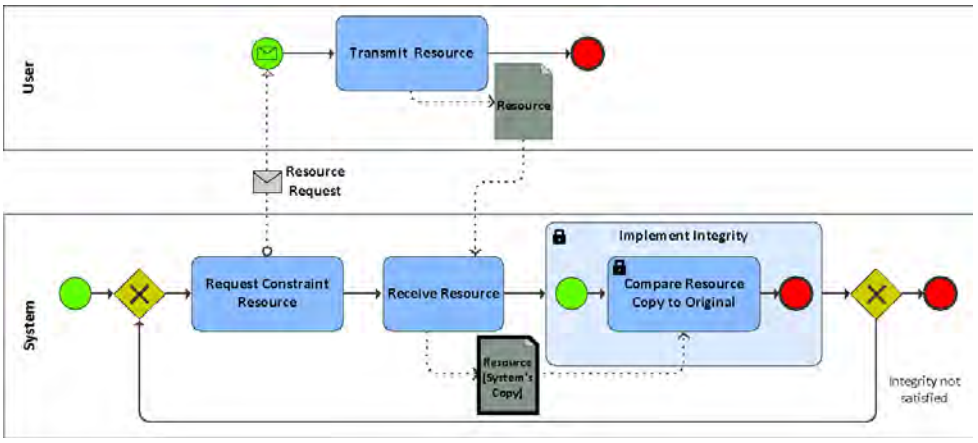

Figure 3.20: Integrity pattern

Availability

Availability describes the property of system resources being accessible and usable upon demand by an entity [117].Therefore, the pattern for availability, presented in Fig. 3.21, is utilised to ensure that critical resources are always available to process participants. To realise that requirement, when a requested resource is not available, the system has to maintain backups, using a number of available implementation technologies, from which the data object can be retrieved and be made available to the user.

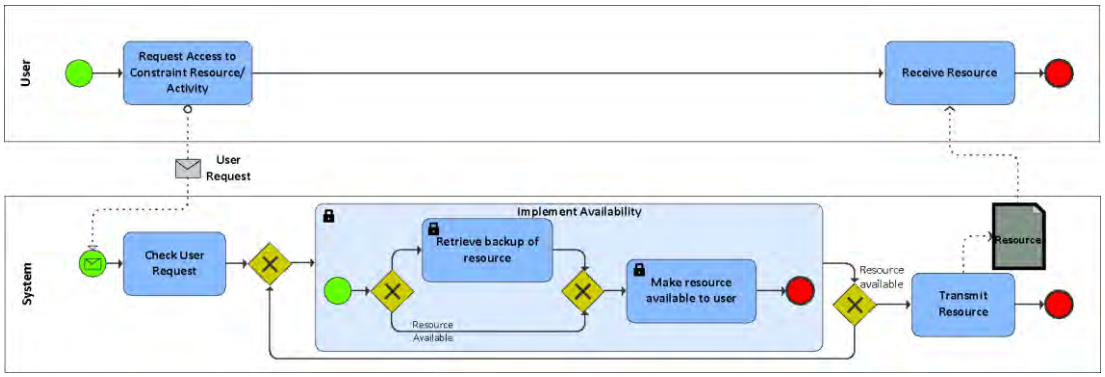

Figure 3.21: Availability pattern

3.5.2 Business Process Modelling Component Application

Other than containing the business process design pattern library, the Business Process Modelling component is also where the final business process model is created. The steps followed for the application of the Business Process Modelling

component are presented in Fig. 3.22 and applied to the e-Prescription system running example. More specifically, the process skeleton captured by the hybrid reference process model is refined with the introduction and instantiation of the security process patterns, followed by the creation of the process control flow.

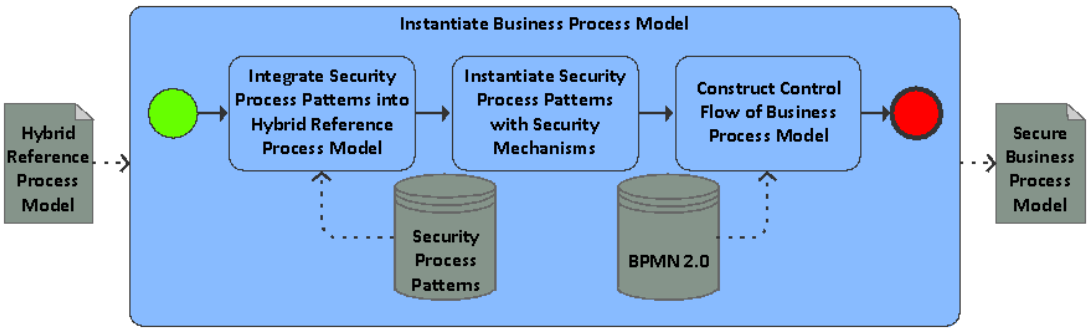

Figure 3.22: Activities for the application of the Business Process Modelling component

Figure 3.23 presents the final business process model originating from the hybrid reference model of the e-prescription system (see Fig.3.15). In the "Medical Practitioner" lane the process fragment for the implementation of *"Confidentiality"* (see Fig. 3.19) has been introduced and instantiated with the *"HTTPS Connection"* mechanism, as selected by the Decision Support component. As a result the activities *"Establish Secure Communications Channel via HTTPS"* and *"Transmit Resources"* have been introduced in the process model before the confidentiality-constraint resources *"Patient Records"* and *"Treatment Plan"* are transmitted to the "E-Prescription System" lane.

In a similar manner, in the *"e-Prescription System"* lane two process patterns have been introduced for the operationalisation of the "Authorisation" and "Integrity" security constraints. More specifically, the process fragment for *"Authorisation"* (see Fig.3.18) is introduced and instantiated with the "2-Factor Authentication" mechanism and placed before the authorisation-constraint activities *"Insert Patient Info"*, *"Insert Treatment Plan"* and *"Issue Prescription"*. Therefore, activities and messages of the authorisation pattern which were abstractly defined, such as *"Request User Input"* are instantiated into more explicit declarations (i.e., *"Request 2-Factor Authentication Details"*) in the final business process model to reflect the implementation of the selected security mechanism. Following a similar set of steps, the process fragment for *"Integrity"* (see Fig.3.20) is also introduced and instantiated with the "SHA-2" security mechanism. It is placed

after the *"Receive Resources"* activity so it can check the integrity-constraint resources received from the "Medical Practitioner" lane.

Other than the introduction of the instantiated business process design pattern for the operationalisation of the identified security constraints, start and end events have been manually added at each lane of the final business process diagram to denote the beginning and end of each of the contained sub-processes. Additionally, message exchanges have been added between lanes for transferring relevant data objects and the activities contained within each of the model's lanes have been ordered and connected with each other to create a control flow. The ordering and connecting of activities is also a manual task since the goal model, which provided us information regarding the basic structure of the system, is inherently not equipped to capture information regarding temporal dimensions of the system, such as the ordering of its plans.

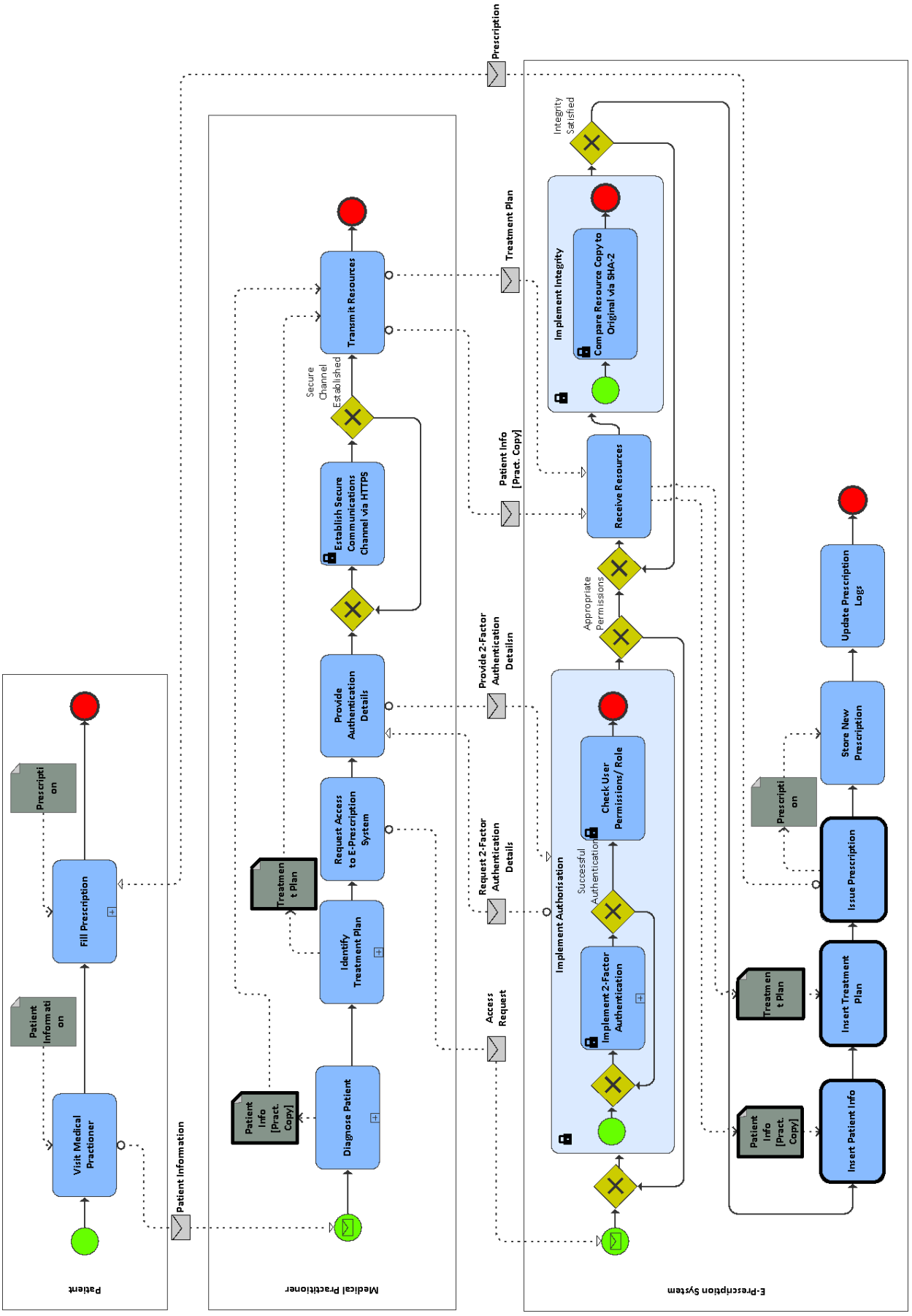

Figure 3.23: Business Process Model of the e-Prescription System

3.6 Security Verification component

The variability introduced by the numerous available process modelling languages, combined with the subjectivity and arbitrariness of manually created business process models, creates the need for formal approaches to verify the produced process designs [138]. Additionally, the verification of the compliance of an organisation's internal business processes to certain restrictions, internally (i.e., organisational standards and policies) and externally (i.e., laws and regulations) imposed, is often a legal requirement [139]. Since information security is a common source of such restrictions, the verification of the security aspects of business process models is an emerging area of research. A common approach for checking the security properties of business process models involves the specification of the process model as a formal graph, the definition of the security properties using formal propositional languages and the use of an automated model checker, which takes as input the graph and the formal property definitions to perform the model checking.

The formalisation approach appears to be widespread in the area of security verification of business process models (i.e., [64], [83], [140]–[145]), but its adoption and applicability remains limited due to its overwhelming complexity for non-expert users [139], [146]. One important drawback of such approaches is their limited support for modelling techniques, as most of them require process models to be transformed in a specific manner (e.g., Petri-nets, FSMs) before they can be used as input for a specific model checker. This contrasts with the variety of modelling languages used in practice and introduces a considerable overhead in terms of time and expert knowledge [147], as large numbers of processes need to be remodelled using a specific modelling technique. In contrast, the approach presented in this work uses BPMN 2.0, the "de-facto" standard for business process modelling [3], without the need to further translate neither the process model, nor the security requirements in formal specifications. Additionally, the range of compliance rules supported by works in the area of security verification is limited [139], as most approaches specialise to a subset of security properties, such as role assignment and user permissions (e.g., separation of duty, access control). Our work shifts the focus towards traditional security requirements (authentication, authorisation, confidentiality, integrity, availability), which can be verified by the structure of the workflow of the process.

The security verification component, introduced in this work, takes as input the business process model, as created by the previous components of the framework, in order to verify its security properties. In order to facilitate the security

verification of business process models, this component introduces an attribute-based security verification approach, which aims to provide increased usability and broad coverage for the traditional types of security requirements (authentication, authorisation, confidentiality, integrity, availability). To achieve that, existing BPMN 2.0 concepts [7] are extended with a series of attributes in order to capture information relevant to the analysis of the security properties of the process model. Using such attributes, conditions that need to apply in a process model, for the satisfaction of each type of security requirement are defined. Finally, for each type of security requirement, an algorithm is introduced, for verifying the compliance to such conditions.

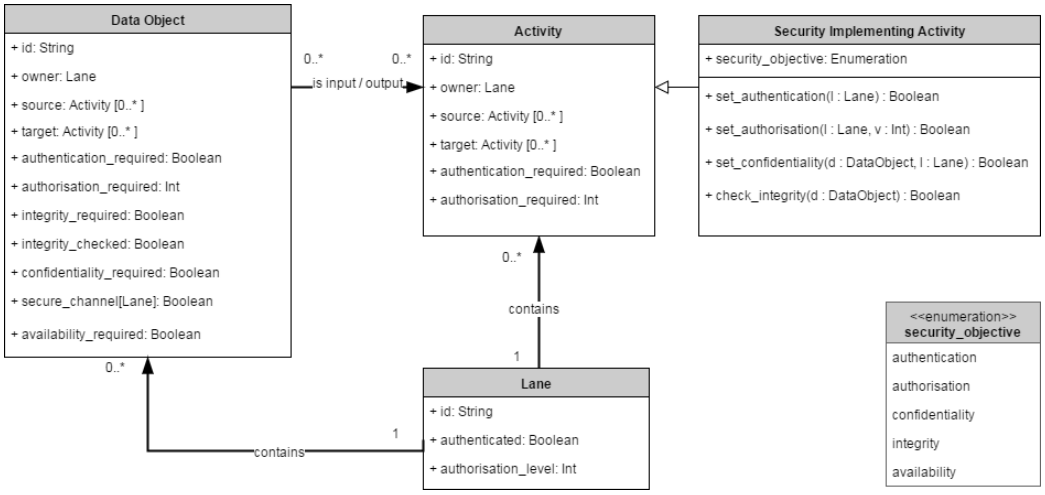

Figure 3.24: Partial BPMN metamodel with security-related attributes

3.6.1 Security Related Attributes

The modelling of security related aspects is not natively supported by contemporary graphical process modelling languages such as BPMN [3]. Nevertheless, the ability to reason and verify the security properties of a business process model requires concepts able to capture security related aspects of its elements. To that end, we propose new attributes to be added to concepts of BPMN collaboration diagrams, which will then be used for security verification purposes. A partial metamodel containing the BPMN concepts relevant to our work, along with their newly introduced attributes is presented in Fig. 3.24.

The newly introduced attributes, an overview of which is provided in Tab. 3.7, capture information regarding properties of the business process elements which are essential for the verification of their security. The type of information they

Attribute	of BPMN concept	Type	Description
id	Lane, Activity, Data Object	String	A unique identification text that describes each element of the process model.
authenticated	Lane	Boolean	A flag indicating whether a lane has been successfully authenticated.
authorisation_level	Lane	Integer	The authorisation level of the lane.
owner	Activity, Data Object	Lane	The lane in which the activity or data object is contained.
source	Activity, Data Object	Lane	The lane which contains the activity that triggers the execution of the activity at hand or creates the data object as output.
target	Activity, Data Object	Lane	The lane, the execution of which is triggered by the activity at hand or uses the data object as input.
authentication_required	Activity, Data Object	Boolean	A flag indicating whether authentication is required for to the execution of the activity or the modification of the data object.
authorisation_required	Activity, Data Object	Integer	The level of authorisation required for the execution of the activity or the modification of the data object.
security_objective	Security Implementing Activity	Enum.	The type of security objective implemented by the activity.
integrity_required	Data Object	Boolean	A flag indicating whether the integrity of the data object needs to be ensured.
integrity_checked	Data Object	Boolean	A flag indicating whether the integrity of the data object has been verified.
confidentiality_required	Data Object	Boolean	A flag indicating whether the confidentiality of the data object needs to be ensured.
secure_channel[Lane]	Data Object	Boolean	A flag indicating whether a secure channel exists for communicating the data object to *Lane*.
availability_required	Data Object	Boolean	A flag indicating whether the availability of the data object needs to be ensured.

Table 3.7: Overview of BPMN security-related attributes used for security verification

capture can be categorised in two groups, workflow related and security related information.

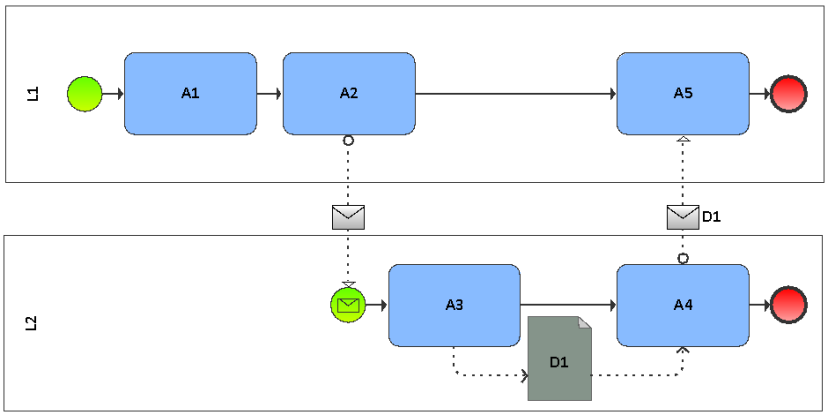

Figure 3.25: Example process fragment

The workflow-related information is captured by the *owner*, *source* and *target* attributes, attached to the concepts of Activity and Data Object. These attributes aim to capture information regarding the position of each instance of activities and data objects within the workflow of a business process model. More specifically, for the concept of Activity, the *owner* attribute indicates the lane of which this activity is part of, thus relating information regarding the entity in charge of the activity's execution. For instance in the example process fragment of Fig. 3.25, the attribute instantiation $A1.owner$ should return the value $L1$, since the activity with id $A1$ belongs to the lane $L1$. The *source* and *target* attributes capture the lanes which, respectively trigger or get triggered by the execution of the activity at hand, as dictated by the workflow of the business process. An example of the use of such attributes can be shown based on the process fragment of Fig. 3.25, where for the activity with id $A3$ the attribute declaration $A3.source$ returns $L1$. Similarly, for the activity with id $A2$ the attribute declaration $A2.target$ returns $L2$. As indicated by the multiplicity of the *source* and *target* attributes of the Activity concept in Fig. 3.24, there can be no source or target for an activity, in case it does not trigger or gets triggered by another lane (e.g., $A1.target = NULL$). It can also be the case that multiple sources or targets exist in case of workflow splits or joins due to gateways.

By comparing the *owner* attribute of an activity with its *source* or *target*, we can deduce whether the workflow of the process is transferred from one lane to another, which is information of high relevance for the analysis and verification of security properties. For instance, in the example of Fig. 3.25, if the lane where the workflow leads after the execution of activity $A2$ needs to be identified, we can

compare the attributes *A2.target* and *A2.owner*. The first part of this comparison (i.e., *A2.target*) returns lane *L1*, while the next part (i.e., *A2.owner*) returns *L2* as the lane that contains the activity which is triggered following the execution of *A2*.

The same applies for the *owner*, *source* and *target* attributes of the Data Object concept, with the only difference being that the source and target represent the lanes that contain activities that create the data object as output or use it as input. For instance, *D1.source* in Fig. 3.25 should return *L2*, since the activity which creates *D1* belongs to lane *L2* while *D1.target* should return both *L1* and *L2* as D1 is input for both activities *A4* and *A5* which respectively belong to lanes *L2* and *L1*.

The second group of attributes captures security needs and properties of the *Lane*, *Activity* and *Data Object* elements. More specifically, the attributes introduced in the Lane concept indicate whether or not the entity represented by such a lane has been authenticated and what is its level of authorisation. Such properties of a lane are vital for the verification of security properties, as they indicate whether the entity modelled by the lane can access certain activities or data objects. The Data Object concept includes a number of attributes in order to capture different types of security needs (e.g., *authentication_required, authorisation_required, confidentiality_required*). The attributes relating to the need of authentication and authorisation are also included in the Activity concept. Such attributes are used for identifying which types of security needs must be checked during the security verification. Other than attributes used to capture needs, the Data Object concept also includes attributes for capturing certain security-related properties, such as the existence of secure channels between the data object and a lane. Such properties are an important component of the security verification process, which will be presented in the next section.

Finally, other than the introduction of attributes to existing concepts, we have also introduced a new type of BPMN activity called *Security Implementing Activity*. Such a type of activity is concerned with the operationalisation of security at the process level by the implementation of security mechanisms and countermeasures. The type of security objective fulfilled by each security implementing activity is captured by its *security_objective* attribute, while a set of methods are available for allowing such activities to interact with the attributes of other process elements. The selection of appropriate security mechanisms is considered to be outside the scope of this work and so security implementing activities are considered as "black boxes". The security verification process proposed in this

work is, therefore, implementation agnostic and mainly concerned with the effect that the structural properties of a business process model have on the satisfaction of the security requirements of the process.

3.6.2 Attribute Instantiation and Security Verification

The attributes presented in Section 3.6.1 are utilised for the verification of security objectives. The process for the instantiation of such attributes and the algorithm used for the verification of each security objective will be presented in the rest of this section.

Authentication

Authentication is defined as the provision of assurance that a claimed characteristic of an entity is correct [117]. In the context of business processes, authentication entails the verification of a credential of a subject using security mechanisms [64]. The subjects of a business process are its participating entities, which can be, among others, individuals or groups of human participants, software systems or organisations. (Swim)lanes are used in BPMN 2.0 as a graphical representation of a participant in a business process model [7]. Therefore, authentication is a security objective associated with the lanes of a business process model.

To capture the authentication property of a process participant, the attribute *authenticated* has been introduced at the Lane concept, as illustrated in Fig. 3.24. Security implementing activities which operationalise the authentication security objective, as indicated by the value of their *security objective* attribute, can access the *authenticated* attribute of a lane l and set it to $TRUE$ using their *set_authentication(l)* method. The attribute *authentication_required* has been introduced to the Activity and Data Object concepts to capture whether they require participants to be authenticated before accessing them.

Algorithm 1 defines the steps for the verification of the authentication property of activities and data objects. The procedure AUTHENTICATION_CHECK_A takes an activity as input (line 1) and identifies all lanes that trigger the execution of the lane containing the activity (line 2). If such lanes are different than the lane in which activity at hand is contained and if such lanes are authenticated (line 3), then the authentication constraint of the activity is considered satisfied. Similarly, the procedure AUTHENTICATION_CHECK_DO takes a data object as input (line 9) and, for each of lanes having the data object as input (line 10), checks whether they are different than the lane which creates the data object and

Algorithm 1 Algorithm for authentication checking

```
1: procedure Authentication check A(Activity)
2:     for all Activity.source do
3:         if Activity.owner = Activity.source and
   (Activity.source).authenticated == TRUE then return TRUE
4:         end if
5:     end for
6: end procedure
7:
8: procedure Authentication check DO(DataObject)
9:     for all DataObject.target do
10:        if DataObject.owner = DataObject.target and
   (DataObject.target).authenticated == TRUE then
11:            return TRUE
12:        end if
13:    end for
14: end procedure
```

whether such lanes are authenticated (line 11).

Authorisation

Authorisation requires the restriction of access to assets based on certain business or security requirements of an entity [117]. In the context of a business process model, authorisation involves a lane, representing the entity that wants to access an asset, the authorisation level of that entity, and the asset itself, which can be either an activity or a data object [64].

A number of attributes have been introduced, as shown in Fig. 3.24, for the instantiation and checking of the authorisation objective. More specifically, the attribute *authorisation_level* is used for capturing the level of authorisation of each process lane. The attribute *authorisation_required* is used to capture the minimum level of authorisation required by an entity for accessing an activity or data object. Finally, security implementing activities with the *security objective* attribute set to *authorisation*, perform the *set_authorisation(l, v)* method to set the *authorisation level* of a lane l to a value v.

In the context of a business process model, authorisation checking, performed using Algorithm 2, involves following the workflow of the process to identify all the entities that interact with the authorisation-constraint process elements. In case of an authorisation-constraint activity, procedure AUTHORISATION_CHECK_A identifies each lane that contains activities that trigger the execution of the activity at hand (line 2). If such lanes are different than the owner lane of the constraint

Algorithm 2 Algorithms for authorisation checking

```
1: procedure Authorisation check A(Activity)
2:     for all Activity.source do
3:         if Activity.owner = Activity.owner then
4:             if (Activity.source).authorisation_level ≥ Activity.authorisation required then
5:                 return TRUE
6:             end if
7:         end if
8:     end for
9: end procedure
10:
11: procedure Authorisation check DO(DataObject)
12:     for all DataObject.target do
13:         if DataObject.owner = DataObject.owner then
14:             if (DataObject.target).authorisation_level ≥ DataObject.authorisation required then
15:                 return TRUE
16:             end if
17:         end if
18:     end for
19: end procedure
```

activity (line 3) and their authorisation level is greater or equal to the minimum authorisation level required by the constraint activity (line 4), the authorisation constraint is satisfied. In the case of a data object, a similar authorisation checking process is followed using the procedure AUTHORISATION_CHECK_DO but, in this case, each lane using the data object as input is identified (line 12). If such lane is different than the data object's owner lane (line 13), then the authorisation level of such lane is compared to the authorisation level required by the constraint data object (line 14) and if it is greater or equal the authorisation constraint is considered satisfied (line 15).

Confidentiality

Confidentiality refers to the protection of information from disclosure to unauthorised entities [136]. Therefore, in terms of business process models, confidentiality is a property of a data object, which is the concept BPMN 2.0 utilises to capture information assets. Defining confidentiality also requires the identification of authorised entities that can access the information [137]. Thus, the concept of a swimlane is, once again, required for the definition of confidentiality in the context of business processes.

Algorithm 3 Algorithm for confidentiality checking

```
1: procedure Confidentiality check(DataObject)
2:     for all DataObject.target do
3:         if DataObject.owner = DataObject.target then
4:             if (DataObject.target.authorisation_level ≥
   DataObject.authorisation required then
5:                 if DataObject.secure channel[DataObject.target] == TRUE
   then
6:                     return TRUE
7:                 end if
8:             end if
9:         end if
10:    end for
11: end procedure
```

A number of attributes have been introduced for reasoning about confidentiality in business process models, as shown in Fig. 3.24. The attribute *confidentiality required* introduced in the Data Object concept indicates whether the confidentiality objective has to be met for accessing a data object. The attribute *secure_channel[Lane]*, also introduced in the data object concept, indicates whether a communication channel capable of confidential data transmission exists between the data object and a specific entity, modelled as a lane in the business process. In order to establish confidentiality, appropriate security implementing activities need to be introduced in the business process. To that end, security implementing activities operationalising the confidentiality security objective (i.e., *security_objective* attribute is set to *confidentiality*) have the method *set_confidentiality()*. That method takes as input a confidentiality-constraint data object and a lane and, if a secure connection exists between them, assigns the value *TRUE* to the *secure channel[Lane]* attribute of the data object.

Algorithm 3 verifies whether the confidentiality objective of a data object is met by a business process model. The algorithm takes a data object as input and checks all the outgoing workflows using that data object (line 2). For each outgoing workflow leading to a lane that is different than the one currently owning the data object (line 3), the *authorisation_level* of that lane is compared to the minimum authorisation level required by the data object (*authorisation required* attribute of data object) (line 4). Finally, the existence of a secure communication channel between any authorised target lane and the data object is checked via the *secure channel[Lane]* attribute of the data object (line 5). If the attribute has a value of *TRUE* for each target lane then the confidentiality objective is satisfied.

Integrity

Integrity is concerned with ensuring that information is protected from improper modifications so as to avoid intentional or accidental unauthorised changes to system data [136]. Similar to confidentiality, the entities relating to integrity, in terms of business process models, are the data object, which models the data handled during the process execution, and the lane which models the entities exchanging said data.

As shown in Fig. 3.24, to capture aspects relating to integrity, the *integrity_required* and *integrity_checked* attributes have been introduced in the data object concept. When the *integrity_required* attribute has a *TRUE* value, an integrity constraint exists on the data object at hand, while if *integrity_checked* attribute is set to *TRUE* the integrity of the data object has been confirmed by appropriate security mechanisms. The activities modelling the operationalisation of such integrity implementing mechanisms are modelled as security implementing activities with their *security_objective* attribute set to *integrity*. To signify that the integrity checking has been performed, such activities include the method *check_integrity()*, which takes a data object as input and changes the value of its *integrity_checked* attribute to *TRUE*.

Algorithm 4 Algorithm for integrity checking

```
1: procedure Integrity_check(DataObject)
2:     for all DataObject.target do
3:         if DataObject.owner = DataObject.target and
   DataObject.integrity_checked == TRUE then
4:             return TRUE
5:         end if
6:     end for
7: end procedure
```

For the verification of the integrity objective of data objects in a business process model, Algorithm 4 has been developed. The algorithm takes as input a data object and identifies the lane of each activity that consumes the data object (line 2). If the data object's source lane is different than its target lane (line 3), which indicates that a data transfer between lanes has taken place, the *integrity_checked* value of the data object is checked (line 3). If the value is *TRUE* a successful integrity checking is assumed to have been executed, thus signifying the satisfaction of the integrity objective.

Availability

Availability describes the property of system resources being accessible and usable upon demand by an authorised entity [117]. Therefore, in terms of a business process model, a system resource, modelled as a data object, needs to be available to an authorised entity, modelled as a lane. To capture aspects relating to availability, the extended metamodel of Fig. 3.24 introduces the *availability required* attribute in the concept of Data Object, which indicates that such an element has an availability constraint placed upon it, if its value equals *TRUE*.

Algorithm 5 Algorithm for availability checking

```
1: procedure Availability check(DataObject)
2:     for all DataObject.target do
3:         if DataObject.owner = DataObject.target then
4:             if (DataObject.target).authorisation_level ≥
   DataObject.authorisation required then
5:                 if DataObject.source = IS UNIQUE then
6:                     return TRUE
7:                 end if
8:             end if
9:         end if
10:    end for
11: end procedure
```

The satisfaction of the availability constraint relates to the structure of the workflow of a process model. Since a data object needs to be available upon demand, there is a need for redundancy built into the workflow in order to ensure that there is always more than one ways to reach the availability-constraint process element. This means that an availability-constraint data object, for instance, should be able to be produced as the output of more than one activity. Therefore, to check the satisfaction of an availability-constraint data object we introduce Algorithm 5. This algorithm first checks if each activity requiring the data object (line 2) belongs to a lane different than the owner of the data object (line 3) and whether that lane has the appropriate authorisation for accessing it (line 4). Finally, it checks whether the constraint data object sources from more than one activity (line 5). If a value of *TRUE* is returned, the availability objective for said data object is satisfied.

3.6.3 Security Verification component Application

The business process model of the e-Prescription system, produced by the application of the previous steps of the framework, will be used as the input of the Security Verification component. The steps followed for the application of the Security Verification component are presented in Fig. 3.26.

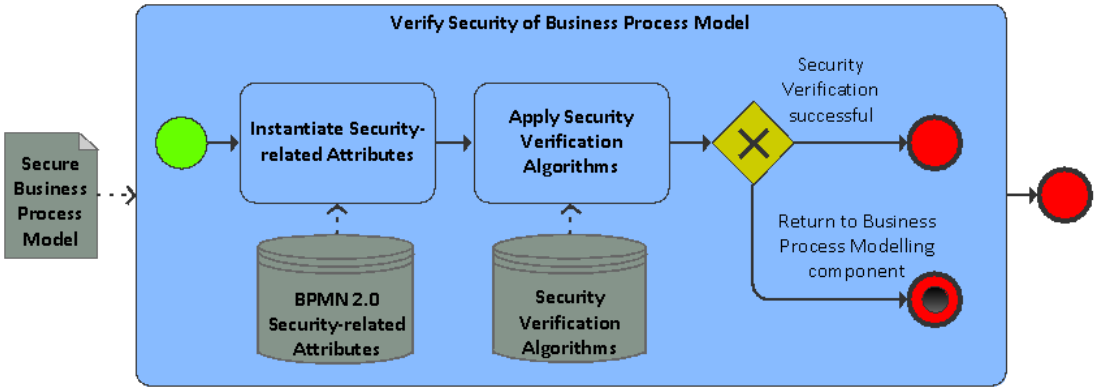

Figure 3.26: Activities for the application of the Security Verification component

The previous analysis of the system has identified three types of security requirements, namely confidentiality, integrity and authorisation. Security process patterns have also been introduced and instantiated within the created process model to satisfy such requirements. The application of the Security Verification component will examine whether the produced process model indeed satisfies the identified requirements. Figure 3.27 presents a fragment of the produced process model including the instantiated attributes of its relevant components.

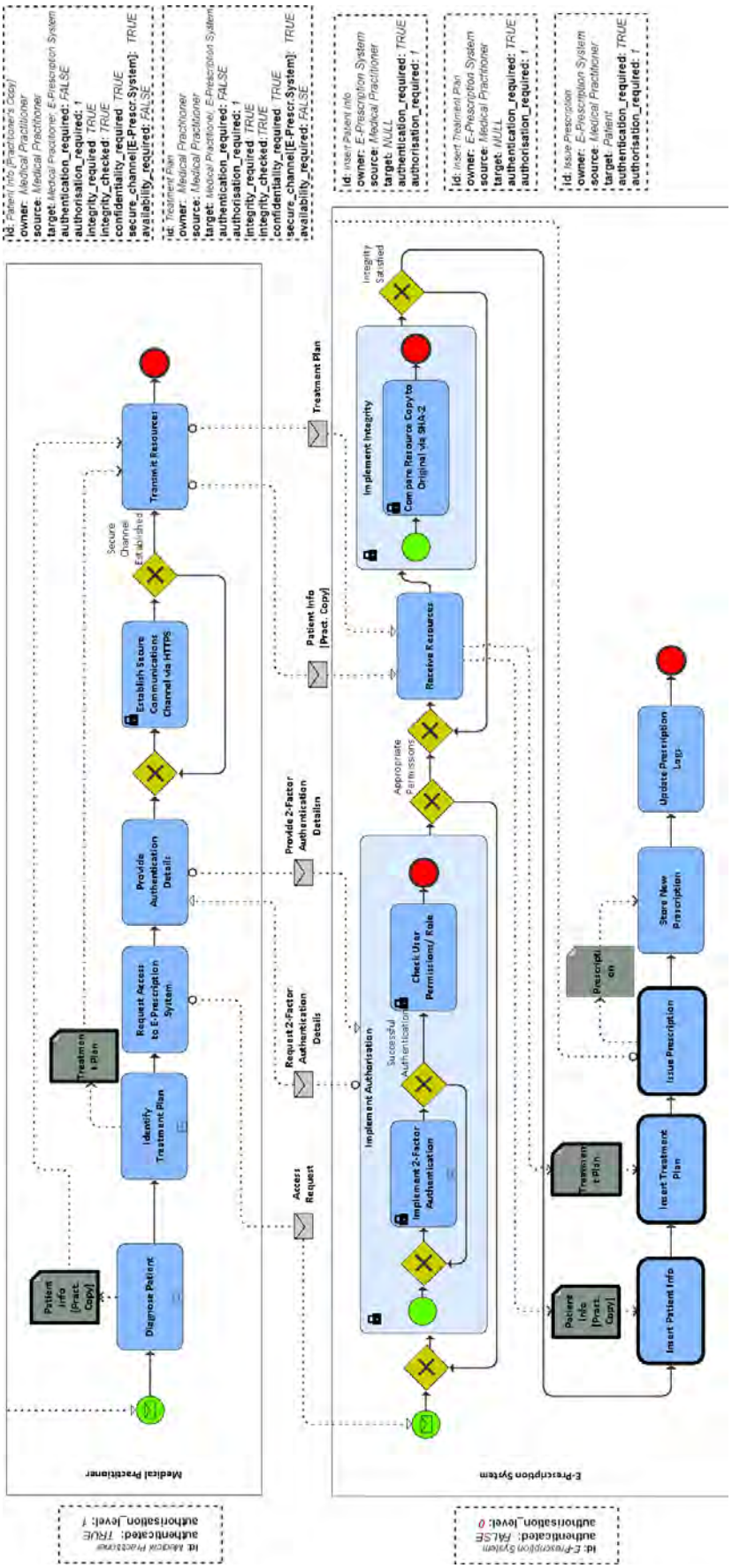

Figure 3.27: Process Fragment of e-Prescription System with Instantiated Verification Attributes

More specifically, the data objects "Patient Records" and "Treatment Plan", which are constrained by confidentiality and integrity, have their attributes instantiated to reflect such constraints (i.e., *confidentiality required = TRUE*, *authorisation required = 1* and *integrity required = TRUE*) and also capture their owner (Medical Practitioner), source (Medical Practitioner) and targets (Medical Practitioner, E-Prescription System). Similarly, for the authorisation constraint activities *"Insert Patient Info"*, *"Insert Treatment Plan"* and *"Issue Prescription"*, their security requirements have been captured (i.e., *authorisation_required= 1*) and their owner("E-Prescription System"), source ("Medical Practitioner") and target, if applicable (i.e., "Patient" for the "Issue Prescription" task) have also been instantiated.

The next part of the attribute instantiation process deals with the manipulation of the attributes of various components by the security-implementing activities introduced in the business process model. In detail, the *"Establish Secure Communication Channel via HTTPS"* activity of the *"Medical Practitioner"* lane operationalises the HTTPS security mechanism to achieve the objective of confidentiality and, as a result, establishes a secure communication channel between the data objects owned by *"Medical Practitioner"* and the *"e-Prescription System"* lane. To reflect that in the model's attributes the security implementing activity uses the methods *set confidentiality(Patient Records, e-Prescription System)* and *set_confidentiality(Treatment Plan, e-Prescription System)* to instantiate the attribute *secure channel[E-Prescription System]= TRUE* for both confidentiality-constraint information resources.

Similarly, the *"Compare Resource Copy to the Original via SHA-2"* activity of the *"E-Prescription System"* satisfies the integrity objective for the two data objects, using the methods *check integrity(Patient Records)* and *check integrity (Treatment Plan)* to set the *integrity_checked* attribute of both resources to *TRUE*. Finally, the *"Implement Authorisation"* sub-process of the *"e-Prescription System"* lane, uses the method *set authorisation(Medical Practitioner, 1)* to assign the appropriate authorisation level to the *"Medical Practitioner"* lane (i.e., *authorisation level=1*).

After the instantiation of all the relevant attributes, the verification algorithms for each security requirement can be applied at the process model to check whether its current composition satisfies the identified security requirements. For the verification of the confidentiality constraint satisfaction, Algorithm 3 was applied for data objects *"Patient Records"* and *"Treatment Plan"* (i.e., CONFIDENTIALITY CHECK(Patient Records, Treatment Plan)). In both cases the proce-

dure did not return a $TRUE$ result as the authorisation level of the E-Prescription System lane has not been established through a security-implementing activity and, as a result, it was not greater or equal to the authorisation level required for handling such data objects (Line 4 of Algorithm 3). Therefore, to fully satisfy the confidentiality constraint, authorisation has to be obtained for the e-Prescription system by the addition to the process model of an appropriate security-implementing mechanism.

The integrity checking algorithm (see Algorithm 4) was also applied for the same data objects (i.e., INTEGRITY_CHECK(Patient Records, Treatment Plan)), as there was an integrity constraint placed upon them at the E-Prescription system lane. The procedure returned $TRUE$ as a result, therefore the satisfaction of the integrity constraint was verified. Finally, the authorisation-constraint activities of the e-Prescription system lane were used as input to the authorisation checking algorithm (see Algorithm 2) to verify the satisfaction of their constraint. The procedures for all three activities (i.e., AUTHORISATION_CHECK_A(Insert Patient Info, Insert Treatment Plan, Issue Prescription)) all returned a $TRUE$ result, as their source lane (*"Medical Practitioner"*) had the appropriate authorisation level and therefore the constraint is considered as satisfied.

The application of the verification algorithms identified some security-related issues at the business process model of the e-Prescription system. The identification of such issues will prompt the system designers to update the business process design by reapplying the previous components of the framework. Thus, the Security Verification component provides valuable insights to system designers regarding the security of the process model during its design time. The component's contribution is not limited to its ability to identify potential security violations but to also pinpoint their location within the workflow of the process. Therefore, the Security Verification component can provide a structured way for ensuring the security of the process design produced through the application of the rest of the framework's components.

3.7 Software Support

The existence of software tool support is a critical aspect for the adoption of modelling approaches for the design of secure business processes, as highlighted by the evaluation of the literature of the area (see Section 2.4). To that end, both existing and purpose-built software tools are used to support the application of different parts of the framework presented in this chapter.

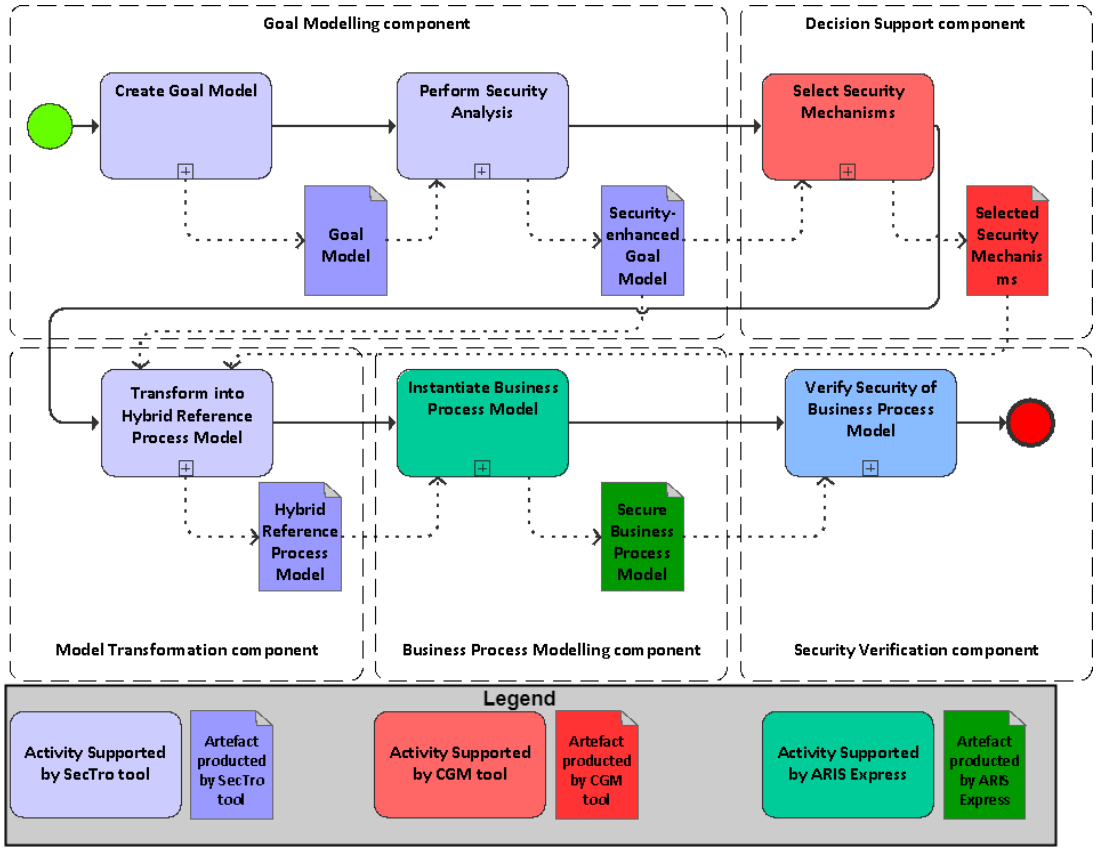

Figure 3.28: Software tool coverage of framework components

The software tools presented in the rest of this section either automate functionalities of the framework (e.g., model transformations, security mechanism selection) or provide the tools and graphical editors necessary for the creation of the intermediate and final modelling outputs (e.g., goal and business process models). The coverage provided by each software tool to each main activity of the framework is illustrated in Fig. 3.28. Despite the coverage provided by the software tools, certain aspects of the process supported by the proposed framework still require manual effort from users, as the software tools are not able to communicate with each other and share the created artefacts. Nonetheless, since

the development of deployable software tools is not within the scope of the current research project, holistic software tool coverage for the proposed framework will be a direction for future work.

3.7.1 Goal Modelling and Automated Transformation

SecTro[1] is a Security Requirements Engineering CASE tool built to support the construction of Secure Tropos models. SecTro supports the modelling and analysis of all of the different types of diagrams necessary for the application of the Secure Tropos approach. It provides a graphical editor for creating Secure Tropos models, automated analysis functionalities for verifying the consistency of the created models and an automated report generator for summarising the created models in textual format. Therefore, it is selected as the software tool of choice since it is able to fully accommodate the creation of the Security Requirements and Security Attacks modelling views of Secure Tropos, which are central artefacts created by the application of the Goal Modelling component of our framework.

The functionality of SecTro was extended, as part of this work, in order to also support the application of the Model Transformation component of our framework[2]. More specifically, a hybrid process view was introduced in the tool to accommodate the handling of hybrid reference process models. The BPMN 2.0 concepts, necessary for the creation of the hybrid reference process model supported by the newly created view (e.g., lanes, activities, data objects), were created and connected to the already existing metamodel within the SecTro tool. This way the proper modelling syntax (e.g. allowed connections between available modelling concepts) can be ensured when users create new models. An additional functionality was also added, allowing users to automatically create hybrid reference process models based on the Secure Tropos goal models they have already built in the Security Requirements view of the tool. To create that functionality, the transformation steps, as presented in Section 3.4.1, were implemented as algorithms developed in Java, which scanned the created Security Requirements view model and transformed the appropriate concepts into their hybrid reference process model counterparts. The complete transformation process was also bundled into a single tool command which, when selected, automatically updates the structure of the hybrid reference process model according to the structure of the

[1] Available for download at http://www.sense-brighton.eu/research/sectro-tool/

[2] Available for download at: http://www.sense-brighton.eu/research/sectro-tool/secure-business-process-sectro/

goal model of the Security Requirements view.

Therefore, the extended prototype of the SecTro tool fully automates the model transformation required for the application of our framework. Thus, a user can create a Secure Tropos goal model at the Security Requirements view of the tool by applying the Goal Modelling component of the framework and then automatically create a hybrid reference process model by selecting the transformation command introduced into the tool without the need of any additional manual input.

3.7.2 Prioritisation and Reasoning Tool Support

CGM-Tool[3] supports modeling and reasoning on Constraint Goal Models. It is a freely distributed CASE tool which encodes constraint goal models using the OptiMathSAT satisfiability solver [130]. Its functionalities include a graphical editor for the creation of constraint goal models, automated model consistency analysis and automated reasoning functionalities by encoding the model into an SMT formula which is solvable by OptiMathSAT. The CGM-Tool is selected due to its ability to support the application of the Decision Support component of the presented framework as it allows the definition of multiple variables (e.g., risk mitigation, cost, performance) that can be associated with nodes of the goal model (e.g., security mechanisms) and the definition of linear equations composed by such variables that can be optimised. This is done with the use of a scalable external reasoner, OptiMathSAT, which is invoked by the tool to identify optimal solutions for the linear equations over the modelled CGMs.

Therefore, a user can apply the Decision Support component by reconstructing the Secure Tropos goal model at the graphical editor of the CGM-Tool. Next, the variables associated with the selection of the security mechanisms (e.g., threat mitigation, constraint coverage, cost) can be defined and instantiated for each node that represents a security mechanism from the same graphical editor. Next, the optimisation scenarios can be created by defining and instantiating global variables within the created goal model (e.g., $ResidualRisk < 50\%$). Such global variables can also be prioritised by the user interface of the CGM-Tool and a model composition that satisfies them can be automatically generated by selecting the "Generate" command. Once the optimisation solver completes its execution on the background, the selected nodes (i.e., security mechanisms) are highlighted in the graphical editor and the final values of the global variables are presented to the user.

[3] Available for download at http://www.cgm-tool.eu/

3.7.3 Business Process Modelling Editor

A wide range of business process modelling editors, supporting BPMN 2.0, are freely available to users. For the purposes of this work, the ARIS Express modelling platform[4] has been used to support the Business Process Modelling component due to its ease-of-use and comprehensive support of the BPMN 2.0 modelling language. The Aris Express platform provides a graphical editor which fully supports the creation of BPMN 2.0 business process models. The security process patterns, developed as part of the Business Process Modelling component of our framework (see Section 3.5.1), have been modelled using this tool and are available as templates[5]. Moreover, all BPMN 2.0 business process models included in this work have been modelled using this platform.

The application of the Business Process Modelling component of the presented framework can be fully accommodated by ARIS Express. A user can recreate the hybrid reference process model using the graphical editor provided by the tool and introduce the appropriate security process pattern from the provided pattern templates. Each pattern can be manually instantiated to reflect the selected security mechanism and integrated to the constructed business process model. Finally, the user has to manually create the control flow of the process by connecting activities, creating message exchanges and introducing gateways and events, according to the syntax of BPMN 2.0 which is enforced by the ARIS Express tool.

[4] Available for download at: http://www.ariscommunity.com/aris-express
[5] Available for download at: http://www.sense-brighton.eu/process-patterns-questionnaire

Chapter 4

Evaluation

The developed framework has been evaluated throughout its development following an iterative "build and evaluate" approach. The development of a prototype of each framework component has been followed by its application to at least one real-life case study as a proof of concept. Such proof of concept applications, presented in Section 4.1, facilitated the incremental refinement of each component before its integration within the overall framework. Additionally, the security process patterns of the Business Process Modelling component were also evaluated via a workshop-based modelling exercise to assess their usability and comprehensibility, as presented in Section 4.2. The additional evaluation effort for that component was undertaken since it was developed from scratch as part of the current research project and as such, no previous attempt for its evaluation had been performed. Finally, at the later stages of the research project a large-scale evaluation of the overall framework was performed via a case study, presented in Section 4.3. An e-government system was selected and the developed framework was applied, in close cooperation with system stakeholders, for the development of a secure business process. Both quantitative and qualitative insights from the large-scale framework application through the case study were collected via previously defined metrics and stakeholder interviews. The rest of this chapter presents the different evaluation efforts undertaken as part of this research project and concludes with discussion regarding the lessons learned from such attempts.

4.1 Proof of Concept Applications

A number of proof of concept applications of the framework's components have been performed through the publications (see Section 1.6) produced during this

research project. These publications present the evaluation of prototypes of individual framework components through their application in small-scale examples. These proof of concept applications facilitated the identification of limitations which led to gradual refinements of the studied components. An overview of the small scale evaluation of the framework's components will be provided in the rest of the section while a discussion for the overall lessons learned will follow in Section 4.4.

An initial version of the model transformation process, which involves the Goal Modelling, Model Transformation and Business Process Modelling components was introduced in [8]. A fragment of the e-Prescription system was used to illustrate its functionality which was mainly focused in the application of an early version of the transformation steps. The same components were also applied in the context of legacy business processes in [9], where the transformation steps were utilised to produce an updated and secure version of the business processes supporting a personal financial application. Through those initial proof of concepts applications, focusing on the transition between goal and business process models, the transformation steps were incrementally refined and later utilised in the context of software product lines in [37], where the process model produced by the transformation of a Secure Tropos goal model was used as the main input for extracting variable, run-time service configurations for a water management system. The same version of the model transformation process was also included in [38], where the produced business process model was used as the input for a framework that produced secure, cloud-based system used by a University for conducting graduate surveys.

The collection of security process patterns used by the Business Process Modelling component, presented in Section 3.5.1, was introduced in the work presented in [40]. An initial version of the security process patterns was presented and applied to the e-Prescription system example. Additionally, the introduced set of patterns was evaluated via a workshop modelling session, as part of this work, as discussed in Section 4.2. The feedback received from this work led to the further refinement of the patterns, followed by a second round of workshop-based evaluation to further solidify our findings, as presented in [44]. The Decision Support component was first introduced in [43] and applied in the e-Prescription system example. The same work was later extended to include a refined version of the risk calculation formulas in [45]. Finally, the Security Verification component was introduced in [41] where it was also applied in a simplified version of a public swimming pool administration system. As a result of this application the

verification algorithms were further refined to the version presented in this work (see Section 3.6).

While the proof of concept applications of the different components, performed through the above publications, do not constitute a large-scale and exhaustive evaluation, they provided useful insights for the further development of the overall framework. The lessons learned from each of the above works facilitated the further refinement of individual aspects of the framework, before it was evaluated as a whole through a large-scale case study (see Section 4.3). Moreover, these small-scale applications of the framework proved its ability to provide meaningful support and analysis capabilities in a diverse range of real life contexts. Finally, the combination of the developed framework with works in the areas of software product lines and cloud-based systems, highlighted its flexibility, as it was able to produce useful artefacts that were used as input for the application of other specialised approaches.

4.2 Workshop-based Modelling Exercise

A workshop-based modelling exercise was conducted for the evaluation of the newly developed security process paterns (see Section 3.5.1). More specifically, the exercise aimed to i) evaluate the perceived understandability and ease-of-use of the proposed security process patterns and ii) compare their implementation to ad-hoc security integration in business process models.

4.2.1 Exercise Setup

Overall, thirty (30) postgraduate students (MSc and PhD level) from two different universities (i.e., University of Brighton, UK and Pantheon-Sorbonne University, France), in the areas of information systems design and information security, participated in two separate supervised workshop sessions, each with a duration of approximately thirty minutes.

A brief introduction to familiarise the participants with business process modelling concepts and BPMN diagrams was provided at the beginning of each session. Next, a brief business process model, shown in Fig. 4.1, was presented to the participants.

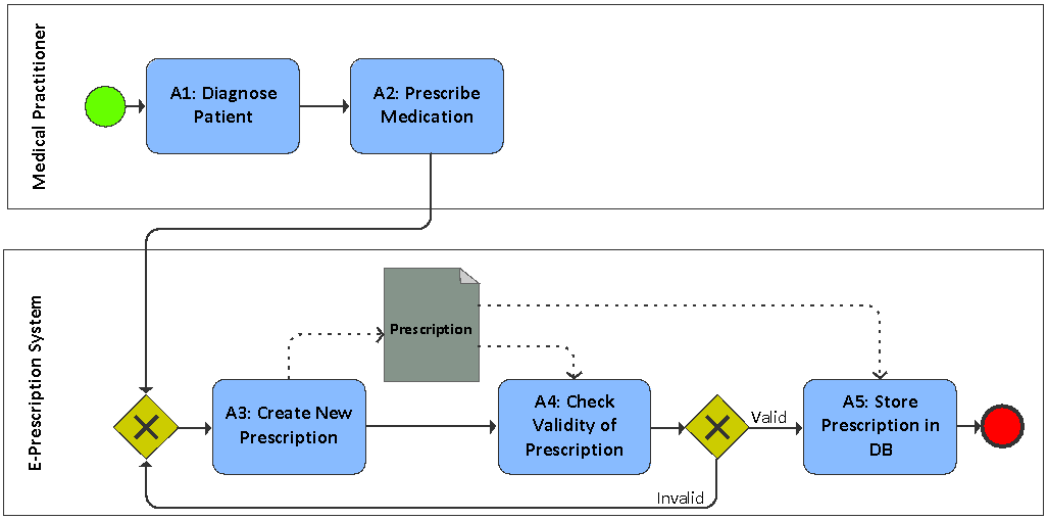

Figure 4.1: Business Process Model of Evaluation Experiment

During the first scenario the participants were asked to redesign the provided process model by introducing any activities they considered necessary, in an ad-hoc manner, in order to satisfy the authentication constraint *"Only registered medical practitioners can create a new prescription"*. Only after the first scenario was completed, the participants were presented with the authentication pattern, as introduced in Fig. 3.17. For the completion of the second scenario, they were asked to instantiate and introduce the pattern to the business process model of Fig. 4.1, in order to, once again, satisfy the same security constraint.

4.2.2 Exercise Results

After both parts of the exercise were completed a short questionnaire was distributed in order to capture the opinions of the participants regarding their experience. The questionnaire entries were phrased as statements accompanied by a 5-point Likert scale, ranging from strongly disagree to strongly agree, from which the responders selected the option best reflecting their opinion. The statements provided to the participants were the following:

- "I found it difficult to identify which activities I needed to add to the process model (Fig. 4.1) in Scenario 1."
- "I found it easier to create a business process model in Scenario 2 than in Scenario 1."
- "The contents and structure of the business process pattern (Fig. 3.17) were easy to understand."

- "I found it easy to integrate the business process pattern into the business process of Fig. 4.1."

At the end of the questionnaire form there was also the option of providing free-form comments and remarks[1].

The participants' responses to the above statements are summarised as follows:

- 10 out of 30 (33%) either agreed (9) or strongly agreed (1) that it was difficult to identify the security related activities needed to be added in the process, in an ad-hoc manner.

- 15 out of 30 (50%) either agreed (10) or strongly agreed (5) that it easier to create a secure business process model using the provided process pattern compared to the ad-hoc security implementation.

- 20 out of 30 (66%) either agreed (15) or strongly agreed (5) that the provided process pattern was easy to understand,

- 18 out of 30 (60%) either agreed (13) or strongly agreed (5) that the provided process pattern was easy to integrate to the provided business process model.

The modelling exercise allowed us to get an indication of the perceived usability and understandability of the proposed process patterns. It also indicated that such patterns are a preferable alternative to ad-hoc approaches, thereby confirming the literature consensus that patterns provide more structure and guidance to process designers. Another insight gained from this modelling exercise was that even non-experts in the area of information security were able to sensibly make use of the provided patterns in order to create consistent models within a reasonable timeframe. This indication is also aligned with literature findings, suggesting that patterns facilitate reusability and model consistency while also reducing the overhead for process designers in terms of time and prerequisite domain knowledge.

4.2.3 Threats to Validity

The main threat to the validity of the workshop-based evaluation of the security process patterns is concerned with the generalisability of the modelling exercise's

[1]The questionnaire and a summary of the responses can be accessed in: http://www.sense-brighton.eu/process-patterns-questionnaire/

results. Since the participants only worked with a small subset of the proposed patterns and a simple process model the generalisability of the workshop's conclusions is limited. Another aspect that has to be considered is the potential of bias introduced by learning effects, since the participants familiarised themselves with the process model of Fig. 4.1 during the first scenario, thus, potentially making it easier for them to apply the pattern in the same model during the second scenario. Other threats to validity include the diverse backgrounds of the participants, since their information security and business process modelling experience varied, while also English was not the native language of a number of participants. Nonetheless, to minimize the effects of such factors, the workshop sessions, during which the exercise was performed, were supervised and any participant enquiries regarding the modelling exercise were answered.

4.3 Case Study

Case studies constitute a common approach for empirical evaluation in the field of information systems research [33]. The objective of the case study presented in this chapter is to identify whether the use of the developed framework is able to facilitate the creation of secure business process designs that describe a real-life, large scale information system. Even though individual components of the framework have already been applied at small scale examples throughout the development process (see Section 4.1), a large scale empirical evaluation will provide us with unique insights regarding its overall applicability and effectiveness.

4.3.1 Case Study Process

According to [34] the process for designing and executing a case study involves five basic steps.

1. *Case Study Design*, where objectives are defined and the case study is planned. In this case the overall objective of the case study is to iden- tify whether the developed framework is able to produce secure business process designs when applied to a real life information system. The selected system and the stakeholders involved in this case study are discussed in the next section.

2. *Preparation for Data Collection*, where the data collection procedures are defined. In our case data is collected during the application of the framework's component to the studied system. This is performed in close coop-

eration with some of the system's stakeholders following a specific set of steps for the application of the developed framework. In addition to that, a number of quantitative metrics are also defined to provide us with conclusions regarding the framework's effectiveness, as presented in the next section.

3. *Collecting Evidence*, where data is collected from the studied system during the execution of the case study. For the purposes of the case study presented in this work, this step involves the application of our framework to the studied system for the creation of different system models, as presented in Section 4.3.3.

4. *Analysis of Collected Data*, where the data is analysed for the extraction of conclusions. In this case study this step includes a qualitative evaluation of the framework's application through a semi-structured interview with the involved stakeholders, as well as the evaluation of certain quantitative metrics.

5. *Reporting*, where the results of the case study are summarised in order to draw conclusions. In our case, the reporting consists of a brief discussion of the main points raised by the stakeholders during their exit interview and the results of the metrics evaluation, as presented in Section 4.3.4.

4.3.2 Case Study Settings and Design

The case study selected for the application of the developed framework involves an e-government system of the Municipality of Athens, Greece. More specifically, the selected system is used for the administration of swimming pool facilities used by Athenian citizens and has been a part of the VisiOn [2] European project, in which the lead supervisor of this work participated. The author was not a part of the project but gained access to some of its participants and deliverables towards the later stages of the project for the purposes of this case study.

The case study was developed and performed in close cooperation with two analysts of DAEM S.A.[3], the organisation in charge of developing all information systems for the municipality of Athens. Both of them were experts in system analysis and design, while one of them was also a security expert. Both of them were familiar with goal modelling, security requirement elicitation with Secure

[2] http://www.visioneuproject.eu/
[3] http://www.daem.gr

Tropos and process design using BPMN due to their previous participation at the VisiOn project. The communication of the stakeholders with the author initiated during June of 2017 and regular teleconferences were performed until the completion of the case study in September of the same year. Since the case study participants were also occupied in other professional engagements during that period, the teleconferences were held twice or three times per month with some attended only by one of the two participants, with the exception of August when no meeting was held. Supplementary communication was performed via email in order to exchange information, answer short questions and arrange further teleconferences. A semi-structured interview was held after the end of the case study, in October of 2017, to document the experiences and insights of the participants. The deliverables produced in collaboration with the case study participants throughout the application of each step of the proposed framework are available as supplementary material in the Appendix section at the end of the document. The rest of this section presents only the final deliverables of each step.

The steps followed in order to elicit information about the system and apply the framework steps during the course of the case study, are as follows:

1. An initial discussion was held with the stakeholders to provide them with a high-level overview of the framework, explain the goals of the case study and initiate communications.

2. A description of the studied system is provided by the stakeholders via teleconferencing, providing details about the participants of the system, their main goals and their interdependencies.

3. An initial draft version of a Secure Tropos goal model is created andsubmitted to the stakeholders for feedback.

4. The goal model is refined according to the received feedback, until an accurate system representation is captured, as per the stakeholders' instructions.

5. The security requirements of the system are elicited after communication with the stakeholders, threats and security mechanisms are identified in coordination with the security expert and the Secure Tropos goal model is updated accordingly.

6. The decision support process is performed with the stakeholders via teleconferencing, the security expert assists in the quantification of the different

parameters while the system analyst is in charge of selecting the final security implementation scenario.

7. The transformation of the final Secure Tropos goal model to a hybrid reference process model is automatically performed by the SecTro CASE tool.

8. The refinement of the hybrid reference process model to a complete business process model is performed in cooperation with the system analyst via teleconferencing. After some iterations a final business process model is created and presented to both stakeholders for their approval.

9. The security properties of the created business process model are verified by the application of the verification algorithms. The verification results are presented to the stakeholders.

10. Final adjustments are made to the business process model in order to successfully pass the security verification process.

The data collected through the use of the framework was then analysed both qualitatively and quantitatively. The exit interview with the involved stakeholders of DAEM provided us with qualitative insights regarding the perceived applicability and effectiveness of the framework. Additionally, a series values for metrics were calculated to provide quantitative insights regarding the framework's performance in this case study.

More specifically, the quantitative metrics, which will be calculated at the end of the case study, will measure the conformance of the produced business process model to the specifications of the initial goal model. In more detail, the specified metrics are the following:

- *Functional Conformance* will be used to evaluate the functional elements of the goal model which have been also captured in the final business process model. More specifically the maximum functional conformance will be achieved if (i) each actor of the goal model is captured by at least one lane in the business process, (ii) all goals of each actor are operationalised by activities within its corresponding lane, and (iii) all resources of each actor are captured by data objects within its corresponding lane. Such measurements will provide an indication of the conformance of the produced business process model to the goal model, which contains the information initially used to identify the structure of the system.

- *Security Conformance* will be used to evaluate the security-related information elicited at the goal model which was also operationalised at the final business process model. More specifically, the metric will take into account (i) whether each of the security constraints elicited for each actor of the goal model was operationalised in the actor's corresponding lane, (ii) whether all of the actors' security-constrained elements (i.e., goals, plans, resources) were also modelled as secured elements (i.e., activities, data objects) within the actor's corresponding lane, and (iii) the amount of security constraints that were successfully verified at the first iteration of the business process model. The above comparisons will reveal the conformance of the final business process model to the security-related aspects elicited at the initial goal model.

The quantitative metrics defined above will help us evaluate how well the proposed framework deals with transferring information between the different levels of abstraction. If the business process model, produced as a result of the application of the model transformation process, conforms to the structural and security-related information captured at the goal model level, then we can assume that the framework can reliably transfer relevant information from the organisational to the operational level of abstraction. Other metrics could be considered to evaluate relevant aspects of the produced business process model (e.g., complexity, size) but since the studied system is yet to be implemented there is no baseline to compare them against. Thus, the information that could result from such metrics would offer no meaningful conclusions in the context of this case study. Nonetheless, as discussed in Section 5.3, in future research attempts, if the framework is evaluated using a legacy information system, such metrics can be used to compare the business process produced as a result of the framework's application against an existing baseline. Therefore, for the purposes of the case study presented in the rest of this chapter we will use the quantitative metrics discussed above to evaluate the completeness of the model transformation process and the qualitative feedback provided by the involved system stakeholders to extract further insights regarding other aspects of the proposed framework (e.g., ease-of-use, understandability).

4.3.3 Framework Application

Over the rest of this section, the application of our framework to the swimming pool administration system will be described in full detail, along with the produced intermediate and final modelling outputs.

System Description

The Swimming Pool Administration (SPA) system aims to support the registration of Athenian Citizens to municipal swimming pool facilities. In order for a citizen to complete the registration process a number of documents have to be issued by different entities. A local clinic has to issue a medical certificate after examining the citizen. The issued certificate is then forwarded by the clinic to the Municipality of Athens Citizen Support (MACS) information system. The MACS system is accessible by registered Athenian citizens and allows the storage, issuing and distribution of citizen certificates to different municipal agencies. Using the MACS system, a citizen can issue a birth and residency certificate, which, bundled with the medical certificate, can be forwarded to the Sports Facility Information system for the registration process to begin. An administrator of the sports facilities manually checks the validity of the received certificates and authorises the creation of a citizen account in the sports facilities' information system. Once the registration is completed, a badge is issued and delivered to the citizen, which can be used for accessing the sports facilities.

Security Constraint	*Security Objective*	*Affected System Elements*
Citizen data shall remain confidential	Confidentiality	AMKA, Bank Account Details
Medical certificate contents shall remain confidential		Medical Certificate
Certificate contents shall not be be disclosed during transfer		Citizen Certificate Certified Copies
Certificate copies shall not be modified after issuing	Integrity	Medical Certificate, Citizen Certificate Certified Copies
Certificate copies shall not be modified		Citizen Certificate Certified Copies
Request shall originate only from authorised users	Authorisation	Receive request for certificates
Personal data shall be accessed only by authorised citizens		Retrieve citizen data
Citizen info shall be handled only by authorised personnel		Registration Approval Form, Bank Account Details

Table 4.1: Security requirements of the Swimming Pool Administration System

Security Requirements Elicitation

The security requirements of the SPA system, as presented in Tab. 4.1 were elicited in the form of sets of security constraints and security objectives, as identified by the system's stakeholders and captured in collaboration with the experts of DAEM. All the resulting Secure Tropos modelling views were created using the SecTro[4] CASE tool.

For each of the identified constraints, the security expert of DAEM initially proposed a high-level type of security mechanisms (e.g., Encryption, File Verification). Next, after some further refinement, alternatives in the form of specific security mechanisms were identified for each of the types of security mechanisms (e.g., HTTPS or Private VPN for Encryption). The final Security Requirements view diagram of Secure Tropos, containing all actors, their goals, resources and interdependencies as well as the security related concepts (i.e., security constraints and mechanisms) for the SPA system are illustrated in Fig. 4.2, while early draft versions of the same diagram are included in the Appendix section.

In addition to the security constraint and mechanism identification, threats were also identified during the security analysis. More specifically, in cooperation with the security expert, three threats were identified (i.e., Man-in-the-Middle, Data Tampering and Account Hijacking) and connected to the elements of the system they can impact. Using the Security Attacks view of Secure Tropos, we were able to further analyse each threat and identify its attack methods and connect security mechanisms with the system vulnerabilities they protect against. The Security Attacks view diagrams in Figs. 4.3,4.4 and 4.5 illustrate that analysis.

[4] http://www.sense-brighton.eu/research/sectro-tool/

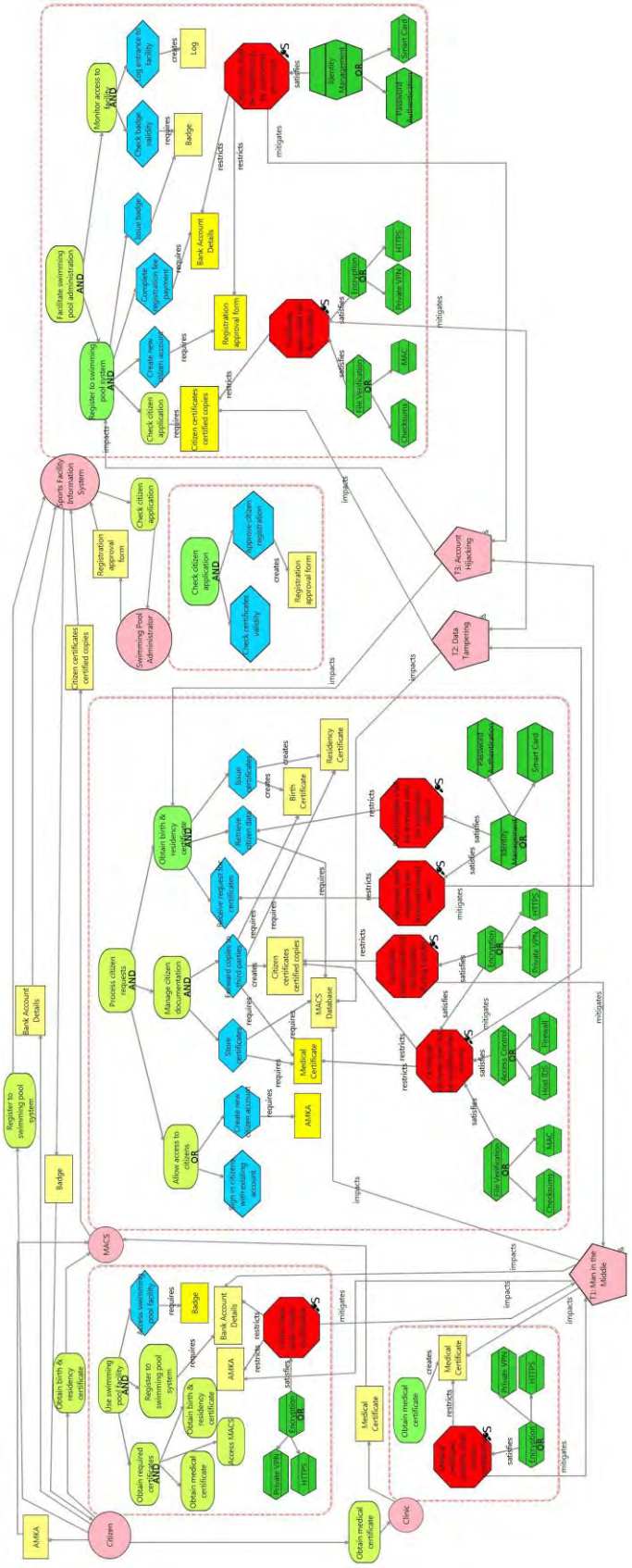

Figure 4.2: Security Requirements view model of the SPA system

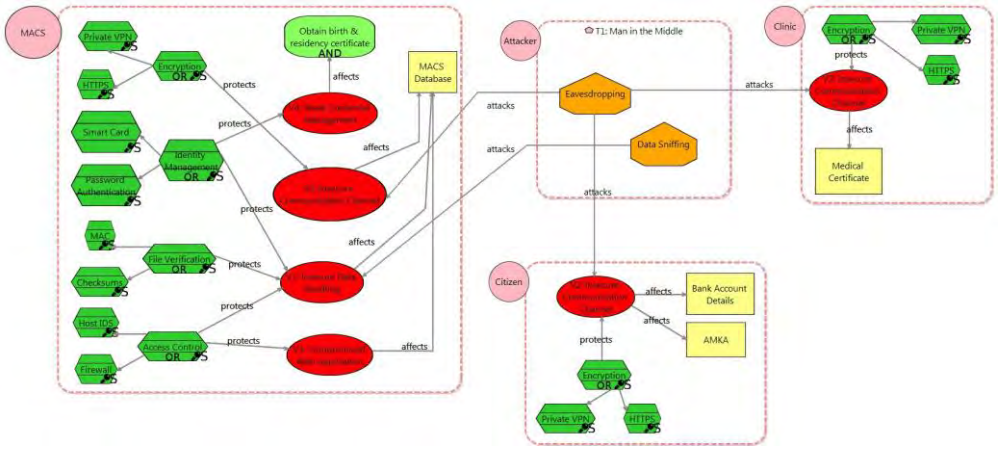

Figure 4.3: Security Attacks view model of threat T1 of SPAsystem

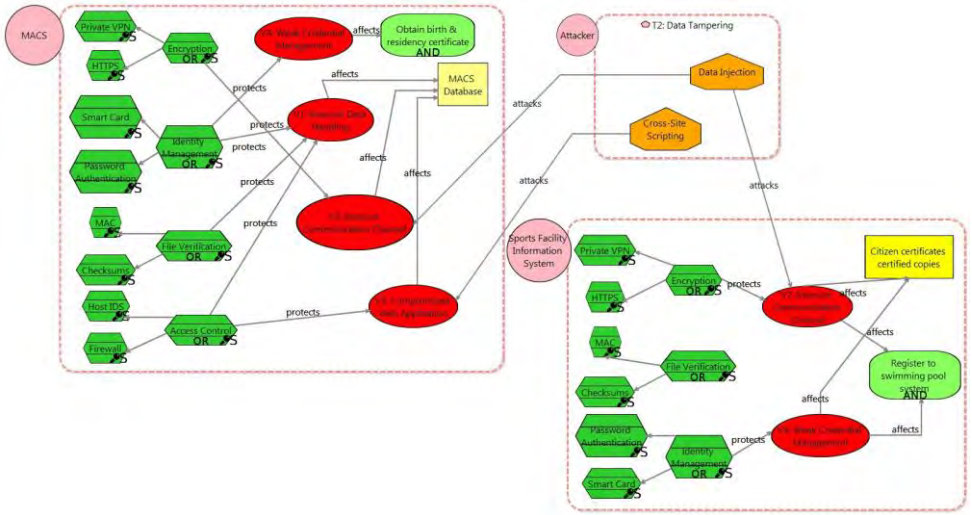

Figure 4.4: Security Attacks view model of threat T2 of SPA system

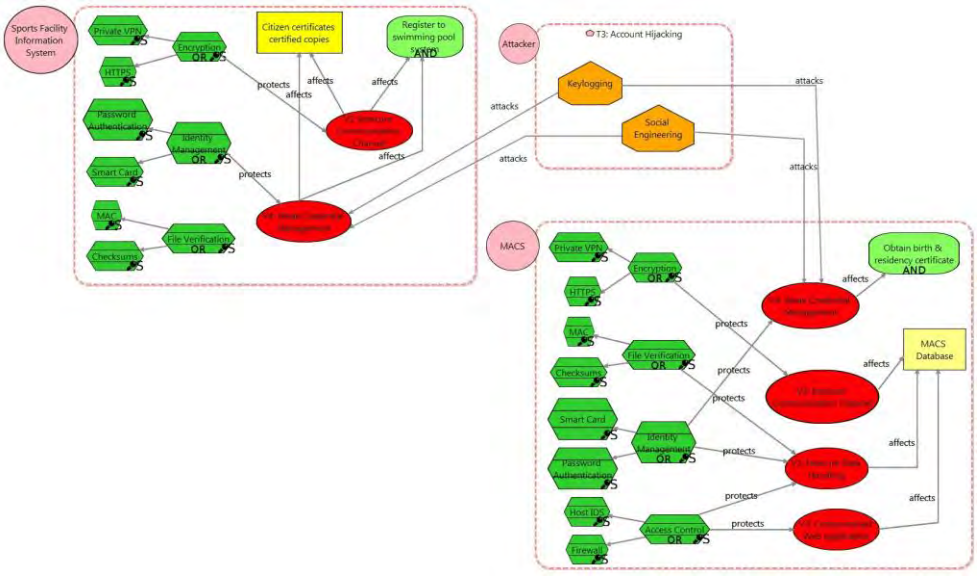

Figure 4.5: Security Attacks view model of threat T3 of SPA system

Decision Support Process

In order to select the security mechanisms that will be operationalised at the final business process model, the decision support process was performed in co-operation with the system analyst and security expert of DAEM. The CGM[5] CASE tool, which utilises the OptiMathSAT satisfiability solver was utilised for supporting the whole process.

First, the parameters according to which the mechanism selection will be performed were identified by the system analyst. In addition to the standard security and risk related parameters (i.e., security constraint satisfaction and risk mitigation), we also included the implementation cost and performance as additional non-functional parameters. Next in cooperation with the security expert values were assigned for the impact and likelihood of each threat's vulnerabilities using AHP. More specifically, an accurate ranking of the vulnerabilities' impacts and likelihoods was created by consulting various online resources (e.g., CVE[6], CVSS[7]), in cooperation with the security expert of DAEM. Next, following a similar process, constraint coverage, vulnerability mitigation, cost and performance coverage values were assigned to each of the identified security mechanisms.

Since all parameters were defined and all mechanisms instantiated with values, the next step required the definition of the optimisation process. To provide

[5] http://www.cgm-tool.eu/
[6] http://www.cvedetails.com/
[7] https://www.first.org/cvss/

Variable	Scenario 1	Scenario 2	Scenario 3
$R_{R(T1)}$	< 50%[3]	< 33%[1]	< 50%[3]
$R_{R(T2)}$	< 50%[4]	< 33%[2]	< 50%[4]
$R_{R(T3)}$	< 50%[5]	< 33%[3]	< 50%[5]
S_{Int}	> 50%[6]	> 50%[6]	> 50%[6]
S_{Auth}	> 50%[7]	> 50%[7]	> 50%[7]
S_{Conf}	> 50%[8]	> 50%[8]	> 50%[8]
$Cost$	min[1]	min[4]	min[2]
$Perform.$	max[2]	max[5]	max[1]

Superscripts next to variable values (e.g., [1], [2]) indicate their optimisation priority.

Table 4.2: Overview of optimisation scenarios for the SPA system

a wider range of choices for the system stakeholders, it was decided that different optimisation scenarios should be created. An overview of the variable thresholds and priorities for each scenario is provided in Tab. 4.2. In that table, the $R_{R(T)}$ values represent the residual risk of each identified threat, the S values represent the percentage of satisfaction of each security constraint and the $Cost$ and $Performance$ variables represent the non-functional system goals.

The resulting security mechanism combinations for each scenario are presented in Tab. 4.3.

- The first scenario represents a system configuration where cost reduction is the top priority, while a mid-level risk mitigation (i.e., residual risk is at least 50% less than the inherent) and security constraint satisfaction are achieved.

- The second scenario is focused on risk reduction, therefore stricter thresholds are set for accepted risk (i.e., residual less than 33% of inherent risk) and the residual risk values of each threat are set as the top optimisation priority. The rest of the parameters have the same thresholds and priorities as in the first scenario.

- Finally, the third scenario represents a system configuration where performance maximisation is the top priority of the stakeholders. The thresholds for accepted residual risks and security constraint satisfaction are set at mid-level, similar to the first scenario.

The stakeholders of the SPA system selected the first optimisation scenario, as the overall implementation cost was their most important concern and the risk reduction provided by that scenario was deemed adequate for the specific

	Scenario 1	Scenario 2	Scenario 3
Encryption	HTTPS	PrivateVPN	HTTPS
Access Control	Host IDS	Host IDS	Firewall
File Verif.	Checksums	Checksums	Checksums
Identity Mgmt.	SmartCard	SmartCard	Password

Table 4.3: Security configurations per scenario for the SPA system

system. Therefore, the security configuration described in the column "Scenario 1" of Tab. 4.3, will be implemented in the SPA system.

Model Transformation

To transition from the high level of system analysis provided by the SPA system's goal model to an operational level of abstraction, we applied the model transformation component of our framework.

The model transformation component uses the security requirements view diagram of the system (see Fig. 4.2) as input and creates the hybrid reference process model of Fig. 4.6 as output. The transformation is automatically performed using the SecTro CASE tool, so no additional input from the system's stakeholders was required. The hybrid reference process model, produced as the output of this step, is the skeleton upon which the final business process model describing the SPA system's functionality, will be built by applying the next components of the framework.

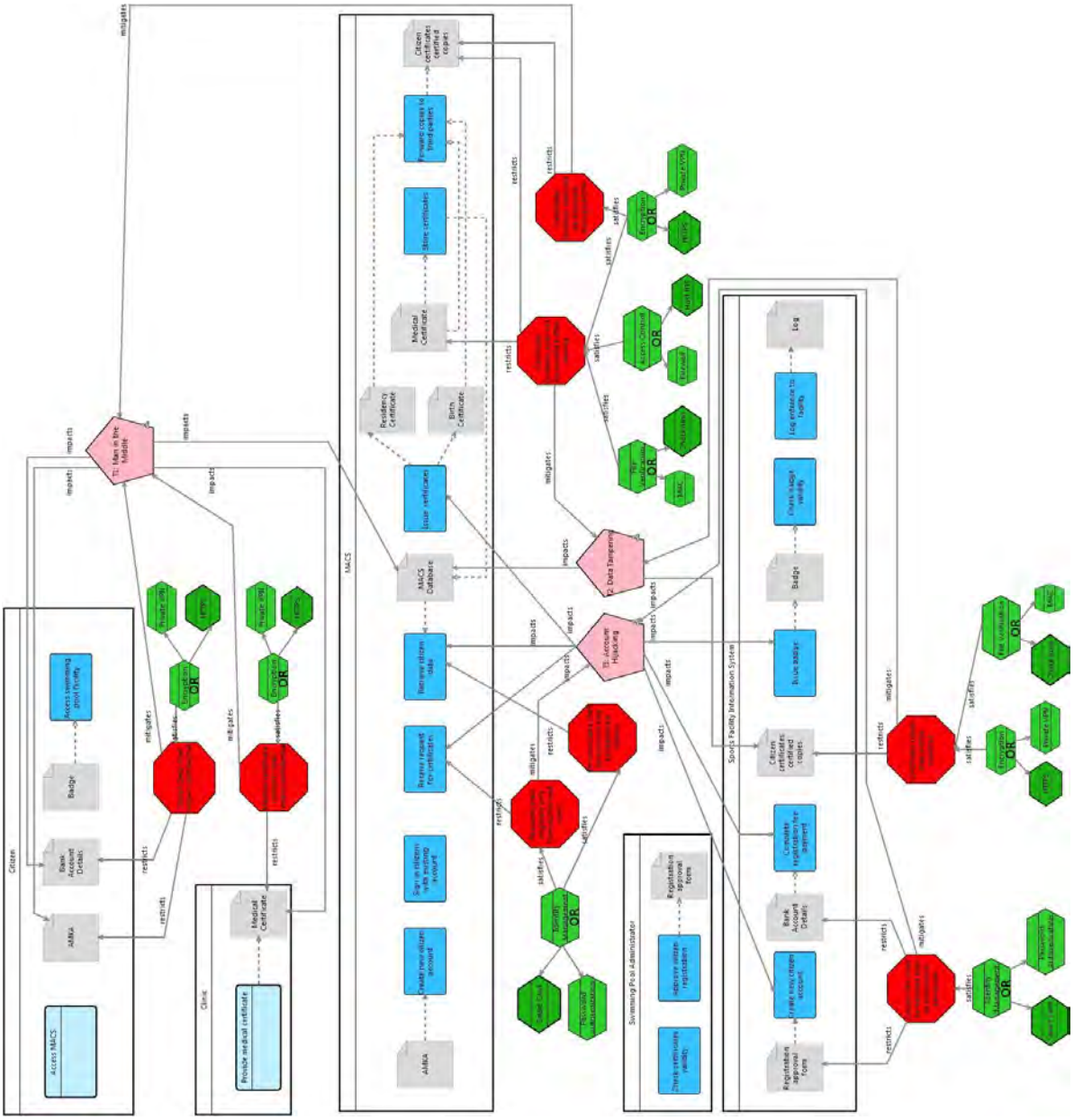

Figure 4.6: Hybrid reference process model of the SPA system

Business Process Model Refinement

The BPMN 2.0 Collaboration diagram that describes the SPA system was constructed in close cooperation with the analysts of DAEM. The automatically generated hybrid reference process model allowed us to identify: i) the basic structural characteristics of the process (lanes, activities, information objects), ii) the types of security constraints and the specific process elements they restrict

and, iii) the security mechanisms to be implemented to satisfy each constraint.

First the business process design patterns, presented in Section 3.5.1, were made available to the analysts. Next we matched each security constraint to its corresponding pattern. For instance the security constraint "Certificate copies shall not be modified after issuing", will be operationalised by the Integrity pattern (see Fig. 3.20) which will be instantiated by the Checksum security mechanism, as selected during the decision support process. The instantiated patterns were manually introduced into the business process diagram, for each constraint activity or data object.

Next, a manual refinement of the process model was performed which focused on introducing control flow elements, such as start and end events, gateways, additional activities and message exchanges between lanes. After some iterations which are available at the Appendix section, a final version of BPMN 2.0 collaboration diagram describing the functionality of the SPA system, as presented in Fig. 4.7, was delivered to the system analysts of DAEM for their final approval. The creation of the model was performed using the ArisExpress[8] modelling tool.

[8] http://www.ariscommunity.com/aris-express

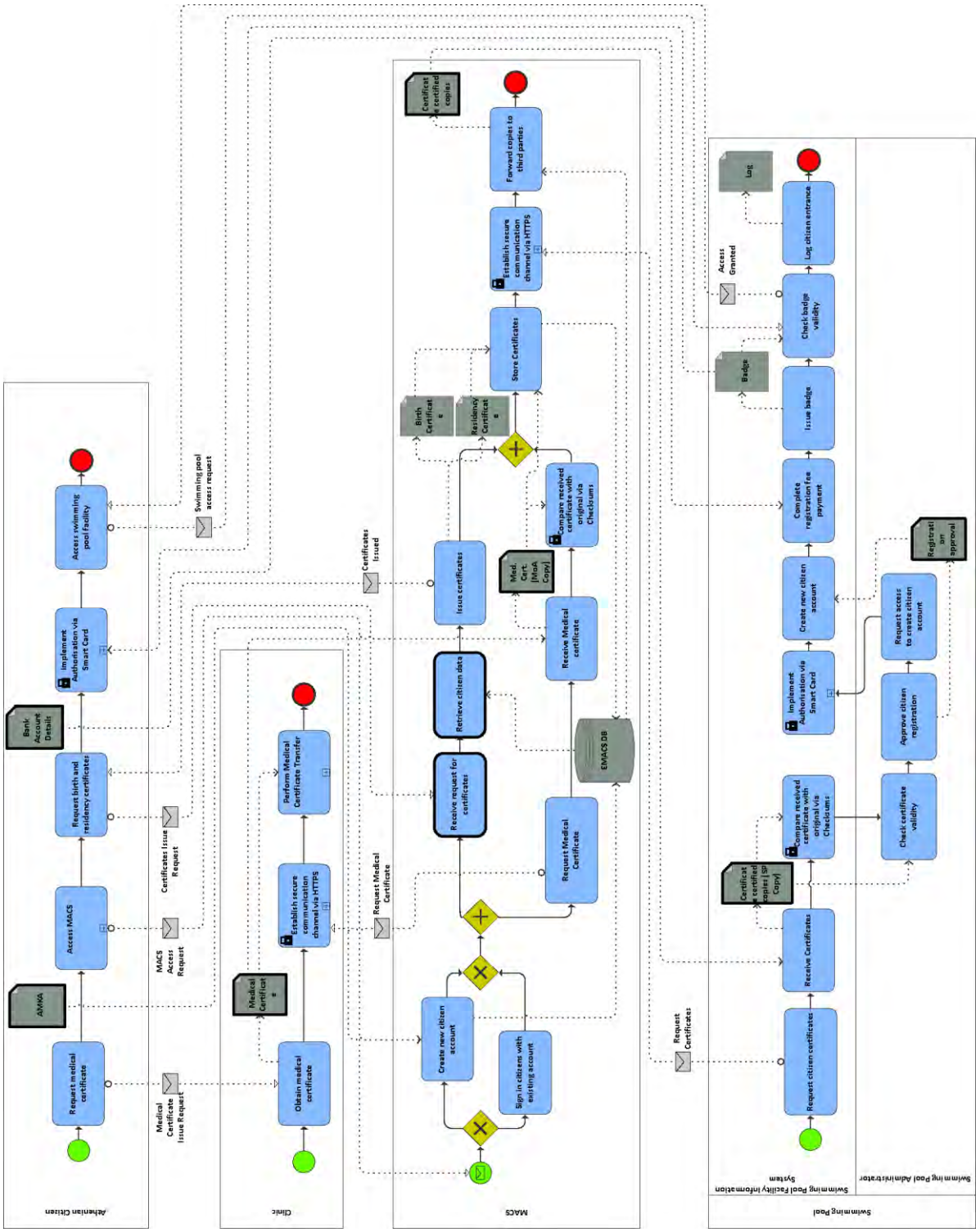

Figure 4.7: Business process model of the SPA system

Security Verification

The final step of the framework application used the latest iteration of the created business process model for the application of the security verification component. The business process model of Fig. 4.7, after being approved by the analysts of DAEM, was used as input for the verification process. The relevant concepts of the model (constraint activities, data objects and lanes) had their security-related attributes manually instantiated (e.g., source, target, owner), as described in Section 3.6. A similar instantiation process also took place for the security-implementing activities, which were previously introduced into the model via the process design patterns.

Next the verification algorithms were executed for each constraint activity and data object. The confidentiality verification algorithm (see Algorithm 3) revealed non successful implementation of confidentiality for the exchange of the AMKA data object between the citizen and the MACS system, since no secure channel had been established between the two lanes. The same issue was identified for the exchange of the citizen's Bank Account Details with the Swimming pool information system. The confidentiality of the exchange of the Medical Certificate between the Clinic and the MACS system could also not be verified due to the lack of the appropriate authorisation level of the MACS lane. The same issue was identified during the application of the authorisation algorithm (see Algorithm 2) for the "Receive request for certificates" and "Retrieve Citizen Data" activities of the MACS system lane. The source of both activities (Citizen lane) did not have the appropriate authorisation level for their execution. The security properties of the rest of the constraint elements of the business process model were successfully verified.

After the execution of the verification algorithms an improved version of the business process model was created. More specifically, a confidentiality-implementing process fragment was added at the citizen lane for its data exchanges with the MACS lane and another for the Swimming pool lane and an authorisation-implementing process fragment was added at the MACS lane.

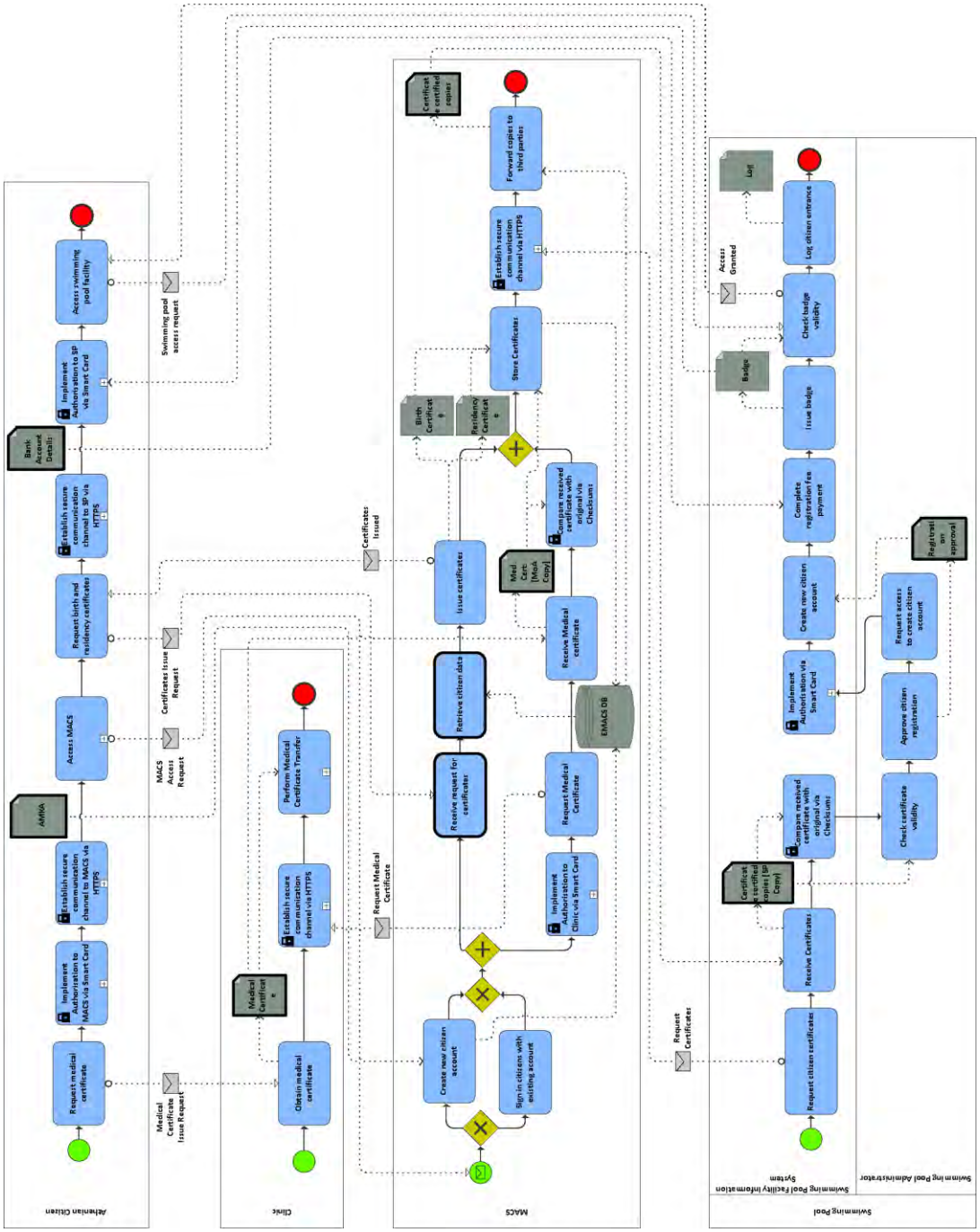

Figure 4.8: Post-verification Business process model of the SPA system

After such modifications, the algorithms that returned a false result in the previous step were, once again, executed for the refined version of the model, this time verifying the security properties of the model elements in question. Finally, the now security-verified business process model, as presented in Fig. 4.8, was delivered to the DAEM analysts and system stakeholders as the final output of

the framework's application.

4.3.4 Case Study Results

The insights gathered from the application of the framework to the SPA system will be discussed in this section. First, quantitative values will be calculated for the metrics introduced in Section 4.3.2, based on the intermediate and final outputs of the framework. Next, the exit interview of the involved stakeholders will be summarised to extract some empirical conclusions regarding their experience during the framework's application.

Metrics Evaluation

The metrics specified in Section 4.3.2 for assessing the conformance of the produced business process model of the SPA system to the initial requirements captured in the goal model, will be evaluated. The Security Requirements view of the Secure Tropos goal model of the system included five (5) actors, all of which were represented by the five (5) lanes in the final business process model. The goals of each actor, as captured at the goal model level, were all successfully operationalised by the activities included in the corresponding lanes of the business process model. For instance, the "Patient" actor included five (5) leaf-level nodes in the goal model, three (3) of which were delegated to other system actors for their achievement through dependency relationships. All five (5) goals were operationalised by corresponding activities at the business process level, either contained within the "Patient" lane or within the lanes corresponding to the dependee actors. Similarly, ten (10) unique resources were elicited at the system's goal model delegated between the different system actors. As a result, the produced process model contained twelve (12) data objects with some duplicate data objects resulting from the elicited resource delegations of the goal model.

Therefore, according to the Functional Conformance metric evaluation, the business process model was able maintain the totality of the information introduced at the goal model level. As a result of the application of the transformation steps introduced by the framework, a process model that conforms to the high level structure of the system, as captured by organisational goal models can be constructed. Thus, a goal model can provide a substantial source of information regarding the contents of a business process model.

Next, the metric related to the Security Conformance will be evaluated by comparing the security-related activities of the final business process model to

the security constraints elicited at the initial goal model. There were eight (8) different security constraints identified for all five (5) system actors in the goal model of the SPA system (see first column of Tab. 4.1). The produced busi- ness process model included six (6) security-implementing activities connected to elements within the five (5) lanes corresponding to the system actors. Nevertheless, some of the security-implementing activities operationalised more than one security constraints, therefore providing complete coverage of the identified security requirements at the business process level. Next, seven (7) different security-constraint elements were identified in the goal model (see third column of Tab. 4.1), some of which being placed within more than one actor containers and being constrained by more than one constraints. As a result of delegated resources leading to the creation of multiple copies of the same data object to different process lanes, nine (9) security-constraint elements were identified in the business process model, fully corresponding with their security-constraint counterpart at the goal model level. Finally, from the nine (9) security-constraint elements of the business process model, six (6) were able to be verified by the application of the Security Verification component at the first iteration of the business process model of Fig. 4.7.

Thus, according to the Security Conformance metric, the created business process model was able to fully operationalise the security related aspects that were captured at the goal model level and verify the majority of them. This highlights the ability of the developed framework to successfully support the capturing and transfer of security-related information across the different levels of abstraction. In terms of security verification, the first iteration of the produced business process model was able to be successfully verified for the majority of the identified security constraints. In conclusion, the above metrics highlight the ability of the framework to use the security analysis at the goal model level and successfully translate it to verifiable security implementations at the business process level.

Stakeholders Interview

A short interview was performed with the participating DAEM analysts to: (i) capture their experiences regarding the design of the SPA business process using the developed framework and (ii) identify what they perceived as its contributions and shortcomings. The Goal Question Metric (GQM) template [148] was utilised to structure each question of the interview as it allows us to specify: (i) the focus of the question, (ii) the objective of the question, (iii) the variable measured, (iv)

the subjects participating and (v) the context of the question.

Analyse the developed framework for the purpose of quantitative evaluation with respect to the perceived complexity and understandability of the utilised modelling languages from the point of view of the system designers and security expert in the context of creating and understanding Secure Tropos and BPMN 2.0 models

Table 4.4: Goal-question-metric template for question 1 of stakeholder interview

The first point of discussion was focused on the complexity and understandability of the modelling languages used by the framework, as indicated by the GQM template of Tab. 4.4. The participants noted that their familiarity with both Secure Tropos and BPMN helped them to create and comprehend the modelling outputs of the different steps of the framework. Despite the large size and information density of the created models, the modelling languages used were clear and easily comprehensible and, since no major extensions were made to any of them, the analysts could use them without the need of further instructions.

Analyse the developed framework for the purpose of quantitative evaluation with respect to the perceived complexity and applicability of the Decision Support component from the point of view of the system designers and security expert in the context of selecting the security mechanisms to be implemented using the component

Table 4.5: Goal-question-metric template for question 2 of stakeholder interview

As indicated by Tab. 4.5, the second interview question focused on the experiences of the participants using the decision support componenent. Regarding the application of the decision support component for the selection of the security mechanisms to be implemented, the participating analyst commended its flexibility but noticed that due to its complexity they required some guidance for its comprehension and application. More specifically, the ability of the component to allow the definition and prioritisation of variables, which can capture a wide range of functional and non-functional system characteristics, adds to the

adaptability of the mechanism selection process. The analysts also identified the ability to generate different prioritisation scenarios as a "very positive" feature of the component, as it provided them with flexibility during decision making. Nevertheless, the number of quantitative values that needed to be instantiated and the specialised tool support required for the application of the decision support process added to its complexity and required some guidance for its successful application. The security expert, whose input was critical for the application of that component, also indicated that some further guidelines or resources for the identification of numerical values for variables related to information secu- rity risks (e.g., likelihood, impact) would greatly improve the effectiveness of the component. Nevertheless, he recognised that the subjectivity involved in the identification of quantitative values for such aspects is an inherent limitation of all risk management frameworks and that the structured and organised approach provided by the developed component is a step towards the right direction.

Analyse the developed framework for the purpose of quantitative evaluation with respect to the perceived usefulness of the Model Transformation component from the point of view of the system designers and security expert in the context of understanding and utilising the hybrid reference process model

Table 4.6: Goal-question-metric template for question 3 of stakeholder interview

Next, regarding the output of the Model Transformation component, as contextualised by the GQM template of Tab. 4.6, the analysts indicated that the hybrid reference process model proved to be a valuable artefact since it provided a solid baseline around which the final business process model can be constructed. The transformation of the goal model to the hybrid reference process model was intuitive and, since it was automatically performed by the same modelling tool that was used to construct the goal model, was also effortless.

Analyse the developed framework for the purpose of quantitative evaluation with respect to the perceived usability and complexity of the Business Process Modelling component from the point of view of the system designers and security expert in the context of refining the hybrid reference process model to a complete business process model

Table 4.7: Goal-question-metric template for question 4 of stakeholder interview

As per the GQM template presented in Tab. 4.7, the process patterns were also useful to the analysts since they provided a structured and predefined way to implement the different types of security constraints. They were also at an appropriate level of abstraction which matched the abstraction level of the final business process model. Some concerns regarding the patterns were focused on their placement within the process model, which was not always obvious, and the additional complexity they introduced to the final process model, which led to the analysts preferring to introduce them as collapsed sub-processes to keep the model manageable. Finally, when asked about the refinement required for the creation of the final business process model, the analysts indicated that it was not considered as a major endeavour since the hybrid reference process model combined with the security patterns had already solidified the larger part of the final process structure.

Analyse the developed framework for the purpose of quantitative evaluation with respect to the perceived usefulness of the Security Verification component from the point of view of the system designers and security expert in the context of understanding and utilising the output of the security verification process

Table 4.8: Goal-question-metric template for question 5 of stakeholder interview

Since the application of the verification component did not involve the analysts of DAEM, their comments were mainly focused on the outcome of the verification process as indicated by the GQM template in Tab. 4.8. They indicated that it was "very important" that the verification process was able to identify, not only

the existence of violations of security properties, but also their exact location within the process, as well as what is required for them to be fixed. They also noted that the integration of the security verification component to a business process modelling tool would be of great benefit in terms of ease-of-use and real-life applicability.

The interview with the involved analysts concluded with some final remarks regarding their overall experience with the usage of the developed framework. They indicated that the produced business process model will be "a very useful" artefact for the further development of the SPA system, since it was the output of a structured and, at large parts, quantitative process which will allow them to provide justification regarding design choices to the system's stakeholders. They also noted that the connection between high level goals and operational level processes is an important contribution resulting from the application of the framework, as it promotes alignment between strategy and operations. They indicated that the overall application of the process can be, at times, demanding in terms of time and complexity, but the available tool support can help reduce that overhead.

4.3.5 Threats to Validity

The case study performed to evaluate the framework proposed in this work involved two participants from the organisation in charge of developing the studied system. The participants were selected due to their relevant background (i.e., information security and system modelling) and their knowledge of the studied system. Nevertheless, the generalisability of the outcomes of the specific case study can be considered limited due to the involvement of a small number of stakeholders using the proposed framework and its application to a single real-life information system. The limited generalisability issue was partially mitigated by the previous smaller scale applications of the framework, as described in Section 4.1, the findings of which were in accordance with the outcomes of the large scale case study presented in this section. Furthermore, the detailed design and protocol of the case study, as presented in the beginning of this section, can facilitate its replication in other large scale information systems in future work to further solidify our findings.

The involvement of the author throughout the application of the proposed framework during the presented case study can also introduce bias to the process. In order to reduce such effect, the participation of the author was limited to providing an overview of each framework component prior to its application by

the case study participants and address any of their inquiries during the process. After the completion of each step the participants and the author communicated to discuss their experience and identify potential aspects of the deliverables in need of further refinement. The only exception to the above process was the application of the Security Verification component, which was the final component of the framework to be developed and tested. The Security Verification component, which was developed in the later stages of this research project, is not currently supported by a software tool and was, therefore, manually applied to the business process model produced by the case study by the author. Nonetheless, the results of the component's application were presented to the case study participants and their implications towards the final deliverable were thoroughly discussed with them.

Finally, even though some quantitative metrics were identified for the evaluation of the results of the case study, the majority of the insights originated from the interviewing the case study participants and, therefore, were qualitative in nature. While the quantitative metrics were able to capture the conceptual and security-related completeness of the produced artefacts, they were not able to provide any further indication of their quality as there was no previous baseline to compare them against. Thus, the opinions and experiences of the involved system stakeholders, while potentially subjective, were the main source for the evaluation of the proposed framework's application to the studied system. To mitigate such issues in future work, researchers could identify legacy information systems which can be redesigned using the proposed framework and compare their new design with the previous baseline. Alternatively, if a similar approach for the design of secure business processes is identified in future literature, it can be applied to the same system selected for our case study and have the results of both applications compared in a quantitative way.

4.4 Lessons Learned

The different evaluation activities, presented in this chapter, facilitated the refinement of the developed framework to its current state. The proof of concept applications of parts of the framework, performed in the earlier stages of this research project, provided valuable insights which led to the improvement of each component in an iterative manner. Next, the case study, which constituted the last step of the framework's evaluation process, facilitated the creation of the final version of the different framework components. This was due to the nature

of the selected system, as it allowed us to observe the application of the different framework components in a relatively large-scale and complex real life scenario and thus, identify potential shortcomings.

In further detail, several versions of the transformation steps, which is the central artefact of the Model Transformation component, have been produced throughout the lifecycle of the current research project, as presented in [8], [9], [37], [38]. The final version, as described in Section 3.4.1, includes the transformation of only leaf-level goals and plans to process activities, as opposed to earlier versions which transformed all goals and plans to process activities. The decision to only transform leaf-level nodes was reached in order to reduce the complexity of the process model by minimising the number of nested activities (i.e., tasks and/or sub-processes nested within higher level sub-processes). This version of the transformation rules was implemented by the extended SecTro tool (see Section 3.7) in order to automate the model transformation process. Therefore, it facilitated the creation of manageable process models, especially when dealing with large scale systems, as was the case for the SPA system of the case study.

Regarding the Decision Support component, when first conceptualised, the evaluation of impact and likelihood values for the identified threats was performed in an ad-hoc manner. That process entailed the instantiation of values for the variables involved in the risk calculation from a continuous zero (0) to one (1) scale and was left at the complete discretion of a security expert. During the refinement of that component, AHP was selected for the assignment of impact and likelihood values, as it allows the ranking of the identified vulnerabilities relative to each other, therefore reducing the subjectivity and arbitrariness of the value assignment process. Thus, AHP provides a more applicable and intuitive structure to support decision making and, as a result, is a popular choice among practitioners [149]. That decision shaped the final version of the decision support component, as introduced in [43] and Section 3.3. The same version of the component was also used during the case study and provided useful support to guide the selection of the final security composition of the SPA system by the involved stakeholders.

A similar refinement process was followed for the security process patterns, which are used for the integration and instantiation of security countermeasures during the application of the Business Process Modelling component. The earliest version of such patterns, as introduced in [40], could only be applied to process lanes existing within the same pool. The latest version of the patterns, as pre-

sented in [44] and Section 3.5.1, were extended to include message exchanges across process lanes, allowing them to be applicable in a broader range of scenarios, where the participating lanes do not belong in the same pool. As a result, the latest version of the process patterns could be easily integrated within the business process model of the SPA system used in the final case study.

The Security Verification component, introduced in [41], was initially only able to verify the security properties of process models with lanes contained within the same process pool where the process flow was continuous across different lanes (i.e., one start and one end point). When applied to the SPA system during the case study, it was initially not able to handle the independent control flows of each lane and the message exchanges used for cross-lane communications. As a result, the attributes used to capture the structure of the control flow had to be adjusted and the verification algorithm had to be modified, in order to support the verification of the security properties of larger and more complex process models. Such refinement, initiated as a result of the large-scale case study, created the final version of the Security Verification component, as presented in Section 3.6.

Chapter 5

Conclusion

In this work we presented a framework for the design of secure business process models originating from high-level organisational goal models. The proposed framework is comprised of different components with varying functionalities which, when applied in sequence, are able to produce a complete business process model, compliant with high-level security requirements. The developed framework demonstrated potential when applied to a e-governance system under development as it provided a structured sequence of steps, which led to the development of a secure business process model that described one of the processes which will be executed by the studied system, upon its implementation.

As the first step to the framework's application, the Goal Modelling component is used to capture the organisational level of abstraction of the system to-be, using the Secure Tropos goal-oriented requirements engineering framework. After the initial security requirements, threats and security mechanisms have been elicited and captured on the organisational goal model, the Decision Support component is utilised for the selection of the most fitting security mechanism combinations, which will be operationalised in the final business process model. The application of that component allows the system's stakeholders to define the evaluation criteria they consider most important and, based on their input, it automatically evaluates all of the alternative security implementing configurations to identify the optimal solution. Next, the Model Transformation component of the framework is utilised for transitioning to the operational level of abstraction. The centrepiece of the model transformation component is the hybrid reference process model, which is created using a series of transformation rules, in order to transfer the information included in the initial goal model to the business process level of abstraction. As a result, the hybrid reference process model uses concepts from both the Secure Tropos modelling language and BPMN 2.0, which is the

most established business process modelling standard, in order to create a midway process reference model that captures both functional and security related aspects of the system to-be. The hybrid reference model along with the optimal security mechanism configurations are used as input to the Business Process Modelling component, which uses a set of security process design patterns to integrate security-implementing activities in the business process skeleton created by the transformation of the goal model. After some manual refinement a BPMN 2.0, secure business process model is created from the the hybrid reference process model, enhanced by the instantiated process patterns. Finally, the Security Verification component utilises the created business process model as input in order to verify its adherence to the elicited security requirements, using a set of security verification algorithms.

As a result of the application of the proposed framework, the stakeholder elaboration of the system to-be, which is performed on a highly abstract level and is mainly influenced by organisational aspects and strategic objectives, is transformed into an operational level business process model, able to capture the sequence of activities required for achievement of such organisational objectives. The transition between system models of different levels of abstraction allows the shift from a high- to a low-level view of the system without information loss, due to the explicit mappings between the concepts belonging to different abstraction levels. Another important aspect is the ability of the framework's artefacts to capture a wide range of alternative system configurations in terms of security and support the selection of the one best-fitting to the system stakeholders' needs. Thus, an alternate business process configuration can be produced without the need to apply the whole design process from scratch, when contextual changes in the system's environment occur. Moreover, the introduction of a security process pattern library provides further structure to the process of security integration at the operational level of abstraction, reducing the overhead required in terms of security-related expertise and effort by the process designers. Finally, the security properties of the produced business process design can be explicitly verified through the application of the developed verification algorithms, thus providing further assurance of the alignment of the final framework's output to the organisational level security requirements. Therefore, the proposed framework provides a flexible and structured approach towards the design and verification of secure business process models which are aligned with high-level organisational strategy and comply with the functional and non-functional constraints of the system's environment.

The capabilities of the presented framework make it a beneficial instrument for system and business process designers in need of producing secure business process models. Its prerequisites in terms of knowledge are limited to the basics of goal-oriented requirements engineering, business process modelling and high-level information security concepts. The involvement of information security experts can further refine the output of the framework's application as some of their input is important for the elicitation of security constraints, threats and countermeasures. As already discussed, the framework is geared towards supporting the design phases of the business process management lifecycle. Therefore, its contribution concludes upon the production and verification of a secure business process model. Nevertheless, a business process model produced as the output of the framework's' application can be used as a blueprint for the later stages of the business process management lifecycle by other specialised approaches for service identification and orchestration and process execution and monitoring frameworks. Furthermore, since sociotechnical systems are the starting point of the analysis supported by the framework, it is better equipped to deal with the design of systems operating in a multi-agent environment rather than describing highly detailed and technical processes of individual system components.

Apart from the contributions of the developed framework, which are discussed in detail below (see Section 5.2), there are assumptions and limitations worth of critical discussion. The design science research approach followed for the development of the framework is a popular choice when developing artefacts in the subject area of information systems. Its wide-spread adoption, in combination with the well-defined research steps it provides, led to its selection as the research method of choice for this project. Nonetheless, that choice was not a result of exhaustive comparison between design science and other research approaches but rather resulted from the fact that the research steps already undertaken in the early stages of the project matched with the guidelines of this specific research approach (i.e., gap identification through literature, develop and evaluate feedback loop of initial framework components). Thus, it may be beneficial for future similar research attempts to thoroughly examine available research methodologies before initiating the development of artefacts. Another research assumption made during the lifetime of this project was to limit the scope of the study to model-driven information security for business processes. This decision directly affected the scope of the literature review (see Chapter 2) as it led us to the exclusion of works which either dealt with security-adjacent concepts (e.g., privacy, trust, access control) or dealt with security in a formal and non diagrammatic

manner (e.g., formal languages, rule-driven). This choice inevitably narrowed the body of literature that was studied to extract research gaps but also allowed us to focus the scope of the project and thoroughly analyse the works which fitted within that scope. The trade-off between the width of a project's scope and the depth of the analysis provided is, therefore, an important aspect to consider during the early stages of a research project.

Another aspect of this research project worth of further discussion is the evaluation of the developed framework (see Chapter 4. As already discussed, both individual components and the complete framework were evaluated and iteratively refined as a result of their application in real life information systems. The large scale evaluation of the complete framework was performed via the case study described in Section 4.3. The application of the framework in collaboration with real life practitioners, performed during this case study, yielded useful insights but with limited generalisability. This was mainly due to the fact the developed framework was applied as a whole only to a singular real life information system, which at the time was still under development. Therefore, there was no benchmark against which the produced business process model could be compared to, in order to gather quantitative data. Instead, the conclusions reached after the completion of the case study were based on the semi-structured interviews with the participating stakeholders and some ad-hoc metrics designed specifically for the context of the system at hand. Therefore, there are still aspects of the developed framework which could benefit from further evaluation in different real life contexts with varying size and complexity. For instance, in regards to the scalability of the framework, the different components were able to be utilised as intended both during small-scale individual applications (see Section 4.1) and during the large scale application of the complete framework at the case study (see Section 4.3). Nonetheless, further applications of the framework to other information systems of similar or greater size and complexity could strengthen the generalisability of such conclusions.

5.1 Research Outputs

The contributions of the different framework components can be matched to the objectives and research questions this research project aims to tackle (see Sections 1.3 and 1.2). More specifically, in regards to the first research question, the combination of the Goal Modelling, Model Transformation and Business Process Modelling components facilitate the creation of a business process model aligned

with the requirements and constraints captured at the goal model level. The hybrid reference process model, which is the main artefact produced by the model transformation component, can be considered a skeleton of a business process model which maps the actors, resources and goals of the goal model to the corresponding business process level concepts (i.e., lanes, data objects, activities). Therefore, through the concept mappings and transformation rules introduced by the Model Transformation component, structural information, captured by goal models at the organisational level of abstraction, dictates the structure of the resulting business process design. Furthermore, the integration of security-related elements, elicited from the goal model, into the produced business process model is also achieved by the combination of the application of the three aforementioned components. The Goal Modelling component facilitates the elicitation of security constraints and potential implementation mechanisms, the Model Transformation component maps such elements on the appropriate parts of the business process skeleton and the Business Process Modelling component integrates them into the final process model via the use of the process design patterns, developed as part of this work. Therefore, security-related information captured by goal models at the organisational level of abstraction is also transferred to the operational level of abstraction to shape the final secure business process design. This combination of components also leads to the achievement of the first two objectives of this research project (i.e., "Obj.I: Create an approach that uses high-level, functional and non-functional organisational goals as input for the design of business processes." and "Obj.2: Develop a structured way for producing business process designs able to operationalise the identified organisational goals."). Furthermore, via the security process patterns introduced by the Business Process Modelling components helps achieve the fourth objective of this research project (i.e., "Obj.IV: Provide a structured way for integrating predefined security configurations into business process models.").

Regarding the second research question, the Decision Support component of the proposed framework facilitates the decision making process regarding design choices at the business process level. The aspects that need to be taken into consideration during the selection between the alternatives in terms of security mechanisms, are defined by the system stakeholders and expressed as optimisation variables during the initial steps of the decision support process. Such variables reflect both security and risk-related coverage provided by each candidate security mechanism, while also being able to capture their contribution towards the achievement of non-functional system goals. Moreover, the ability of the compo-

nent to allow the prioritisation of each variable's prioritisation and definition of soft and hard-caps for their values, allows the definitions of optimisation scenarios able to accurately reflect the needs of the system's stakeholders. Finally, the identification of optimal solutions for each scenario, through the automated application of satisfiability solvers provides further structure for the decision-making process regarding the security configuration of the business process designs under development. The introduction of such component into the proposed framework also helps achieve the third research objective of this project (i.e., "Obj. III: Provide a new approach to support the selection of appropriate security configurations to be implemented at the business process level, according to situational needs and constraints.").

Finally, regarding the third research question, the compliance of the final business process design to the initial security constraints is verified by the application of the Security Verification component. This component provides model checking capabilities in order to ensure that the business process model produced as the result of the framework's application has specific properties which will make it compliant with the security requirements elicited from the initial goal model. To achieve that a series of attributes have been defined to capture properties of process elements related to their security needs and their position within the control flow of the process. Additionally, verification algorithms have been developed for each of the main types of security requirements, which check the values of certain instantiated attributes of security-constraint process elements and identify potential security violation. Therefore, the application of that component can pinpoint the location of security violations within the process model for each security requirement elicited by the organisational level goal model. Furthermore, such component contributes towards the achievement of the final objective of the research project (i.e., "Obj. V: Develop an approach that enables the verification of the compliance of the security properties of a business process model to the security constraints identified at the organisational level.").

5.2 Main Contributions

The framework presented in this work contributes towards a multitude of different areas of interest, including security requirements engineering, risk management, business process modelling, organisational and operational level alignment and decision support. More specifically, the major contributions of the proposed framework can be summarised as follows:

- The extension of the already established Secure Tropos modelling language, allowing it also cover risk related concerns via the introduction of new concepts (e.g., risk). In addition to that, new attributes have been introduced to existing concepts, such as mechanisms and soft goals, to allow for a more accurate and quantifiable description of the relationships with each other (e.g., contribution of mechanism towards soft goal, degree of mechanism's threat mitigation).

- The introduction of concept mappings between Secure Tropos and BPMN 2.0 which allow entities from the organisational perspective to be transformed to their process-level counterparts based on their conceptual similarities. Such concept mappings play a major role in the construction of an intermediate business process model skeleton, known as hybrid reference process model within our framework, which essentially acts as a bridge connecting the organisational with the operational level of abstraction.

- Transformation rules built on top of the concept mappings in order to guide the construction of the hybrid reference process model, using the organisational goal model as input. The explicitness of these rules offers the potential to automate the model transformation process by computer-aided software engineering (CASE) tools in order to minimise the manual intervention required to create and transition between the different model types supported by this framework. Such automated functionality has been introduced into an existing software tool which supports the construction of Secure Tropos goal models and their automatic transformation to hybrid reference process models.

- The introduction of the hybrid reference process model, created by the application of the transformation rules as an intermediate artefact, aiming to transfer the information captured in an organisational security oriented goal model and express it in business process terms. Through the use of the hybrid reference process model, high level goals of the organisation can guide the design of its business processes, creating an alignment between organisational strategy and operations.

- The development of a Decision Support component which receives stakeholder and expert input concerning a number of functional and non-functional aspects of the system and uses it to identify optimal combinations of security implementing technologies. The decision support process component

provides flexibility by allowing stakeholders to select and prioritise the aspects they want to be taken into account during the decision making process (e.g., soft goals) and experts to evaluate the coverage that different implementation technologies provide towards such aspects (e.g., contribution of a mechanism towards a soft goal).

- The introduction of a process design pattern library which includes generic, predefined process fragments which are able to express the operationalisation of different types of security requirements. By creating a process design pattern to each type of security requirement (i.e., authentication, authorisation, confidentiality, integrity, availability) in a technology-agnostic manner, a useful collection of reusable business process fragments is established, which can be easily integrated to new or existing CASE tools to minimise the manual effort required for the creation of secure business process designs.

- The ability to extract a number of similar but slightly different business process models from the same hybrid reference model. The variation in the final process model originates from the alternatives in terms of security implementing technologies, which the stakeholders can select from the hybrid reference process model, assisted by the decision support framework. The hybrid reference process model has the ability to maintain information regarding all the different alternatives and therefore, can be re-used to produce a new business process design if the stakeholders' decision criteria or the context in which the system operates is altered. Thus, the framework offers a flexible and adaptable approach via the re-use of the hybrid reference model artefact.

- The security verification capabilities provided by the introduction of process element attributes and security verification algorithms. Such security verification capabilities provide a structured way of verifying the compliance of the produced business process model to the security requirements initially elicited by Secure Tropos goal models and the identification of the type and location of security violations within the control flow of the process model.

5.3 Future Research Directions

The development and evaluation of a framework for the creation of secure business process designs, undertaken through this research project, also revealed directions

for future research attempts. More specifically, even though this work focused on security, the extension of the developed framework to support aspects related to other security-adjacent concepts, such as privacy and trust, can be explored in future work. Privacy and trust are often treated as another type of security requirement during the design of information system that support the execution of business processes. Nevertheless, research in the area of privacy and trust requirements engineering reveals that there are multiple, discrete aspects worth of analysis in such areas of research. Some preliminary efforts to incorporate privacy concerns into the developed framework have already been undertaken during this research project, as a set of privacy process patterns have been developed in [36] and [39]. Nevertheless, potential conflicts between privacy and security require further consideration, which even though is outside the scope of this research project, is worth considering in future research efforts in the area.

Another direction for future work is the connection of the output of this work to service level compositions that can support the execution of the produced business process designs. Since the scope of this work was focused at the design level, such aspects have not been considered during this research project. Nevertheless, the output of the application of our framework can assist in the identification of implementation level artefacts to support the process execution, since the produced process designs can capture a detailed description of their functional and security related aspects. Steps towards that direction have already been undertaken in [37] and [38] were the developed framework has provided input for implementation-level efforts in the development of software product lines and secure cloud service compositions.

Simultaneously to the development of the framework, a computer-aided software engineering (CASE) tool was also extended to support and automate the creation and transition between the different models of the proposed framework. The CASE tool is able to provide users with a graphical environment in which they can create goal models using the Secure Tropos modelling language and automatically transform them into hybrid reference process models. Other existing tools were also identified to support other framework functionalities, as discussed in Section 3.7. The CGM tool is used for the application of the Decision Support component, while a variety of business process modelling tools can be used for the application of the Business Process Modelling component. Nevertheless, since the development of deployable software tools was outside the scope of the current research project, future work can extend the coverage of software tools for the developed framework. More specifically, the transition between the different tools

can be automated by forwarding the output of one tool to be used as input for the next. Finally, the security verification attributes and algorithms, introduced by the Security Verification component of the framework, can be implemented to a new or existing business process modelling tool which could allow users to instantiate the relevant attributes of different components of a process model and automatically execute the verification algorithms to identify potential security violations.

Bibliography

[1] M. Weske, *Business process management: concepts, languages, architectures*. Springer Publishing Company, Incorporated, 2010.

[2] T. Neubauer, M. Klemen, and S. Biffl, "Secure business process management: A roadmap," in *Availability, Reliability and Security (ARES), 2006 First International Conference on*, IEEE, 2006, pp. 457–464.

[3] M. Leitner, M. Miller, and S. Rinderle-Ma, "An analysis and evaluation of security aspects in the business process model and notation," in *Availability, Reliability and Security (ARES), 2013 Eighth International Conference on*, IEEE, 2013, pp. 262–267.

[4] K. Decreus and G. Poels, "A goal-oriented requirements engineering method for business processes.," in *CAiSE Forum 2010, LNBP*, Springer, vol. 72, 2010, pp. 29–43.

[5] J. Horkoff, T. Li, F.-L. Li, M. Salnitri, E. Cardoso, P. Giorgini, J. Mylopoulos, and J. Pimentel, "Taking goal models downstream: A systematic roadmap," in *Research Challenges in Information Science (RCIS), 2014 IEEE Eighth International Conference on*, IEEE, 2014, pp. 1–12.

[6] H. Mouratidis and P. Giorgini, "Secure Tropos: A security-oriented extension of the tropos methodology," *International Journal of Software Engineering and Knowledge Engineering*, vol. 17, no. 02, pp. 285–309, 2007.

[7] Object Management Group, "Business Process Model and Notation (BPMN) 2.0," Tech. Rep., 2011.

[8] N. Argyropoulos, H. Mouratidis, and A. Fish, "Towards the derivation of secure business process designs," in *International Conference on Conceptual Modeling*, Springer, 2015, pp. 248–258.

[9] N. Argyropoulos, L. M. Alcañiz, H. Mouratidis, A. Fish, D. G. Rosado, I. G.-R. de Guzmán, and E. Fernández-Medina, "Eliciting security requirements for business processes of legacy systems," in *IFIP Working Conference on The Practice of Enterprise Modeling*, Springer, 2015, pp. 91–107.

[10] A. Yousfi, R. Saidi, and A. K. Dey, "Variability patterns for business processes in bpmn," *Information Systems and e-Business Management*, vol. 14, no. 3, pp. 443–467, 2016.

[11] W. Van Der Aalst, "Business process management: A comprehensive survey," *ISRN Software Engineering*, vol. 2013, 2013.

[12] M. Zur Muehlen and D. T.-Y. Ho, "Risk management in the bpm lifecycle," in *International Conference on Business Process Management*, Springer, 2005, pp. 454–466.

[13] M. Dumas, M. La Rosa, J. Mendling, H. A. Reijers, *et al.*, *Fundamentals of business process management*. Springer, 2013.

[14] A. H. Ter Hofstede and M. Weske, "Business process management: A survey," in *Proceedings of the 1st International Conference on Business Process Management, volume 2678 of LNCS*, Citeseer, 2003.

[15] R. K. Ko, S. S. Lee, and E. Wah Lee, "Business process management (bpm) standards: A survey," *Business Process Management Journal*, vol. 15, no. 5, pp. 744–791, 2009.

[16] C. Haley, R. Laney, J. Moffett, and B. Nuseibeh, "Security requirements engineering: A framework for representation and analysis," *IEEE Transactions on Software Engineering*, vol. 34, no. 1, pp. 133–153, 2008.

[17] H. Mouratidis, "Secure software systems engineering: The secure tropos approach," *JSW*, vol. 6, no. 3, pp. 331–339, 2011.

[18] E. Paja, F. Dalpiaz, and P. Giorgini, "Managing security requirements conflicts in socio-technical systems," in *International Conference on Conceptual Modeling*, Springer, 2013, pp. 270–283.

[19] E. Yu, P. Giorgini, N. Maiden, and J. Mylopoulos, "Social modeling for requirements engineering: An introduction," *Social Modeling for Requirements Engineering*, pp. 3–10, 2011.

[20] E. Yu and J. Mylopoulos, "Why goal-oriented requirements engineering," in *Requirements Engineering: Foundations of Software Quality, Proceedings of the 4th International Workshop on*, vol. 15, 1998, pp. 15–22.

[21] J. Horkoff, F. B. Aydemir, E. Cardoso, T. Li, A. Maté, E. Paja, M. Salnitri, L. Piras, J. Mylopoulos, and P. Giorgini, "Goal-oriented requirements engineering: An extended systematic mapping study," *Requirements Engineering*, pp. 1–28, 2017.

[22] E. S. Yu, "Towards modelling and reasoning support for early-phase requirements engineering," in *Requirements Engineering, 1997., Proceedings of the Third IEEE International Symposium on*, IEEE, 1997, pp. 226–235.

[23] F. Massacci, J. Mylopoulos, and N. Zannone, "Security requirements engineering: The si* modeling language and the secure tropos methodology," *Advances in Intelligent Information Systems*, pp. 147–174, 2010.

[24] G. Elahi and E. Yu, "A goal oriented approach for modeling and analyzing security trade-offs," *International Conference on Conceptual Modeling*, pp. 375–390, 2007.

[25] P. Bresciani, A. Perini, P. Giorgini, F. Giunchiglia, and J. Mylopoulos, "Tropos: An agent-oriented software development methodology," *Autonomous Agents and Multi-Agent Systems*, vol. 8, no. 3, pp. 203–236, 2004.

[26] F. Dalpiaz, E. Paja, and P. Giorgini, "Security requirements engineer- ing via commitments," in *Socio-Technical Aspects in Security and Trust (STAST), 2011 1st Workshop on*, IEEE, 2011, pp. 1–8.

[27] D. Mellado, C. Blanco, L. E. Sánchez, and E. Fernández-Medina, "A systematic review of security requirements engineering," *Computer Standards & Interfaces*, vol. 32, no. 4, pp. 153–165, 2010.

[28] B. Fabian, S. Gürses, M. Heisel, T. Santen, and H. Schmidt, "A comparison of security requirements engineering methods," *Requirements engineering*, vol. 15, no. 1, pp. 7–40, 2010.

[29] A. R. Hevner, "A three cycle view of design science research," *Scandinavian journal of information systems*, vol. 19, no. 2, p. 4, 2007.

[30] S. T. March and V. C. Storey, "Design science in the information systems discipline: An introduction to the special issue on design science research," *MIS quarterly*, pp. 725–730, 2008.

[31] A. R. Hevner, S. T. March, J. Park, and S. Ram, "Design science in information systems research," *MIS quarterly*, vol. 28, no. 1, pp. 75–105, 2004.

[32] S. T. March and G. F. Smith, "Design and natural science research on information technology," *Decision support systems*, vol. 15, no. 4, pp. 251–266, 1995.

[33] M. D. Myers *et al.*, "Qualitative research in information systems," *Management Information Systems Quarterly*, vol. 21, no. 2, pp. 241–242, 1997.

[34] P. Runeson, M. Host, A. Rainer, and B. Regnell, *Case study research in software engineering: Guidelines and examples*. John Wiley & Sons, 2012.

[35] H. Mouratidis, N. Argyropoulos, and S. Shei, "Security requirements engineering for cloud computing: The secure tropos approach," in *Domain-Specific Conceptual Modeling*, Springer, 2016, pp. 357–380.

[36] N. Argyropoulos, C. Kalloniatis, H. Mouratidis, and A. Fish, "Incorporating privacy patterns into semi-automatic business process derivation," in *Research Challenges in Information Science (RCIS), 2016 IEEE Tenth International Conference on*, IEEE, 2016, pp. 1–12.

[37] D. Sprovieri, N. Argyropoulos, C. Souveyet, R. Mazo, H. Mouratidis, and A. Fish, "Security alignment analysis of software product lines," in *Enterprise Systems (ES), 2016 4th International Conference on*, IEEE, 2016, pp. 97–103.

[38] N. Argyropoulos, S. Shei, C. Kalloniatis, H. Mouratidis, A. Delaney, A. Fish, and S. Gritzalis, "A semi-automatic approach for eliciting cloud security and privacy requirements," in *Proceedings of the 50th Hawaii International Conference on System Sciences*, 2017.

[39] V. Diamantopoulou, N. Argyropoulos, C. Kalloniatis, and S. Gritzalis, "Supporting the design of privacy-aware business processes via privacy process patterns," in *Research Challenges in Information Science (RCIS), 2017 11th International Conference on*, IEEE, 2017, pp. 187–198.

[40] N. Argyropoulos, H. Mouratidis, and A. Fish, "Supporting secure business process design via security process patterns," in *Enterprise, Business-Process and Information Systems Modeling*, Springer, 2017, pp. 19–33.

[41] ——, "Attribute-based security verification of business process models," in *Business Informatics (CBI), 2017 IEEE 19th Conference on*, IEEE, vol. 1, 2017, pp. 43–52.

[42] M. Pavlidis, H. Mouratidis, E. Panaousis, and N. Argyropoulos, "Selecting security mechanisms in secure tropos," in *International Conference on Trust and Privacy in Digital Business*, Springer, 2017, pp. 99–114.

[43] N. Argyropoulos, K. Angelopoulos, H. Mouratidis, and A. Fish, "Decision-making in security requirements engineering with constrained goal models," in *SECurity and Privacy Requirements Engineering, 2017 1st International Workshop on (SECPRE 2017)*, IEEE, 2017.

[44] N. Argyropoulos, H. Mouratidis, and A. Fish, "Enhancing secure business process design with security process patterns," *Software and Systems Modeling*, 2018.

[45] N. Argyropoulos, K. Angelopoulos, H. Mouratidis, and A. Fish, "Risk-aware decision support with constrained goal models," *Information and Computer Security*, 2018.

[46] P. Brereton, B. A. Kitchenham, D. Budgen, M. Turner, and M. Khalil, "Lessons from applying the systematic literature review process within the software engineering domain," *Journal of systems and software*, vol. 80, no. 4, pp. 571–583, 2007.

[47] Object Management Group, "MDA Guide, Version 1.0.1," Tech. Rep., 2003.

[48] A. D. Brucker and I. Hang, "Secure and compliant implementation of business process-driven systems," in *Joint Workshop on Security in Business Processes (SBP)*, Springer-Verlag, vol. 132, 2012, pp. 662–674.

[49] M. Leitner and S. Rinderle-Ma, "A systematic review on security in process-aware information systems–constitution, challenges, and future directions," *Information and Software Technology*, vol. 56, no. 3, pp. 273–293, 2014.

[50] A. Lapouchnian, Y. Yu, and J. Mylopoulos, "Requirements-driven design and configuration management of business processes," in *International Conference on Business Process Management*, Springer, 2007, pp. 246–261.

[51] M. Séguran, C. Hébert, and G. Frankova, "Secure workflow development from early requirements analysis," in *on Web Services ECOWS'08, IEEE Sixth European Conference*, IEEE, 2008, pp. 125–134.

[52] C. Kalloniatis, E. Kavakli, and S. Gritzalis, "Addressing privacy requirements in system design: The pris method," *Requirements Engineering*, vol. 13, no. 3, pp. 241–255, 2008.

[53] ——, "Using privacy process patterns for incorporating privacy requirements into the system design process," in *Availability, Reliability and Security, 2007. ARES 2007. The Second International Conference on*, IEEE, 2007, pp. 1009–1017.

[54] H. A. López, F. Massacci, and N. Zannone, "Goal-equivalent secure business process re-engineering," in *International Conference on Service-Oriented Computing Workshops*, Springer, 2007, pp. 212–223.

[55] G. Frankova, M. Séguran, F. Gilcher, S. Trabelsi, J. Dörflinger, and M. Aiello, "Deriving business processes with service level agreements from early requirements," *Journal of Systems and Software*, vol. 84, no. 8, pp. 1351–1363, 2011.

[56] E. Paja, P. Giorgini, S. Paul, and P. H. Meland, "Security requirements engineering for secure business processes.," in *1st International Workshop on Alignment of Business Process and Security Modeling*, Springer, 2011, pp. 77–89.

[57] K. Decreus, G. Poels, M. E. Kharbili, and E. Pulvermueller, "Policy-enabled goal-oriented requirements engineering for semantic business process management," *International Journal of Intelligent Systems*, vol. 25, no. 8, pp. 784–812, 2010.

[58] E. Goettelmann, W. Fdhila, and C. Godart, "Partitioning and cloud deployment of composite web services under security constraints," in *Cloud Engineering (IC2E), 2013 IEEE International Conference on*, IEEE, 2013, pp. 193–200.

[59] A. Rodríguez, E. Fernández-Medina, and M. Piattini, "M-bpsec: A method for security requirement elicitation from a uml 2.0 business process specification," *Advances in Conceptual Modeling–Foundations and Applications*, pp. 106–115, 2007.

[60] A. Rodríguez, I. G.-R. de Guzmán, E. Fernández-Medina, and M. Piattini, "Semi-formal transformation of secure business processes into analysis class and use case models: An mda approach," *Information and Software Technology*, vol. 52, no. 9, pp. 945–971, 2010.

[61] A. Rodríguez, E. Fernández-Medina, and M. Piattini, "Towards cim to pim transformation: From secure business processes defined in bpmn to use-cases," in *BPM*, Springer, 2007, pp. 408–415.

[62] ——, “Analysis-level classes from secure business processes through model transformations,” *Trust, Privacy and Security in Digital Business*, pp. 104–114, 2007.

[63] M. Alam, “Model driven security engineering for the realization of dynamic security requirements in collaborative systems,” in *International Conference on Model Driven Engineering Languages and Systems*, Springer, 2006, pp. 278–287.

[64] C. Wolter, M. Menzel, A. Schaad, P. Miseldine, and C. Meinel, “Model-driven business process security requirement specification,” *Journal of Systems Architecture*, vol. 55, no. 4, pp. 211–223, 2009.

[65] M. Menzel, I. Thomas, and C. Meinel, “Security requirements specification in service-oriented business process management,” in *Availability, Reliability and Security, 2009. ARES’09. International Conference on*, IEEE, 2009, pp. 41–48.

[66] B. Hoisl, S. Sobernig, and M. Strembeck, “Modeling and enforcing secure object flows in process-driven soas: An integrated model-driven approach,” *Software & Systems Modeling*, vol. 13, no. 2, pp. 513–548, 2014.

[67] F. Lins, J. Damasceno, R. Medeiros, E. Sousa, and N. Rosa, “Automation of service-based security-aware business processes in the cloud,” *Computing*, vol. 98, no. 9, pp. 847–870, 2016.

[68] M. Zur Muehlen and M. Indulska, “Modeling languages for business processes and business rules: A representational analysis,” *Information systems*, vol. 35, no. 4, pp. 379–390, 2010.

[69] J. Dorn, C. Grun, H. Werthner, and M. Zapletal, “A survey of b2b methodologies and technologies: From business models towards deployment artifacts,” in *System Sciences, 2007. HICSS 2007. 40th Annual Hawaii International Conference on*, IEEE, 2007, 143a–143a.

[70] W. M. Van der Aalst, “Formalization and verification of event-driven process chains,” *Information and Software technology*, vol. 41, no. 10, pp. 639–650, 1999.

[71] J. Recker, “Opportunities and constraints: The current struggle with bpmn,” *Business Process Management Journal*, vol. 16, no. 1, pp. 181–201, 2010.

[72] M. Salnitri, F. Dalpiaz, and P. Giorgini, “Modeling and verifying security policies in business processes,” in *Enterprise, Business-Process and Information Systems Modeling*, Springer, 2014, pp. 200–214.

[73] R. Braun and W. Esswein, "Classification of domain-specific bpmn extensions," in *IFIP Working Conference on The Practice of Enterprise Modeling*, Springer, 2014, pp. 42–57.

[74] A. Rodríguez, E. Fernández-Medina, and M. Piattini, "A bpmn extension for the modeling of security requirements in business processes," *IEICE transactions on information and systems*, vol. 90, no. 4, pp. 745–752, 2007.

[75] S. H. Turki, F. Bellaaj, A. Charfi, and R. Bouaziz, "Modeling security requirements in service based business processes," *Enterprise, Business-Process and Information Systems Modeling*, pp. 76–90, 2012.

[76] A. R. Souza, B. L. Silva, F. A. Lins, J. C. Damasceno, N. S. Rosa, P. R. Maciel, R. W. Medeiros, B. Stephenson, H. R. Motahari-Nezhad, J. Li, *et al.*, "Incorporating security requirements into service composition: From modelling to execution," in *Service-Oriented Computing*, Springer, 2009, pp. 373–388.

[77] ——, "Sec-mosc tooling-incorporating security requirements into service composition," in *Service-Oriented Computing*, Springer, 2009, pp. 649–650.

[78] C. J. Pavlovski and J. Zou, "Non-functional requirements in business process modeling," in *Proceedings of the fifth Asia-Pacific conference on Conceptual Modelling-Volume 79*, Australian Computer Society, Inc., 2008, pp. 103–112.

[79] C. Wolter, M. Menzel, and C. Meinel, "Modelling security goals in business processes.," in *Modellierung*, vol. 127, 2008, pp. 201–216.

[80] I. Ciuciu, G. Zhao, J. Mülle, S. von Stackelberg, C. Vasquez, T. Haberecht, R. Meersman, and K. Böhm, "Semantic support for security-annotated business process models," in *Enterprise, business-process and information systems modeling*, Springer, 2011, pp. 284–298.

[81] M. Rekik, K. Boukadi, and H. Ben-Abdallah, "Bpmn meta-model extension with deployment and security information," in *13th International Arab Conference on Information Technology ACIT*, 2012.

[82] Y. Cherdantseva, J. Hilton, and O. Rana, "Towards securebpmn-aligning bpmn with the information assurance and security domain," *Business Process Model and Notation*, pp. 107–115, 2012.

[83] A. D. Brucker, I. Hang, G. Lückemeyer, and R. Ruparel, "Securebpmn: Modeling and enforcing access control requirements in business processes," in *Proceedings of the 17th ACM symposium on Access Control Models and Technologies*, ACM, 2012, pp. 123–126.

[84] M. Salnitri, F. Dalpiaz, and P. Giorgini, "Designing secure business processes with secbpmn," *Software & Systems Modeling*, pp. 1–21, 2015.

[85] M. Salnitri, E. Paja, and P. Giorgini, "Maintaining secure business processes in light of socio-technical systems' evolution," in *Requirements Engineering Conference Workshops (REW), IEEE International*, IEEE, 2016, pp. 155–164.

[86] J. L. Vivas, J. A. Montenegro, and J. López, "Towards a business process-driven framework for security engineering with the uml," in *International Conference on Information Security*, Springer, 2003, pp. 381–395.

[87] A. Maña, J. A. Montenegro, C. Rudolph, and J. L. Vivas, "A business process-driven approach to security engineering," in *Database and Expert Systems Applications, 2003. Proceedings. 14th International Workshop on*, IEEE, 2003, pp. 477–481.

[88] J. Lopez, J. A. Montenegro, J. L. Vivas, E. Okamoto, and E. Dawson, "Specification and design of advanced authentication and authorization services," *Computer Standards & Interfaces*, vol. 27, no. 5, pp. 467–478, 2005.

[89] G. Sindre, "Mal-activity diagrams for capturing attacks on business processes," in *International Working Conference on Requirements Engineering: Foundation for Software Quality*, Springer, 2007, pp. 355–366.

[90] M. Q. Saleem, J. Jaafar, and M. F. Hassan, "Security modeling of soa system using security intent dsl," in *International Conference on Software Engineering and Computer Systems*, Springer, 2011, pp. 176–190.

[91] ——, "Security modelling along business process model of soa systems using modified uml-soa-seci," in *Computer & Information Science (ICCIS), 2012 International Conference on*, IEEE, vol. 2, 2012, pp. 880–884.

[92] A. Rodriguez, E. Fernandez-Medina, and M. Piattini, "Security requirement with a uml 2.0 profile," in *Availability, Reliability and Security, 2006. ARES 2006. The First International Conference on*, IEEE, 2006, 8–pp.

[93] A. Rodríguez, E. Fernández-Medina, and M. Piattini, "Capturing security requirements in business processes through a uml 2.0 activity diagrams profile," *Advances in Conceptual Modeling-Theory and Practice*, pp. 32–42, 2006.

[94] ——, "Towards a uml 2.0 extension for the modeling of security requirements in business processes," *Trust and Privacy in Digital Business*, pp. 51–61, 2006.

[95] A. Rodríguez, E. Fernández-Medina, J. Trujillo, and M. Piattini, "Secure business process model specification through a uml 2.0 activity diagram profile," *Decision Support Systems*, vol. 51, no. 3, pp. 446–465, 2011.

[96] M. Jensen and S. Feja, "A security modeling approach for web-service-based business processes," in *Engineering of Computer Based Systems, 2009. ECBS 2009. 16th Annual IEEE International Conference and Workshop on the*, IEEE, 2009, pp. 340–347.

[97] T. Stocker and F. Böhr, "If-net: A meta-model for security-oriented process specification," in *International Workshop on Security and Trust Management*, Springer, 2013, pp. 191–206.

[98] S. Jakoubi, T. Neubauer, and S. Tjoa, "A roadmap to risk-aware business process management," in *Services Computing Conference, 2009. APSCC 2009. IEEE Asia-Pacific*, IEEE, 2009, pp. 23–27.

[99] T. Neubauer and J. Heurix, "Defining secure business processes with respect to multiple objectives," in *Availability, Reliability and Security, 2008. ARES 08. Third International Conference on*, IEEE, 2008, pp. 187–194.

[100] ——, "Objective types for the valuation of secure business processes," in *Computer and Information Science, 2008. ICIS 08. Seventh IEEE/ACIS International Conference on*, IEEE, 2008, pp. 231–236.

[101] A. J. Varela-Vaca, R. M. Gasca, and S. Pozo, "Opbus: Risk-aware framework for the conformance of security-quality requirements in business processes," in *Security and Cryptography (SECRYPT), 2011 Proceedings of the International Conference on*, IEEE, 2011, pp. 370–374.

[102] A. J. Varela-Vaca, "Opbus: A framework for improving the dependability of risk-aware business processes," *AI Communications*, vol. 29, no. 1, pp. 233–235, 2016.

[103] A. J. Varela-Vaca, R. Warschofsky, R. M. Gasca, S. Pozo, and C. Meinel, "A security pattern-driven approach toward the automation of risk treatment in business processes," in *International Joint Conference CISIS12-ICEUTE´ 12-SOCO´ 12 Special Sessions*, Springer, 2013, pp. 13–23.

[104] A. J. Varela-Vaca and R. M. Gasca, "Towards the automatic and opti- mal selection of risk treatments for business processes using a constraint programming approach," *Information and Software Technology*, vol. 55, no. 11, pp. 1948–1973, 2013.

[105] K. Weldemariam and A. Villafiorita, "Procedural security analysis: A methodological approach," *Journal of Systems and Software*, vol. 84, no. 7, pp. 1114–1129, 2011.

[106] E. Goettelmann, N. Mayer, and C. Godart, "Integrating security risk management into business process management for the cloud," in *Business Informatics (CBI), 2014 IEEE 16th Conference on*, IEEE, vol. 1, 2014, pp. 86–93.

[107] B. Marcinkowski and M. Kuciapski, "A business process modeling notation extension for risk handling," *Computer Information Systems and Industrial Management*, pp. 374–381, 2012.

[108] O. Altuhhova, R. Matulevičius, and N. Ahmed, "Towards definition of secure business processes," in *International Conference on Advanced Information Systems Engineering*, Springer, 2012, pp. 1–15.

[109] N. Ahmed and R. Matulevičius, "A taxonomy for assessing security in business process modelling," in *Research Challenges in Information Science (RCIS), 2013 IEEE Seventh International Conference on*, IEEE, 2013, pp. 1–10.

[110] ——, "Securing business processes using security risk-oriented patterns," *Computer Standards & Interfaces*, vol. 36, no. 4, pp. 723–733, 2014.

[111] M. Kirikova, R. Matulevičius, and K. Sandkuhl, "The enterprise model frame for supporting security requirement elicitation from business processes," in *International Baltic Conference on Databases and Information Systems*, Springer, 2016, pp. 229–241.

[112] M. Kirikova, R. Matulevicius, and K. Sandkuhl, "Application of the enterprise model frame for security requirements and control identification.," in *DB&IS (Selected Papers)*, 2016, pp. 129–142.

[113] P. H. Meland and E. A. Gjære, "Representing threats in bpmn 2.0," in *Availability, Reliability and Security (ARES), 2012 Seventh International Conference on*, IEEE, 2012, pp. 542–550.

[114] T. Abe, S. Hayashi, and M. Saeki, "Modeling security threat patterns to derive negative scenarios," in *Software Engineering Conference (APSEC), 2013 20th Asia-Pacific*, IEEE, vol. 1, 2013, pp. 58–66.

[115] Á. J. Varela-Vaca, D. Borrego, M. T. Gómez-López, and R. M. Gasca, "A usage control model extension for the verification of security policies in artifact-centric business process models," in *International Conference on Business Information Systems*, Springer, 2016, pp. 289–301.

[116] Y. Alotaibi, "Business process modelling challenges and solutions: A literature review," *Journal of Intelligent Manufacturing*, vol. 27, no. 4, pp. 701–723, 2016.

[117] ISO, "ISO/IEC 27000 : 2009, Information technology-Security techniques-Information security management systems-Overview and vocabulary," Tech. Rep., 2009.

[118] C. Kalloniatis, "Designing privacy-aware systems in the cloud," in *International Conference on Trust and Privacy in Digital Business*, Springer, 2015, pp. 113–123.

[119] J. Koehler, R. Hauser, J. Küster, K. Ryndina, J. Vanhatalo, and M. Wahler, "The role of visual modeling and model transformations in business-driven development," *Electronic Notes in Theoretical Computer Science*, vol. 211, pp. 5–15, 2008.

[120] E. Yu, "Modelling strategic relationships for process reengineering," *Ph.D. thesis, Department of Computer Science, University of Toronto, Canada*,

[121] H. Mouratidis, S. Islam, C. Kalloniatis, and S. Gritzalis, "A framework to support selection of cloud providers based on security and privacy requirements," *Journal of Systems and Software*, vol. 86, no. 9, pp. 2276–2293, 2013.

[122] A. Cailliau and A. Van Lamsweerde, "A probabilistic framework for goal-oriented risk analysis," in *2012 20th IEEE International Requirements Engineering Conference (RE)*, IEEE, 2012, pp. 201–210.

[123] R. Matulevičius, N. Mayer, H. Mouratidis, E. Dubois, P. Heymans, and N. Genon, "Adapting secure tropos for security risk management in the early phases of information systems development," in *International Conference on Advanced Information Systems Engineering*, Springer, 2008, pp. 541–555.

[124] P. Mell, K. Scarfone, and S. Romanosky, "A complete guide to the common vulnerability scoring system version 2.0," in *FIRST-Forum of Incident Response and Security Teams*, 2007, pp. 1–23.

[125] V. Viduto, C. Maple, W. Huang, and A. Bochenkov, "A multi-objective genetic algorithm for minimising network security risk and cost," in *High Performance Computing and Simulation (HPCS), 2012 International Conference on*, IEEE, 2012, pp. 462–467.

[126] T. L. Saaty, *Analytic hierarchy process*. Wiley Online Library, 1980.

[127] ——, "What is the analytic hierarchy process?" In *Mathematical models for decision support*, Springer, 1988, pp. 109–121.

[128] J. Karlsson and K. Ryan, "A cost-value approach for prioritizing requirements," *IEEE software*, vol. 14, no. 5, pp. 67–74, 1997.

[129] O. S. Vaidya and S. Kumar, "Analytic hierarchy process: An overview of applications," *European Journal of operational research*, vol. 169, no. 1, pp. 1–29, 2006.

[130] C. M. Nguyen, R. Sebastiani, P. Giorgini, and J. Mylopoulos, "Requirements evolution and evolution requirements with constrained goal models," in *35th International Conference om Conceptual Modeling, ER 2016*, 2016, pp. 544–552.

[131] R. Sebastiani and P. Trentin, "Optimathsat: A tool for optimization modulo theories," in *27th International Conference on Computer Aided Verification, CAV*, 2015, pp. 447–454.

[132] N. Yoshioka, H. Washizaki, and K. Maruyama, "A survey on security patterns," *Progress in informatics*, vol. 5, no. 5, pp. 35–47, 2008.

[133] D. M. Kienzle and M. C. Elder, "Security patterns for web application development," *University of Virginia*, 2002.

[134] E. B. Fernandez and R. Pan, "A pattern language for security models," in *In Proc. of PLoP*, vol. 1, 2001.

[135] H. Mouratidis, M. Weiss, and P. Giorgini, "Modeling secure systems using an agent-oriented approach and security patterns," *Int. J. of Software Engineering and Knowledge Engineering*, vol. 16, no. 03, pp. 471–498, 2006.

[136] G Stonebumer, A Goguen, and A Fringa, "Risk management guide for information technology systems," *Recommendations of the National Institute of Standards and Technology*, 2002.

[137] Y. Cherdantseva and J. Hilton, "A reference model of information assurance & security," in *The 8th International Conference on Availability, reliability and security (ARES)*, IEEE, 2013, pp. 546–555.

[138] S. Morimoto, "A Survey of Formal Verification for Business Process Modeling," in *The 8th International Conference on Computational Science*, Springer, 2008, pp. 514–522.

[139] J. Becker, P. Delfmann, M. Eggert, and S. Schwittay, "Generalizability and Applicability of Model-Based Business Process Compliance-Checking Approaches A State-of-the-Art Analysis and Research Roadmap," *BuR-Business Research*, vol. 5, no. 2, pp. 221–247, 2012.

[140] A. Schaad, V. Lotz, and K Sohr, "A model-checking approach to analysing organisational controls in a loan origination process," in *The 11th Symposium on Access Control Models and Technologies*, ACM, 2006, pp. 139–149.

[141] A. Armando and S. E. Ponta, "Model checking of security-sensitive business processes," *Lecture Notes in Computer Science*, vol. 5983, pp. 66–80, 2010.

[142] A. Lehmann and D. Fahland, "Information flow security for business process models - Just one click away," in *The 10th International Conference on Business Process Management*, Springer, 2012, pp. 34–39.

[143] W. Arsac, L. Compagna, G. Pellegrino, and S. E. Ponta, "Security validation of business processes via model-checking.," *ESSoS*, vol. 6542, pp. 29–42, 2011.

[144] L. Compagna, P. Guilleminot, and A. D. Brucker, "Business process compliance via security validation as a service," in *Software Testing, Verification and Validation (ICST), 2013 IEEE Sixth International Conference on*, IEEE, 2013, pp. 455–462.

[145] L. Compagna, D. R. Dos Santos, S. E. Ponta, and S. Ranise, "Cerberus: Automated synthesis of enforcement mechanisms for security-sensitive business processes," in *International Conference on Tools and Algorithms for the Construction and Analysis of Systems*, Springer, 2016, pp. 567–572.

[146] G. Müller and R. Accorsi, "Why are business processes not secure?" *Lecture Notes in Computer Science*, vol. 8260, pp. 240–254, 2013.

[147] H. Groefsema and D. Bucur, "A Survey of Formal Business Process Verification: From Soundness to Variability," in *Proceedings of the 3rd International Symposium on Business Modeling and Software Design*, 2013, pp. 198–203.

[148] R. Van Solingen, V. Basili, G. Caldiera, and H. D. Rombach, "Goal question metric (gqm) approach," *Encyclopedia of software engineering*, 2002.

[149] A. Ishizaka and A. Labib, "Analytic hierarchy process and expert choice: Benefits and limitations," *Or Insight*, vol. 22, no. 4, pp. 201–220, 2009.

Appendix

Draft Case Study Outputs of Goal Modelling Component Application

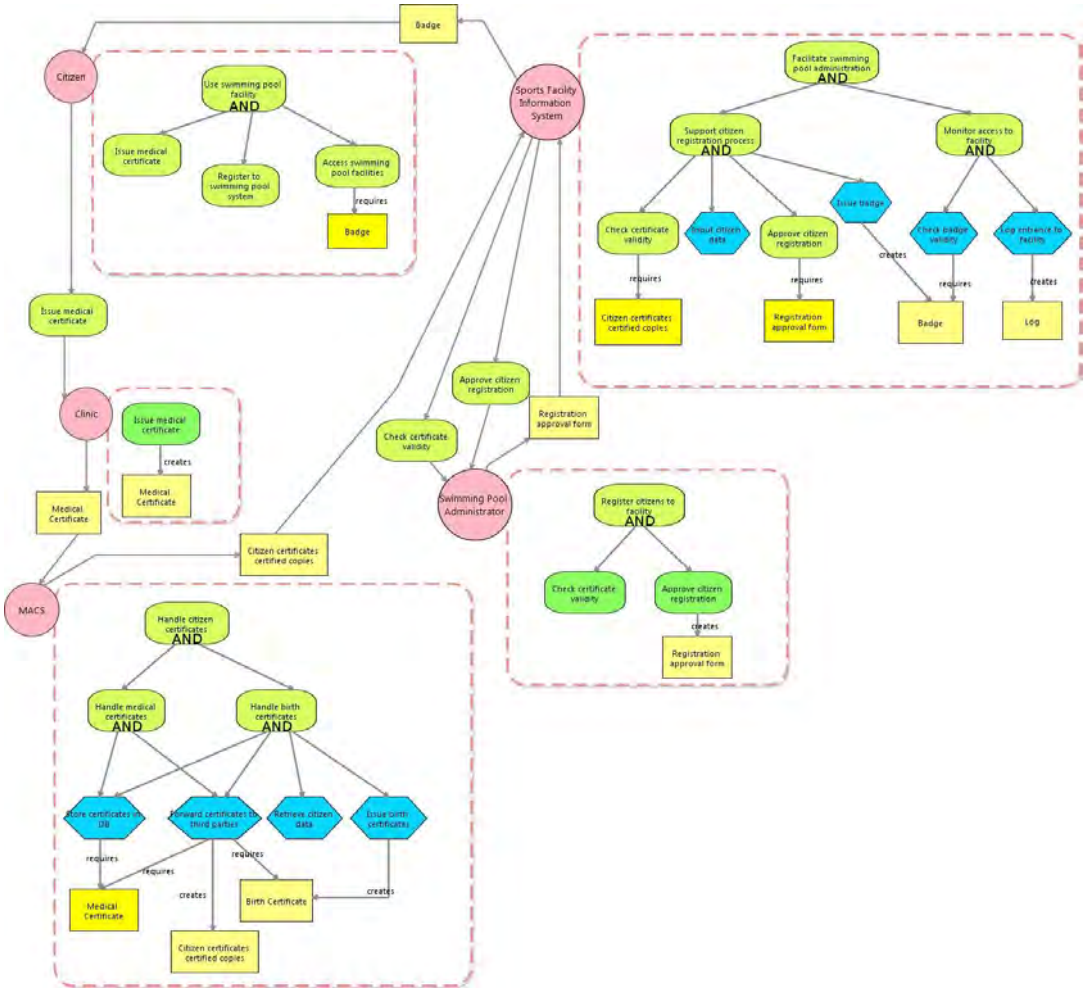

Figure 5.1: First draft of SPA system goal model

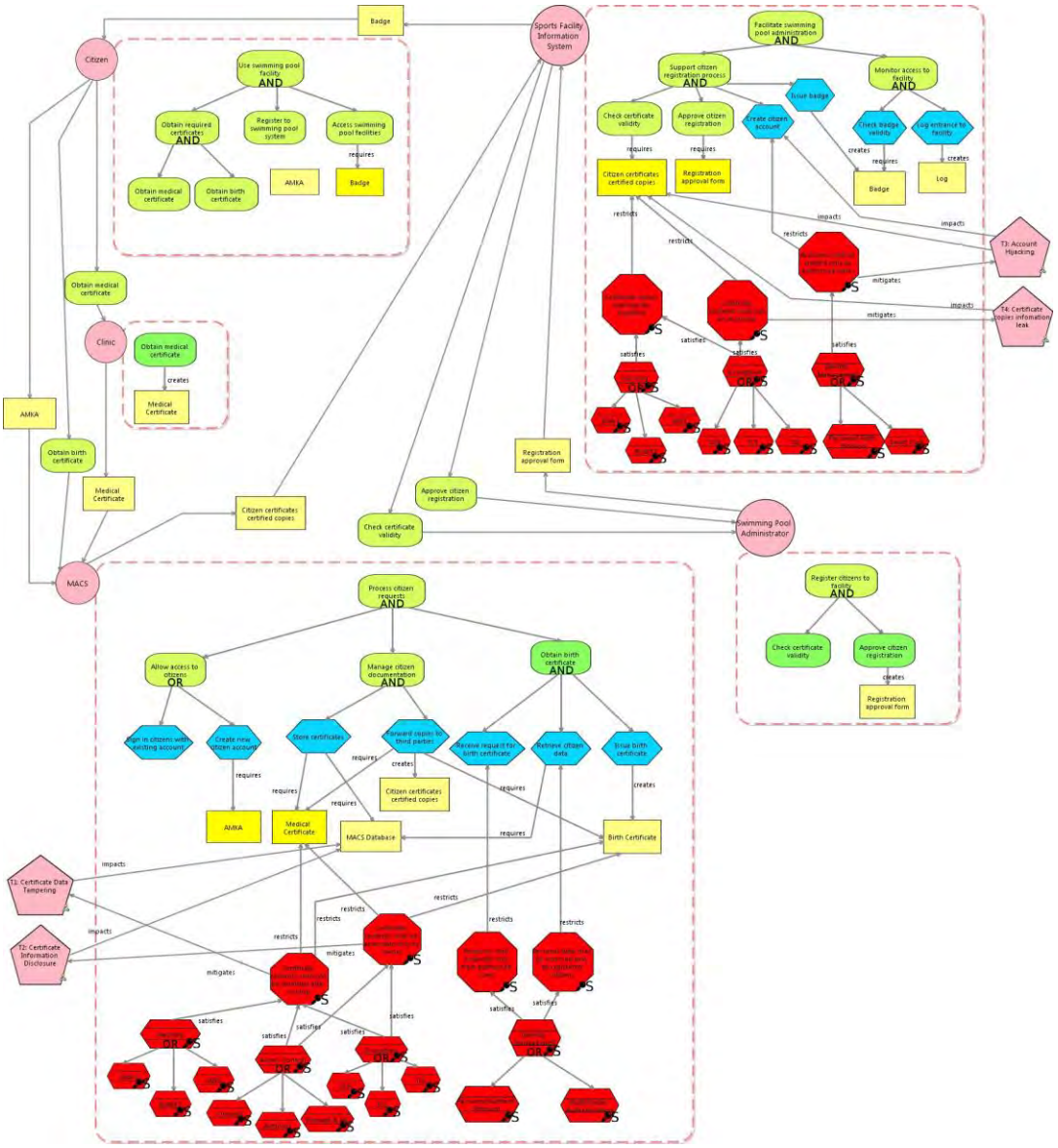

Figure 5.2: Second draft of SPA system goal model

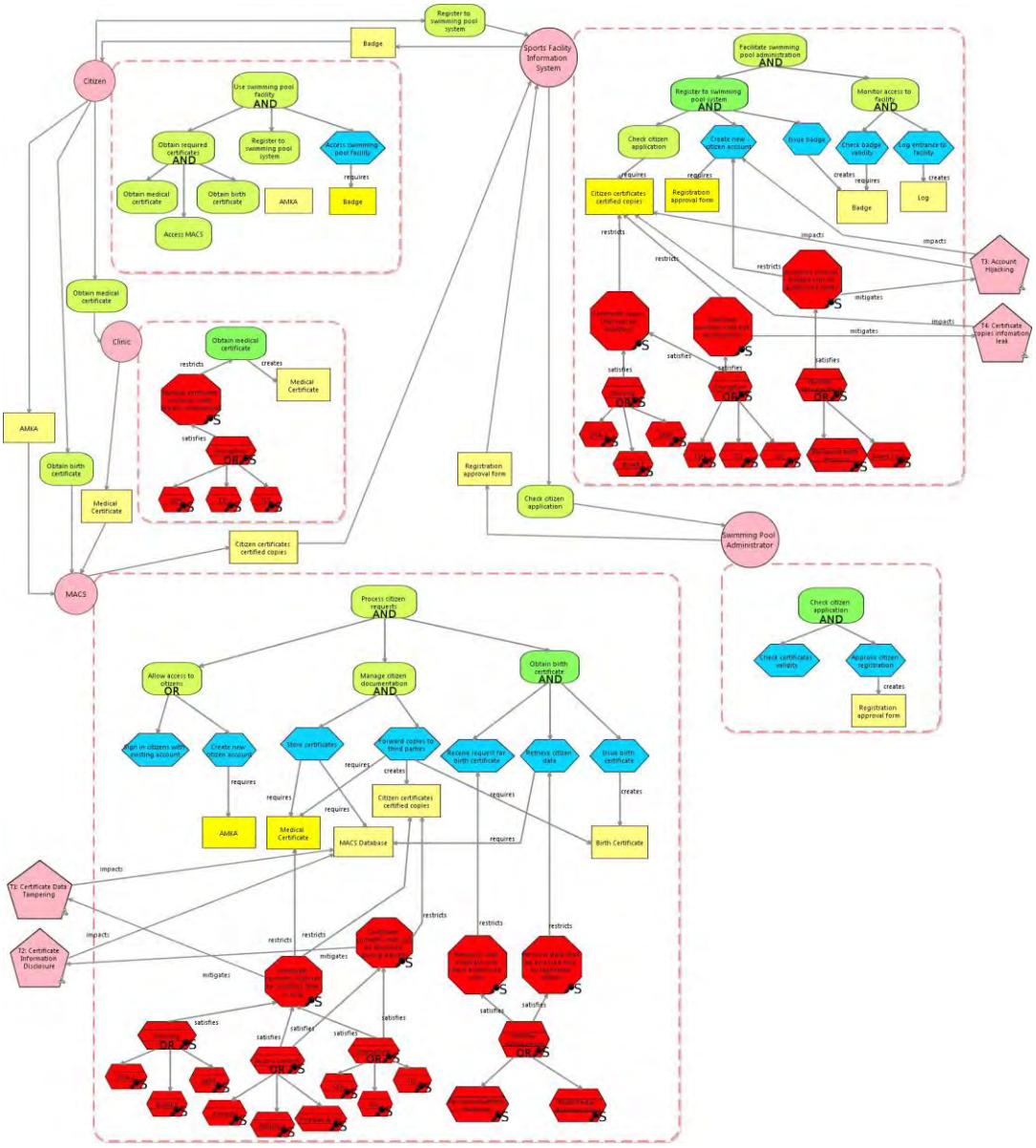

Figure 5.3: Third draft of SPA system goal model

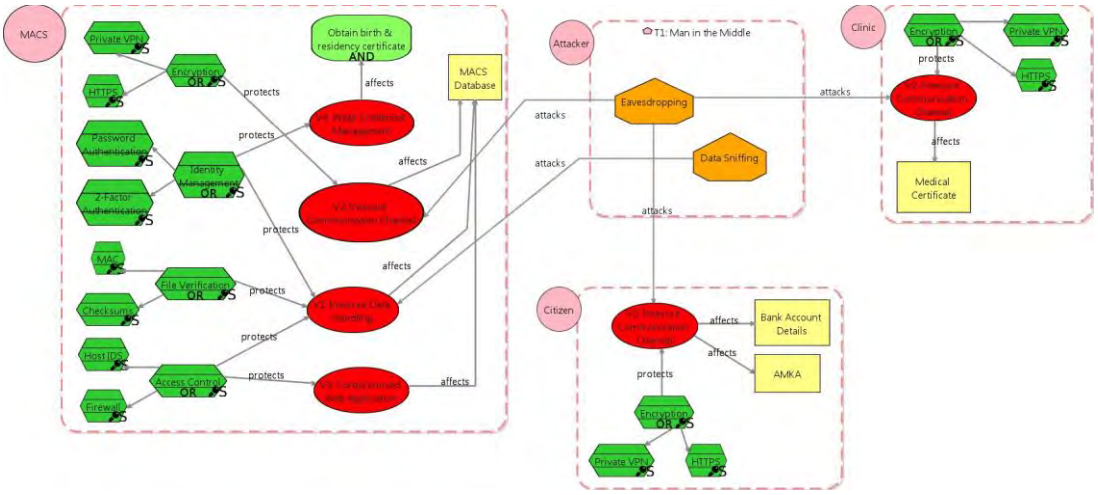

Figure 5.4: First draft of SPA system security attacks view for T1

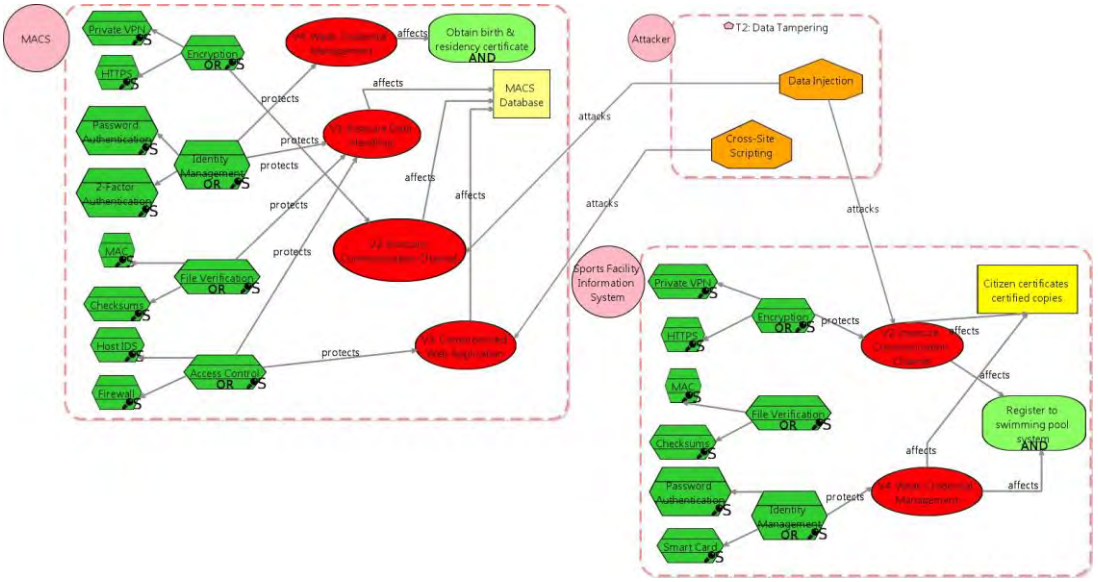

Figure 5.5: First draft of SPA system security attacks view for T2

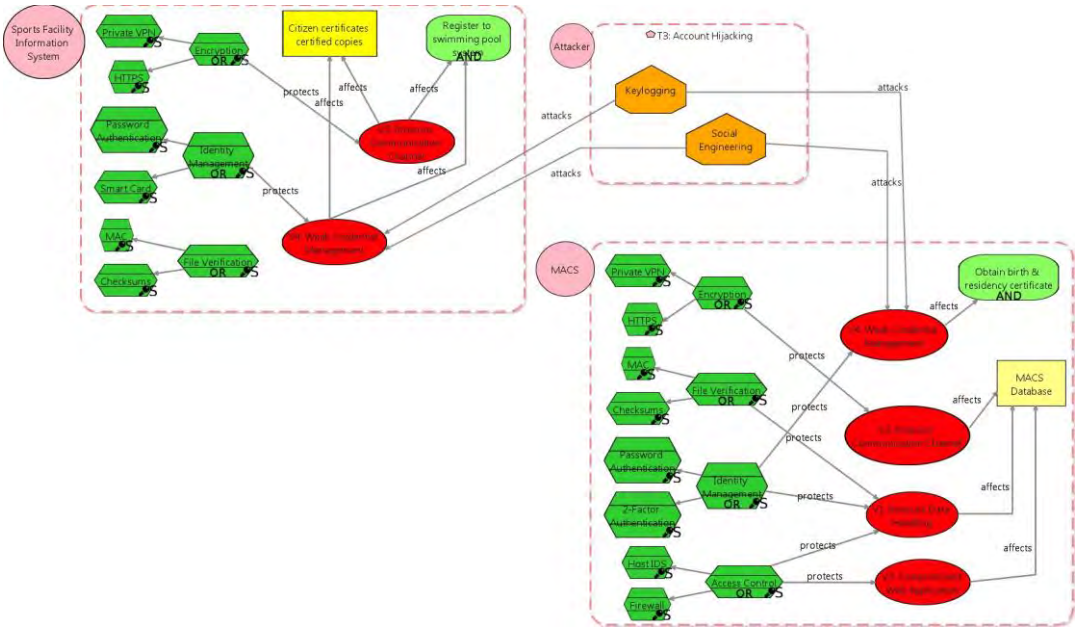

Figure 5.6: First draft of SPA system security attacks view for T3

Figure 5.7: First draft of SPA system hybrid reference process model

Case Study Outputs of Business Process Modelling Component Application

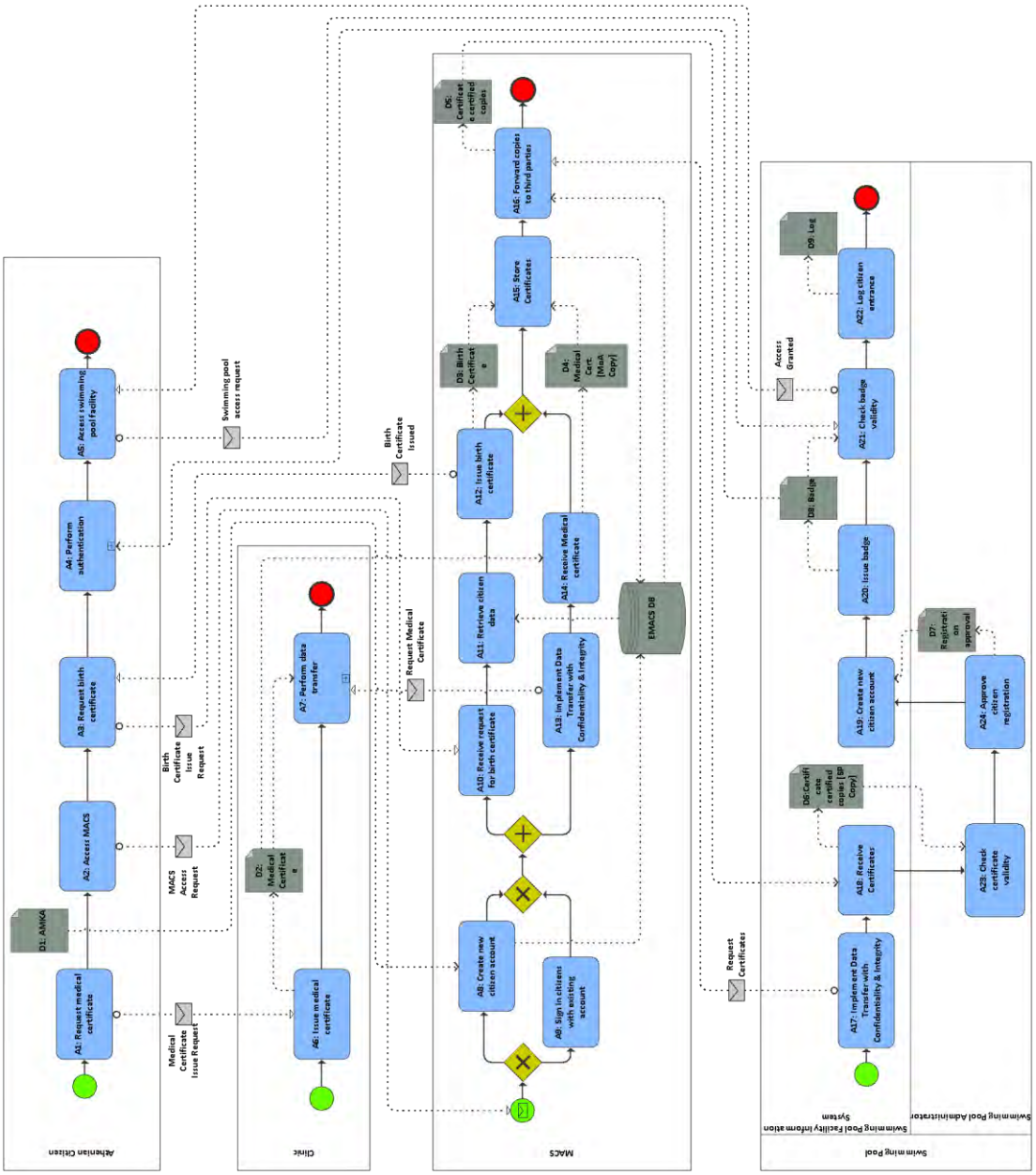

Figure 5.8: First draft of SPA system business process model

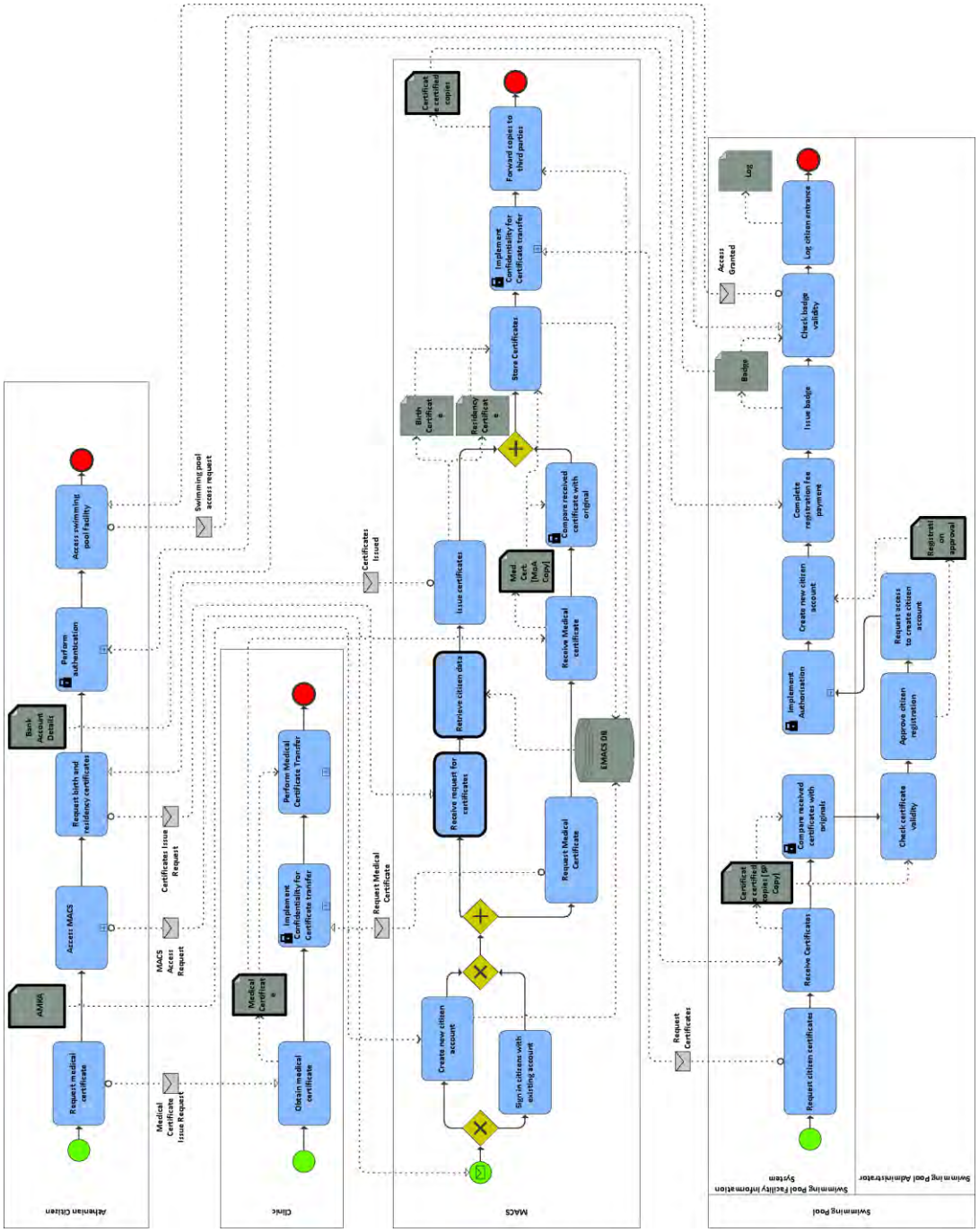

Figure 5.9: Second draft of SPA system business process model