Lightweight Modeling and Analysis of Security Concepts[1]

Jörn Eichler

Fraunhofer Institute for Secure Information Technology SIT,
Rheinstr. 75, 64295 Darmstadt, Germany
Joern.Eichler@sit.fraunhofer.de

Abstract. Modeling results from risk assessment and the selection of safeguards is an important activity in information security management. Many approaches for this activity focus on an organizational perspective, are embedded in heavyweight processes and tooling and require extensive preliminaries. We propose a lightweight approach introducing SeCoML – a readable language on top of an established methodology within an open framework. Utilizing standard tooling for creation, management and analysis of SeCoML models our approach supports security engineering and integrates well in different environments. Also, we report on early experiences of the language's use.

Keywords: Risk Assessment, Information Security Management, Security Engineering, DSML

1 Introduction

Flexibility and adaptability are key drivers for success of today's enterprises. Therefore smart and tailored processes and applications are crucial, especially for small and medium sized enterprises (SME) [22]. The execution of information technology (IT) related projects to develop or adapt applications to the needs of the organization, the integration of applications to better support business processes, and the adaptation of IT-supported business processes to changing needs in the market is a recurring task in order to provide the necessary infrastructure. Those projects are confronted with scarce resources, especially on expert knowledge outside the organization's core competences. Hence, security engineering activities have to be addressed on the basis of restricted knowledge, delivering quick results and a flexible integration in existing tool chains and processes [4].

A vital part of the security engineering activities in these projects is the assessment and treatment of IT security risks [2]. To conduct risk assessment and risk treatment

[1] The work presented in this paper was partly developed in the context of the project Alliance Digital Product Flow (ADiWa) that is funded by the German Federal Ministry of Education and Research. Support code: 01IA08006F. The original publication is available at http://www.springerlink.com/index/X7068743G7P38470.pdf.

IT security related information about the environment is necessary. Security models created to support information security management systems (ISMS) can provide valuable information [15]. To distinguish security models for that purpose from others the term security concept is common and will be used in the following [8]. Security concepts capture corporate assets, their dependencies and protection requirements, threats to the assets, as well as necessary and implemented safeguards to protect them. Using security concepts, environmental protection requirements and design constraints for new or adapted applications based on consolidated information can be introduced early in the security engineering process. Furthermore, impacts on the environment by the applications to be developed or adapted can be analyzed, and different design alternatives can be evaluated.

The development and integration of a new service into the organization's environment to exchange data between business partners might serve as example. The new service has to meet different security requirements depending e.g. on the protection requirements of the data to be communicated as well as those of the applications and business processes utilizing this service. Furthermore the integration of the new service affects the existing infrastructure as new requirements for systems or communication nodes supporting the new service might be imposed.

A variety of methods and tools for risk assessment in the context of ISMS have been presented focusing on different aspects of risk assessment or targeting different environments [14]. Unfortunately only very few methodologies are suitable for the use in SME environments. Moreover, candidate methodologies identified in the following sections focus on an organizational perspective and bring only general or heavyweight tooling to support users documenting their security concept and analyzing it.

We propose a lightweight approach for modeling security concepts introducing the Security Concept Modeling Language (SeCoML). SeCoML is a domain specific modeling language (DSML) with a readable textual syntax. It is based on the IT Baseline Protection Methodology (IT-BPM, also known as IT Grundschutz [8]), a well known methodology in SME environments conformant to accepted international standards. SeCoML allows for an easy creation, validation, and analysis of security concepts. Its modularity provides support for reusability of models, incremental creation of security concepts, and adaptability. Applying current frameworks for model driven software development, state of the art tooling is provided to use SeCoML effectively, and to integrate it in existing tool chains.

The rest of this paper is structured as follows. Section 2 identifies requirements, selects an appropriate information security management (ISM) approach, provides some background on IT-BPM, and discusses related work. SeCoML and corresponding tooling is presented in section 3. In section 4, we report early experiences using SeCoML in SME environments. We summarize our results in section 5 and discuss further research topics.

2 Requirements, Background, and Related Work

In this section, at first we identify basic requirements considering a lightweight approach to model security concepts. Then, IT-BPM as underlying ISM approach for SeCoML is selected and some background on IT-BPM is provided. The section closes with a discussion of related work.

2.1 Basic Requirements

To allow for a beneficial use of security concepts in integration and adaption projects of SMEs we identified the following basic requirements for a lightweight modeling and analysis approach:

(R1) The security concept needs an established methodology as foundation that is appropriate for the use in the targeted environment.
(R2) At least a semi-formal modeling of the security concept must be possible.
(R3) Incremental creation, refinement, and analysis of security concepts as well as modular partitioning of security concepts should be supported.
(R4) (Technical) assistance in the creation of valid security concepts should be available.
(R5) Security concepts and tooling must integrate in existing tool chains and processes without extensive preliminaries.

Generally security concepts are created and maintained in the course of the initiation and operation of an ISMS. A modeling approach should therefore build upon an established methodology for the targeted environment to become feasible and accepted (R1). Furthermore only an at least semi-formal modeling of security concepts allows for a precise common understanding and (semi-) automated validation and analysis [5] (R2). Usually security concepts are created in multiple (incremental) steps involving different stakeholders. In the lifetime of an ISMS security concepts are refined for different purposes (reporting, auditing, analysis etc.) [8]. Initially only core assets are included into the security concept and other assets, additional threats etc. are refined on demand. Parts of the security concepts might be reused to reflect organizational subdivisions and subsidiaries. Incremental creation, refinement, and modular partitioning are also important to support differing scenarios (e.g. to analyze design alternatives in integration and adaption projects) (R3).

To leverage security concepts the creation, manipulation, and validation should be assisted by respective tooling targeting not only security experts [29] (R4). To allow for a lightweight approach modeling artifacts as well as corresponding tooling must be easy to integrate into existing tool chains (e.g. development tool chains including application lifecycle management, document management, management information systems for security certification and ISMS operation), and independent of individual tooling or heavyweight processes (e.g. requiring multiple stakeholders to participate in every project and several activities or presupposing large documentation) (R5).

2.2 Information Security Management

As security breaches based on IT issues gain media interest, more and more organizations are introducing internal initiatives to improve their protection of their IT infrastructure, processed data, and other IT-related assets. One keyword accompanying those initiatives is ISM. Understanding information security (IS) as preservation of confidentiality, integrity, and availability of information as well as other properties as authenticity and accountability, information security management refers to the process of the implementation and ongoing management of IS in an organization [19].

Targeting not only the (cost efficient) improvement of their IS but also indications of trust that consumers or partners can have in the organization's IS, standards and corresponding certifications are becoming an important foundation of the initiatives. Important international standards in the field of ISM are:

- ISO/IEC 13335-1 [18] is a general guide for initiating and implementing the IT security management process focusing on concepts and models of IT security. Other parts of the current ISO/IEC 13335 series describes techniques for IT security risk management and guidance for network security.
- ISO/IEC 27001 [19] provides requirements for an ISMS. An ISMS is part of the overall management system to establish, implement, operate, monitor, review, maintain and improve IS. ISO/IEC 27001 is one of the very few international ISM standards that allow for certification. ISO/IEC 27001 is part of a series that includes also complementing standards on risk management, metrics, measurement, and implementation guidance.

Analyzing risk assessment and management methodologies best suited for SMEs in Europe, the European Network and Information Security Agency (ENISA) identified six methodologies [14]: the Austrian IT Security Handbook [9], the Dutch A&K Analysis [24], EBIOS [12], IT-BPM, MEHARI [11], and OCTAVE-S [1]. We used ENISA's report to select from the methodologies those that fulfill the requirements R1 and R5 and therefore

- provide special information for SMEs,
- are compliant to international standards listed above,
- provide interfaces to other organizational processes, and
- are applicable without consultancy support.

Only the Austrian IT Security Handbook and IT-BPM comply with all criteria. MEHARI and the A&K Analysis for example lack interfaces to other organizational processes, OCTAVE-S is not compliant to international standards, and Ebios requires consultancy support. IT-BPM is widely used not only in Germany but also in Austria, Switzerland, and other countries, documentation is also given in English, and technical guidelines covering various application domains (e.g. [3]) as well as examples including security concepts for current developments and real world applications are publicly available (e.g. [16]). We therefore decided to use the IT-BPM as basis for our approach.

2.3 IT Baseline Protection Methodology

The ISO standards explicate ISM and ISMS generically. IT-BPM tries to bridge the gap between those generic descriptions and a practical implementation. It provides detailed guidance for an organization for establishing an ISMS compliant with ISO/IEC 27001.

The IT-BPM security process comprises four phases: process initiation, creation of the security concept, implementation of the security concept and maintenance and improvement of the security concept (cf. Figure 1). The creation of the security concept encompasses six steps. The structure analysis as first step details interdependencies between business processes, applications, and existing IT infrastructure. For those items the protection requirements are determined using an ordinal rating scale and – based on the IT-BPM catalogues – appropriate safeguards are selected and adapted. As IT-BPM uses an (extended) baseline approach the following basic check generates a target/actual comparison. If the protection requirements are exceeding the normal rating or elements of the organization's structure are not covered by the catalogues a supplementary risk analysis is executed to assess the risks and integrate corresponding safeguards in the security concept.

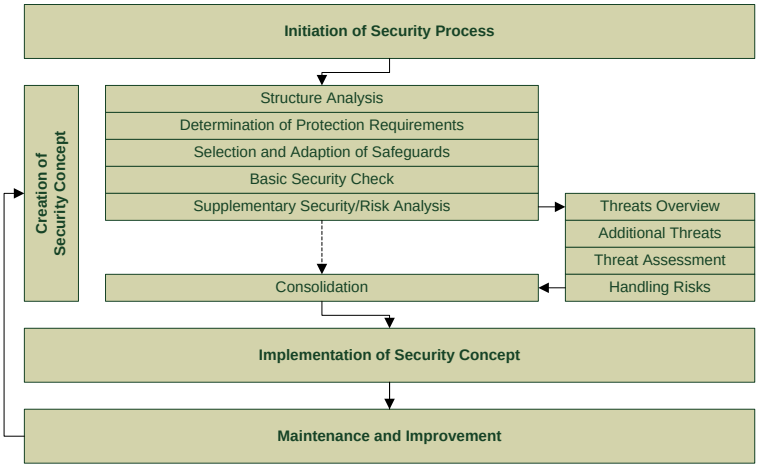

Fig. 1. Security process of the IT-BPM [8]

To implement the IT-BPM most organizations use the checklists and forms of the IT-BPM directly (using word processors and spreadsheets) or deploy a dedicated ISMS tool that supports the IT-BPM. An approach that offers (semi-) formal modeling and fulfills the requirements given in section 2.1 is not available.

2.4 Related Work

Beside the risk assessment and ISM methodologies discussed previously, several model- and/or modeling language-based approaches have been presented recently.

Zambon et al. present in [31] a model-based risk assessment approach for IT infrastructures. Introducing QualTD they suggest a time dependency model and techniques to analyze risks on availability and the propagation of incidents in an IT architecture. Confidentiality or integrity are not considered in QualTD models. In [7] den Braber et al. present CORAS, a method to conduct security risk analysis that uses a customized graphical DSML based on UML. CORAS follows the AS/NZS 4360:2004 standard [28] that does not offer special support for the targeted environment. Other authors propose further modeling approaches and tooling in the area of risk assessment but do not use a methodology appropriate for the targeted environment or do not provide support for incremental creation or modular partitioning of their models [13, 23, 25].

Further model-driven approaches highlight the gap between security models, system design models and implementation. The focus is on the generation of implementation artifacts from security (enriched) models providing modeling languages and transformations. Prominent in this domain is SecureUML presented by Basin et al. [6] for access control modeling and infrastructure generation. Wolter et al. [30] derive security policies from business process models with security annotations; Rodriguez et al. use analogue models to derive design models [27]. Creation and analysis of security models using DSMLs and OCL is presented also in [5], targeting access control. UMLsec presented by Juerjens [20] uses extensions to UML to include security relevant information in design models and to analyze security properties of those models. In [17] UMLsec is integrated with the heuristic requirements editor HeRA and the security standard ISO 14508 Common Criteria to comprise with SecReq a security engineering methodology to elicit security requirements and trace them to design models. Together the approach provides valuable insights and guidance for security engineering but is rather heavyweight and directed to security professionals.

The aim to support flexible and adaptable applications and processes touches the agenda of Agile development and Agile security engineering (e.g. [10]). The special requirements of the targeted environment with regard to security engineering are also analyzed by Bartsch et al. in [4] addressing authorization rules for SME applications providing a dedicated DSML and a corresponding enforcement implementation.

3 Modeling Security Concepts with SeCoML

To support a lightweight approach to model and analyze security concepts based on IT-BPM in SME environments we developed SeCoML. In the following we will present the modeling language SeCoML, describe how to analyze security concepts modeled using SeCoML in the course of development, adaption, and integration projects, and sketch the tooling that we implemented for an effective use of SeCoML.

3.1 The Modeling Language

The design of SeCoML was guided methodically by the software language engineering approach from Kleppe in [21]. We present SeCoML describing the abstract syntax

model (provided as Ecore[2] metamodel) first. A short summary of properties of the concrete syntax model and the syntax mapping of SeCoML follows. Examples of security concepts modeled using SeCoML are given in section 3.3. Informal descriptions detailing the semantics are given in [8]. As formal semantics for IT-BPM security concepts are not defined we documented our understanding formulating constraints for the metamodel using OCL [26].

The concepts necessary to model an IT-BPM security concept are reflected in the metamodel of SeCoML. Figure 2 shows an overview of the core concepts and their relations. The following paragraphs will give a short introduction to the metamodel following the security process of IT-BPM (cf. the phase "Creation of Security Concept" in Figure 1).

The structure analysis is the first step of the creation of a security concept. The `Asset` and its relation `supports` in the metamodel capture the results of the structure analysis (i.e. the organization's assets and their interdependencies considering protection requirements).

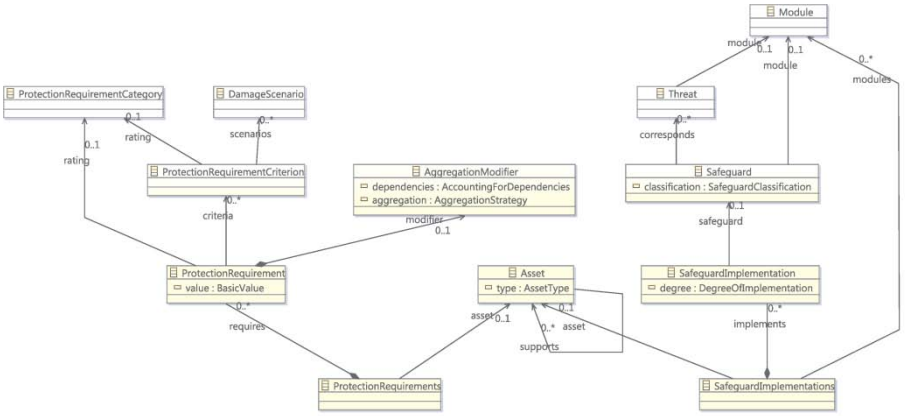

Fig. 2. Core metamodel of SeCoML

In the next step of the security process, protection requirements (`ProtectionRequirements`) considering the basic values confidentiality, integrity, and availability are analyzed and documented for each asset. Protection requirements are rated using an ordinal rating scale from normal to very high (`ProtectionRequirementCategory`). The ratings are based on applicable criteria (`ProtectionRequirementCriterion`) depicted in damage scenarios (`DamageScenario`). The derivation of protection requirement ratings follows aggregation strategies given by IT-BPM (`AggregationStrategy`). In the normal case the rating is derived from the highest rating of applicable criteria and dependent assets' ratings (called "maximum principle" and "accounting for dependencies"). Alternative aggregation strategies can be chosen for each protection requirement to cover risk accumulation or

[2] http://www.eclipse.org/modeling/emf/

distribution increasing or decreasing derived ratings. Also, protection requirements can be rated independently from dependent assets (`AggregationModifier`).

Appropriate safeguards to meet the protection requirements are selected in the following step of the security process. Therefore, modules from the IT-BPM catalogues are assigned to assets (`Module`). Modules cover consolidated threat scenarios and recommended safeguards for various components, procedures, and IT systems (`Threat`, `Safeguard`). Safeguards are tagged to indicate their importance with regard to later certifications of the organization's ISMS (`SafeguardClassification`).

To perform a basic security check the implementation status for each safeguard is evaluated (`SafeguardImplementation`). The degree of implementation is recorded using one of the following statuses: `Unnecessary`, `Yes`, `Partially`, and `No`. After this evaluation a target/actual comparison for the aspired classification level can be derived and a corresponding implementation plan can be developed.

One distinguishing property of SeCoML is the support for the derivation of protection requirement ratings following the different aggregation strategies of the IT-BPM. Ratings are derived using the "maximum" and "accounting for dependencies" principle per default but consider also deviating aggregation strategies. "Cumulation" increases, "distribution" decreases the derived rating. The following example demonstrates how the semantics are defined in the metamodel:

```
context ProtectionRequirement::maxRatingCriteria() : ProtectionRequirementCategory
  body: if criteria->notEmpty() then criteria.rating->sortedBy(ordinal)->first()
  else rating endif
context ProtectionRequirement::maxRatingSA() : ProtectionRequirementCategory
  body: requirements.asset.supports.requires.requires->select(value=self.value)
  .criteria.rating->sortedBy(ordinal)->first()
context ProtectionRequirement::maxDerivedRating() : ProtectionRequirementCategory
  body: if modifier.dependencies=AccountingForDependencies::NoDependencies
  then maxRatingCriteria()else maxRatingCriteria()->union(maxRatingSA()->
  asSet())->sortedBy(ordinal)->first() endif
context ProtectionRequirement::derivedRating : ProtectionRequirementCategory
  derive: if modifier.aggregation=AggregationStrategy::Cumulation
  then maxDerivedRating().higher()
  else if modifier.aggregation=AggregationStrategy::Distribution
  then maxDerivedRating().lower()else maxDerivedRating() endif endif
context ProtectionRequirements
```

The textual syntax of SeCoML has been defined using a domain specific language provided by Xtext[3] resembling the Extended Backus-Naur Form. A concept for namespaces has been included to support recurring names in different packages. To allow for better reusability, modularization, and separation of concerns models can be split using multiple resources (e.g. files or databases). The metamodel respects the incremental nature of results of different phases of the security process. One security concept can be split into several resources, each resource can potentially be used in several security concepts (e.g. reusing the threat and safeguard catalogues). Examples of SeCoML resources are presented in Figure 3.

In comparison with graphical approaches the textual syntax of SeCoML has the main advantages that models can be manipulated, searched, compared, and put under

[3] http://www.eclipse.org/Xtext/

version control using efficient and commonly used tools and techniques. Furthermore it is impossible to break the model in such a way that it cannot be reopened with the delivered editors (and other editors as well), and that they can be fixed using the same tools if the metamodel or the syntax is adapted in future versions. It thus fits very well to our lightweight approach and frequently changing environments.

3.2 Analysis of Security Concepts

An obvious aim of the analysis of security concepts is to support mandatory steps in the security process (e.g. checking whether all assets are analyzed concerning their protection requirements and safeguard implementations, or the generation of target/actual comparisons to develop implementation plans etc.).

Not that obvious are questions concerning dependencies within the security concept (e.g. dependencies of assets' protection requirement ratings and the influence of given rating criteria or the choice of aggregation strategies). The evaluation of design or implementation alternatives requires answers to these questions. We provide with SeCoML a lightweight basis to analyze those questions.

The following paragraphs introduce a short example security concept and exemplify several analysis operations. Generally, security concepts entail much more elements and therefore underline the necessity for corresponding analysis support (e.g. more than 3000 threats and safeguards are given in the IT-BPM catalogues, more than 25 criteria are recommended as starting point).

Asset	**Supports**	**Requires**	**Implements**
A1: Production		R1: C:N (C2, C3), R2: I:N (C2), R3: A:H (C1)	S1:Y, S3:Y
A2: Server	A1	R4: C:N, R5: I:N, R6: A:N:Dis	S1:Y, S2:U, S3:Y
A3: Communication	A2	R7: C:N, R8: I:N, R9: A:N	S1:Y, S2:Y, S3:Y

Table 1. Example security concept (assets, requirements and safeguard implementation)

Table 1 shows a very small example security concept. All elements have an identifier (given before the colon). The basic values are abbreviated using their initial character (Confidentiality, Integrity, Availability) as well as the rating (Normal, High, Very high). In brackets aggregation strategies (Acc: Accumulation, Dis: Distribution) as well as criteria (Cx) are given to substantiate the rating. Safeguards (Sx) are given with the degree of implementation (Y: Yes, N: No, P: Partly, U: Unnecessary). All safeguards are from module M1 and cover the following threats: S1 and S2 threat T1, S2 and S3 threat T2, both from module M1 as well. The criteria C2 and C3 apply for assets with normal protection requirements, C1 applies for those with high protection requirements.

In the example asset A1 (a production application) requires normal protection with regard to confidentiality following from criteria C2 and C3 (requirement R1) but high protection with regard to availability following from criteria C1. It implements safe-

guards S1 and S3. The corresponding server A2 runs the production application A1, its protection requirements derive from the supported assets (A1). The protection requirement rating considering availability is reduced because of the risk distribution aggregation strategy (R6).

In addition to the validations based on the constraints of the metamodel analysis, operations can be used to answer questions within the evaluation of security concepts. The following examples show some of the common queries in the analysis of security concepts that are provided with SeCoML. Ad-hoc queries can utilize these operations as well to further analyze the security concept.

- Which asset lack consideration for a given safeguard depending on the chosen modules for that asset? The operation `Safeguard::getMissingSafeguardImplementations` returns the implementation of {A1} since the security concept does not consider safeguard {S2}.
- If a given safeguard fails, are there assets that will be unprotected with regard to a threat (i.e. implement no other safeguard that covers that threat)? In our example the failure of S1 results in the assets {A1, A2} that are not protected with regard to threat {T1} (`Safeguard::unprotectedAssetsOnFail()`).
- Which protection requirements are affected (directly and indirectly) by altering a given protection requirement criterion definition, leading possibly to changed protection requirement ratings? Changing criterion C2 affects {R1, R2, R4, R5, R7, R8} in our example (`ProtectionRequirementCriterion::affectedRequirements()`).

3.3 Implementation and Integration in the Tool Chain

To use SeCoML efficiently, we implemented an editor for security concepts formulated in SeCoML and integrated the editor with application lifecycle management tooling (versioning, branching, tagging and merging based on Apache Subversion[4], lightweight project management based on Edgewall Software Trac[5] and Eclipse Mylyn[6]) as well as an analysis console (using Interactive OCL from Eclipse MDT[7]) together in a security workbench as Eclipse application.

The core component – the SeCoML editor – was implemented using Eclipse Modeling Framework[8] (EMF) and Xtext[9]. We customized several parts of the editor including labeling and filtering of entities in outlines and content assist, scoping for content assist, formatting of the concrete syntax, validation, quick fixes, and templates to insert new entities quickly. Figure 3 shows a screenshot of the workbench.

The upper part (1) shows the textual editor for SeCoML demonstrating content assist (pop-up showing possible basic values) and constraint-based validation (under-

[4] http://subversion.apache.org/
[5] http://trac.edgewall.org/
[6] http://www.eclipse.org/mylyn/
[7] http://www.eclipse.org/modeling/mdt/?project=ocl
[8] http://www.eclipse.org/modeling/emf/
[9] http://www.eclipse.org/Xtext/

lined text). For easy navigation in textual representations of the security concept the outline view can be used (4). Another view (2) presents an alternative, synchronized tree editor for the same resource that is analyzed using an interactive console (3).

To integrate the workbench further more in the existing tool chain we generated and adopted transformations using EMF and XSL[10] to acquire information given e.g. in Microsoft Excel files (excerpts from a configuration management database) and transform it into SeCoML and back (e.g. for target/actual comparisons).

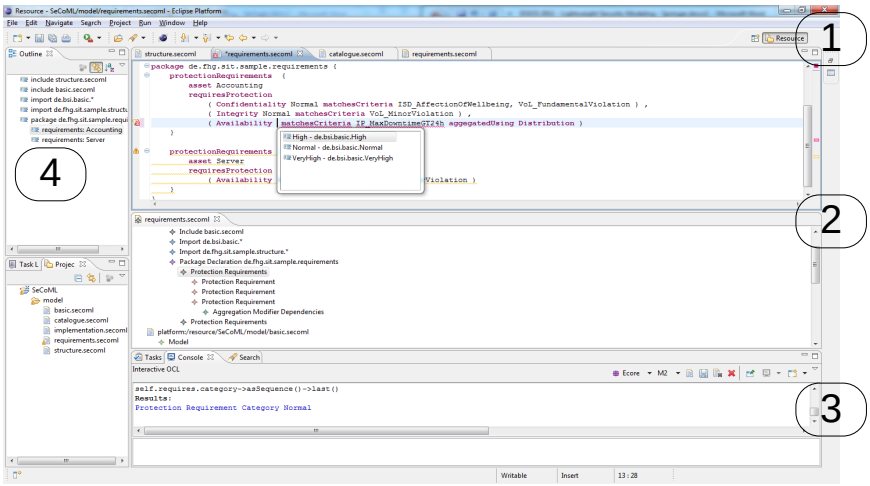

Fig. 3. Editor and analysis workbench for SeCoML

Considering the requirements (cf. section 2.1), SeCoML is able to meet all of them: With IT-BPM an established methodology that is appropriate for the use in the targeted environment is used as foundation (R1). The definition of SeCoML comprising an Ecore metamodel enriched with OCL constraints, a corresponding concrete syntax model and mapping model provides solid ground for semi-formal modeling of security concepts (R2). Incremental creation and refinement as well as modular partitioning of security concepts is supported by SeCoML given the modular metamodel, the support for namespaces, and the possibility to split security concepts into multiple resources (R3). Delivering state of the art editors for SeCoML offers assistance in the creation and validation of security concepts (R4). The combination of the modular metamodel, the corresponding textual syntax and the implementation of accompanying tooling on the basis of an open and well established (model driven software development) framework allows for an easy integration in existing tool chains without extensive preliminaries (R5).

[10] http://www.w3.org/Style/XSL/

4 Early Experience with SeCoML

In order to evaluate SeCoML and its corresponding tooling with respect to its use in the targeted environments, we employed SeCoML in projects with SMEs. In the latest case the SME executed a project to integrate a new communication service to exchange production data with business partners. Therefore the existing security concept of the information security management system should be validated and different solutions for the service integration should be analyzed and presented.

In that project we used SeCoML mainly together with the security officer of the SME, a project manager and several domain experts (for IT infrastructure, operations and the production application). The security officer as well as domain experts for the IT infrastructure used SeCoML regularly as end users. Based on the security concept modeled in SeCoML we discussed alternatives to deploy the new communication service and integrated the chosen alternative in the security concept. Additionally we conducted interviews with the participants to capture their subjective views of the feasibility of the application of SeCoML.

In the course of the project SeCoML proved to be very helpful. Several errors and inconsistencies in the existing security concept were identified. Most errors had been induced by the wrong application of aggregation strategies for protection requirement ratings but also protection requirement criteria had been applied inconsistently. The use of a known methodology and terminology as basis for SeCoML helped to work with SeCoML very quickly. Additionally, the ad-hoc analysis gave solid ground to the discussions about implementation alternatives and the propagation of protection requirement ratings. Providing the security workbench allowed for an easy integration in the existing tool chain (e.g. versioning security concepts for different scenarios along with the sources in the repository).

On the downside the application of SeCoML in larger enterprises had been questioned as the textual approach does not allow for a fine grained access control to elements of the security concept. Also, the appearance of the security workbench attracts people with background as software developer more. For long-term maintenance of the security concept the stakeholders voted for an additional form-based user interface.

Within the project executed, the given environment, and the intended use the participants rated SeCoML as a very helpful approach to improve the projects execution and result.

5 Summary and Outlook

With SeCoML we contribute a lightweight approach to model security concepts on top of an established methodology appropriate for the environment of SMEs. Providing with SeCoML a textual DSML for security concepts establishes a solid foundation for security concept modeling and analysis. The modular design of the metamodel in combination with corresponding properties of the textual syntax supports the lightweight approach. SeCoML integrates well in existing tool chains and processes

and delivers state of the art tooling for the creation, validation and analysis of security concepts. Therefore, SeCoML leverages the use of security concepts in the course of development, adaption, and integration of applications and supports corresponding security engineering activities. Early experiences underline the suitability of SeCoML for the use in SME environments.

Further research will analyze the use of additional models to support security engineering and document security engineering best practices using our lightweight approach. Also the integration and transformation of existing models to facilitate the security concept creation, modification, and validation will be examined.

6 References

[1] Alberts, C., Dorofee, A., Stevens, J., Woody, C.: OCTAVE®-S implementation guide, version 1.0 (2005), http://www.sei.cmu.edu/reports/04hb003.pdf

[2] Anderson, R.: Security Engineering: A Guide to Building Dependable Distributed Systems. Wiley & Sons (2001)

[3] Bartels, C., Kelter, H., Oberweis, R., Rosenberg, B.: Technical guidelines for the secure use of RFID – application area trade logistics. Tech. Rep. TR 03126-4, Bundesamt für Sicherheit in der Informationstechnik (2009)

[4] Bartsch, S., Sohr, K., Bormann, C.: Supporting agile development of authorization rules for SME applications. Collaborative Computing: Networking, Applications and Worksharing pp. 461–471 (2009)

[5] Basin, D., Clavel, M., Doser, J., Egea, M.: Automated analysis of security-design models. Information and Software Technology 51(5), 815–831 (2009)

[6] Basin, D., Doser, J., Lodderstedt, T.: Model driven security: From UML models to access control infrastructures. ACM Transactions on Software Engineering and Methodology 15(1), 39–91 (2006)

[7] den Braber, F., Hogganvik, I., Lund, M., Stølen, K., Vraalsen, F.: Model-based security analysis in seven steps – a guided tour to the CORAS method. BT Technology Journal 25(1), 101–117 (2007)

[8] Bundesamt für Sicherheit in der Informationstechnik: BSI-Standard 100-2: IT-Grundschutz methodology (2008), https://www.bsi.bund.de/cae/servlet/contentblob/-471430/publicationFile/27993/standard_100-2_e_pdf.pdf

[9] Bundeskanzleramt Österreich: Österreichisches Informationssicherheitshandbuch (2007), http://www.a-sit.at/pdfs/OE-SIHA_I_II_V2-3_2007-05-23.pdf

[10] Chivers, H., Paige, R., Ge, X.: Agile security using an incremental security architecture. Extreme Programming and Agile Processes in Software Engineering pp. 57–65 (2005)

[11] Club de la Sécurité Informatique Français (CLUSIF): Méthodologie d'Analyse des Risques Informatiques et d'Optimisation par Niveau (MEHARI) (2010)

[12] Direction Centrale de la Sécurité des Systèmes d'Information, Premier Ministre: Expression des Besoins et Identification des Objectifs de Sécurité (EBIOS) - Méthode de Gestion des Risques (2010), http://www.ssi.gouv.fr/IMG/pdf/EBIOS-1-GuideMethodologique-2010-01-25.pdf

[13] Ekelhart, A., Fenz, S., Neubauer, T.: AURUM: A framework for supporting information security risk management. In: Proceedings of the 42nd Hawaii International Conference on System Sciences (2009)

[14] European Network and Information Security Agency: Risk assessment and risk management methods: Information packages for small and medium sized enterprises (SMEs) (2006), http://www.enisa.europa.eu/act/rm/files/deliverables/information-packages-for-small-and-medium-sized-enterprises-smes/at_download/fullReport
[15] Evans, R., Tsohou, A., Tryfonas, T., Morgan, T.: Engineering secure systems with ISO 26702 and 27001. In: 5th International Conference on System of Systems Engineering (2010)
[16] Gesellschaft für Telematikanwendungen der Gesundheitskarte mbH: Übergreifendes Sicherheitskonzept der Telematikinfrastruktur (2008), http://www.gematik.de/upload/-gematik_DS_Sicherheitskonzept_V2.4.0_4493.zip
[17] Houmb, S., Islam, S., Knauss, E., Jürjens, J., Schneider, K.: Eliciting security requirements and tracing them to design: an integration of Common Criteria, heuristics, and UMLsec. Requirements Engineering 15(1), 63–93 (2009)
[18] ISO/IEC: ISO/IEC 13335-1: Information technology – security techniques – management of information and communications technology security – part 1: Concepts and models for information and communications technology security management (2004)
[19] ISO/IEC: ISO/IEC 27001: Information technology – security techniques – information security management systems – requirements (2005)
[20] Jürjens, J.: Secure Systems Development with UML. Springer-Verlag (2005)
[21] Kleppe, A.: Software Language Engineering: Creating Domain-Specific Languages Using Metamodels. Addison-Wesley Professional (2008)
[22] Laforet, S., Tann, J.: Innovative characteristics of small manufacturing firms. Journal of Small Business and Enterprise Development 13(3), 363–380 (2006)
[23] Mayer, N., Heymans, P., Matulevicius, R.: Design of a modelling language for information system security risk management. In: Proceedings of the 1st International Conference on Research Challenges in Information Science. pp. 121–131 (2007)
[24] Ministerie van Binnenlandse Zaken en Koninkrijksrelaties: Afhankelijkheids- en kwetsbaarheidsanalyse (1996)
[25] Normand, V., Félix, E.: Toward model-based security engineering: developing a security analysis DSML. In: Proceedings of the First International Workshop on Security in Model Driven Architecture (SEC-MDA) (2009)
[26] Object Management Group: Object constraint language (OCL) specification (2006), http://www.omg.org/spec/OCL/2.0/
[27] Rodríguez, A., Fernández-Medina, E., Piattini, M.: Towards CIM to PIM transformations: From secure business processes defined in BPMN to use-cases. In: BPM. pp. 408–415 (2007)
[28] Standards Australia/Standards New Zealand: AS/NZS 4360:2004: Risk management (2004)
[29] Talhi, C., Mouheb, D., Lima, V., Debbabi, M., Wang, L., Pourzandi, M.: Usability of security specification approaches for UML design: A survey. Journal of Object Technology 8(6), 103–122 (2009)
[30] Wolter, C., Menzel, M., Schaad, A., Miseldine, P., Meinel, C.: Model-driven business process security requirement specification. Journal of Systems Architecture 55(4), 211–223 (2009)
[31] Zambon, E., Etalle, S., Wieringa, R., Hartel, P.: Model-based qualitative risk assessment for availability of IT infrastructures. Software and Systems Modeling pp. 1–28 (2010)