Association for Information Systems
AIS Electronic Library (AISeL)

AMCIS 2011 Proceedings - All Submissions

8-6-2011

Model Driven Information Security Management - Evaluating and Applying the Meta Model of ISO 27001

Danijel Milicevic
Frankfurt am Main, d.milicevic@fs.de

Matthias Goeken
Frankfurt am Main, Matthias.Goeken@bundesbank.de

Follow this and additional works at: http://aisel.aisnet.org/amcis2011_submissions

Recommended Citation
Milicevic, Danijel and Goeken, Matthias, "Model Driven Information Security Management - Evaluating and Applying the Meta Model of ISO 27001" (2011). *AMCIS 2011 Proceedings - All Submissions*. 376.
http://aisel.aisnet.org/amcis2011_submissions/376

This material is brought to you by AIS Electronic Library (AISeL). It has been accepted for inclusion in AMCIS 2011 Proceedings - All Submissions by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.

Model Driven Information Security Management – Evaluating and Applying the Meta Model of ISO 27001

Danijel Milicevic
IT-Governance-Practice-Network
Frankfurt School of Finance & Management
Frankfurt am Main, Germany
d.milicevic@fs.de

Matthias Goeken
IT-Governance-Practice-Network
Frankfurt School of Finance & Management
Frankfurt am Main, Germany
m.goeken@fs.de

ABSTRACT

Information technology has had a significant impact on business operations and allowed the emergence of new business models. These IT-enabled processes and businesses however depend on secure information systems which need to be managed. The management of information systems security (ISS) is a highly dynamic and complex task due to constant change in the information technology domain. In this paper we propose the use of a meta model to aid ISS managers in setting up a holistic information security management system (ISMS). For this we describe how an adapted meta model of ISO 27001, a security standard for ISMS, can be used to aid with general phases of ISS management. We demonstrate how models can support ISS managers in their endeavors. The paper concludes with a pragmatic evaluation by providing an example of how such a meta model can be operationalized for vulnerability identification, before discussing potential future research.

Keywords

Model Driven Engineering, Information Security Management, Modeling, Meta model.

INTRODUCTION

From the common workplace computer to systems for order fulfillment to entire business models like online shops; our increased dependence on both information as a resource and information systems is beyond question. The absence of either causes significant disruptions in our lives. Whether it is in our personal environment or on an enterprise scale, the confidentiality of our information, the availability of the systems and their integrity are some of the crucial security goals that must be uphold in order for us to benefit from the use of information systems.

In an enterprise setting the preservation of these security goals is the task of the information systems security (ISS) manager. When looking at the areas of work and responsibilities of ISS security managers, we see a constantly growing set of tasks and factors an ISS manager has to take into account to ensure the security of the entrusted systems. As [2] point out, software vulnerabilities and virus attacks are only two typical threats managers need to address along with disgruntled employees, social engineering attacks and industrial espionage to name a few. This evolving repertoire of threats is being complemented by multiple perspectives and requirements on ISS. Laws and regulations need to be upheld and business requirements met, all while keeping operational aspects of ISS in sight.

This multi-dimensionality of ISS [21] is a major cause for the complexity in the decision-making process that managers are facing, along with the complexity that is inherent in any socio-technical system. As information systems evolve over time along with their security requirements and threat landscape, IS managers increasingly need assistance with decision-making. In these decisions the manager needs to consider factors like the value at risk, changing business requirements, new threats and compliance issues to name a few that are relevant for this process. Handling this complexity in ISS practice has been addressed so far mostly with checklist-like approaches and guidelines as found e.g. in ISS standards. However these approaches do not easily lend themselves to a continuous and comprehensive ISS management, as their scope and level of detail does not change over time as the information security management systems (ISMS), which were initially built upon these guidelines and standards, do.

A possible remedy for the complexity and solution for the evolution and consistency of such ISMS is the use of models. So far semi-formal models do not play an important role in the ISS management domain and have only scarcely been used to support ISS management. This paper advocates a model driven approach to ISS management. We assume that such an

approach has various advantages. We consider ISS standards like ISO 27001 to be models as they abstract – just as models do – from specific solutions to broadly applicable generic solutions. A meta model representing the underlying, often implicit structures of models can be used to harmonize the aforementioned perspectives and views on the ISS domain. Furthermore, meta models are a means for better and ad hoc integration of new and important information on the subject matter; and by organizing this new information according to a sound structure, a model driven approach might outperform the traditional planning models, checklists and guidelines. In addition, the reliance on the structures defined by meta models allows for company specific extensions/adaption of existing models and might help in upholding consistency during such changes.

The paper is structured as follows: in section 2 we discuss related work in the ISS domain, specifically the use of models, meta models and related methods in ISS management. In the third section we describe our proposed research methodology. In section 4 we present a previously constructed meta model of the ISO 27001 standard. In section 5 we conduct a pragmatic evaluation by showing how sub-models of the meta model can support core ISS management processes before we conclude in section 6 and describe future research in this field.

RELATED WORK

In regard to our proposed ISS management methodology several areas of research are important and will be discussed as related work. At first an overview of existing models and modeling approaches in ISS will be presented in order to illuminate differences between them and short-comings when it comes to broad ISS management support. Secondly Model Driven Engineering will be discussed as methodology which we loosely base our approach on.

Modeling in IS

Modeling and models exist in a large variety in the ISS domain; however their scope, use, level of detail and level of formalism varies greatly. Several modeling languages and notations have been derived from the semi-formal UML. In order to give the reader insight into their scope we will briefly describe three representative notations based on UML. One such UML derivate is SecureUML [13], which extends UML via a meta model that incorporates the role-based access control (RBAC [18]) approach in order to model authentication and authorization mechanisms during the development of software. As it focuses on a specific type of control (access controls) in the context of software development, its scope is rather limited and the meta model does not incorporate any concepts that'd allow e.g. the modeling of threats or security requirements apart from authentication/authorization. Another UML derivate is UMLsec [11] which in a two-step approach firstly transforms existing UML specifications into UMLsec specification and in a second step allows for a security analysis. The UMLsec specification mainly incorporates dependencies and behavior, a concept which mainly describes communication between components and systems. This is also the scope of UMLsec, which focuses on threats in the communication of information, not e.g. the processing. A third semi-formal UML derivate we found was Misuse Cases [20]. This extension of UML use case diagrams allows for the modeling of actions an adversary/attacker might perform. The notation adds actions that threaten the use cases within the system boundaries as well as additional security-related use cases which mitigate those attacks.

Along with these – mostly academic – semi-formal modeling approaches many informal models exist in ISS practice. IT practitioners often consolidate their experience in so-called best practice frameworks like ITIL or COBIT [7]. In ISS specifically models like the Generally Accepted Information Security Principles (GAISP [6]) or standards like ISO 27001 [10] are common and widely used. These informal models usually are comprised of continuous text, bullet points and checklists and in the case of the two mentioned examples have a broad scope in ISS. The informal approach makes the use of such documents easy, however, it does not lend itself to advanced techniques such as the integration of multiple models or the transfer of these models into software tools, e.g. for automation, model-consistency checks or logic reasoning.

The opposite of such informal models are formal models like ontologies. Identifying core concepts of a domain and their relations with each other has been the motivation and goal of many modeling approaches. Ontologies specify concepts and relations between them for a given domain or context [8]. Although the ISS research community has long identified the need for a comprehensive and common set of core concepts for the ISS domain (e.g. [4]), [3] conclude from their comparative analysis of 30 ISS ontologies that no such thing has been found (or agreed upon) so far. Nonetheless numerous ISS ontologies exist. ISS researchers have identified ontologies as a means to structure either the entire information systems security problem domain or specific subdomains and made contributions (e.g. [1, 10, 24]). However the high degree of formalization leads to an equally high degree of difficulty in creating and maintaining ontologies along with e.g. their axioms. A level of difficulty and inevitable time-investment that goes beyond what ISS practitioners can often afford.

As such, currently deployed models and practiced modeling in the ISS domain each have their strengths and can potentially contribute to a model driven ISS management approach. In order to benefit from the collective knowledge of ISS practitioners we propose the use of the informal best-practice models as a knowledge base, while increasing formalization in

order to allow the use of advanced techniques like model integration. As a targeted formalization level we've chosen a semi-formalism as it allows a methodological approach, while still being accessible to ISS practitioners (as the heavy use of UML-based languages and diagrams in IT practice shows).

Model Driven Engineering

The main idea behind Model Driven Engineering (MDE) [19] is to lift the specification during application development (or more broadly: information systems, as it is also used for enterprise architectures amongst other things) on a higher abstraction level in order to gain access to automation opportunities. During the phases of specification and development the constructed model of the to-be-developed information system is being transformed into increasingly lower-level models using automated model transformations and model interpretation mechanisms. The most prominent example of MDE would be Model Driven Architecture (MDA).

In MDE terms we propose the construction of a domain-specific-model (DSM), which will incorporate concepts and relations of the ISS management domain. However we will not (at this stage) create our own domain-specific-language (DSL) to do so, but rely on the semi-formal UML as a language (just as MDA for instance does) for the aforementioned reasons. As such we only loosely follow the ideas behind MDE at this point. It would not be beneficial for an ISS practitioner to work with a model which is supposed to improve the handling of complexity, when the same model requires the practitioner to learn a new modeling language in order to use it.

RESEARCH METHODOLOGY

As the nucleus for our approach – an initial high-level model – we use the ISO 27001 meta model [16]. As mentioned before, by using a best-practice model as our foundation we build upon the consolidated knowledge base of ISS practitioners. The meta model has been developed using a rigorous approach based on grounded theory and qualitative data analysis (QDA) methods.

The basic idea in grounded theory (as with most QDA methods) is to work with empirical data like transcripts from interviews, protocols and documents a researcher is confronted with in the field. The focus is on inductively developing a theory, which is 'grounded' in the respective empirical data. One central activity is the "coding", which means conceptualizing qualitative data and assigning categories as well as relations between them. The events and instances a researcher is facing in the data are analyzed as potential "indicators of phenomena … which are thereby given conceptual labels" [8, p. 7]. With this approach the structure of ISO 27001 was derived by identifying relevant categories/concepts as well as their relations. Figure 1 depicts the proposed research methodology starting with the ISO 27001 meta model.

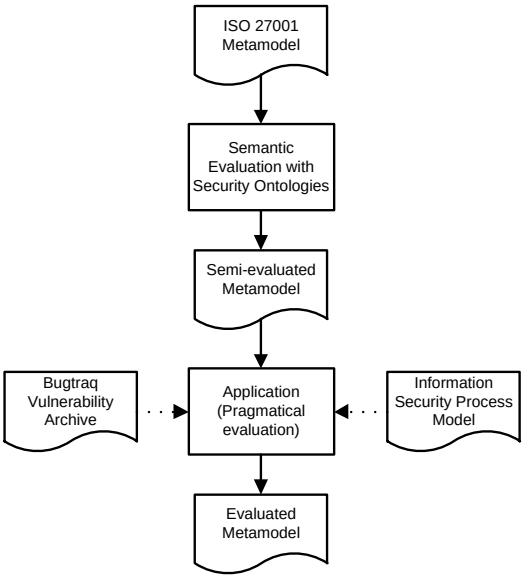

Figure 1. Research Methodology

Like in information systems development (ISD) we assume, that the quality of an end-product (in our case the ISS management system) depends to a great extent on the quality of early models, which conceptualize the problem domain in

earlier stages of the development process. Therefore, prior to the model application in the context of ISS managerial processes, we evaluate the meta model by applying criteria from literature. We rely on the evaluation approach of [12] which – according to [14] – "is the only one that has both a theoretical basis and has been empirically validated".

[12] present a framework for evaluating conceptual models which is based on linguistic concepts, and, furthermore, distinguishes between goals and means. The syntactical quality describes whether the model adheres to the rules of the grammar. In this research we used UML class diagrams, which were checked for syntactic correctness. Usually this quality dimension can be controlled [14].

The semantic quality evaluates how well the model reflects the reality or subject matter. Important issues in this respect are, if the model lacks something that the domain contains or if it includes something the domain doesn't have. As we do not model the real world in our meta model, but the structures of another model, the criteria proposed by [12] and [14] – e.g. validity, completeness, feasible validity – cannot be applied in our work. We therefore operationalize the semantic quality criteria using coherence and consistency with prior literature. In a first step of our research process we compare the first version of our meta model with related models and ontologies that share the same domain and scope. This semantic evaluation has been conducted in prior work [17]. Figure 2 shows the relevant quality criteria for conceptual models, which should be operationalized for an evaluation.

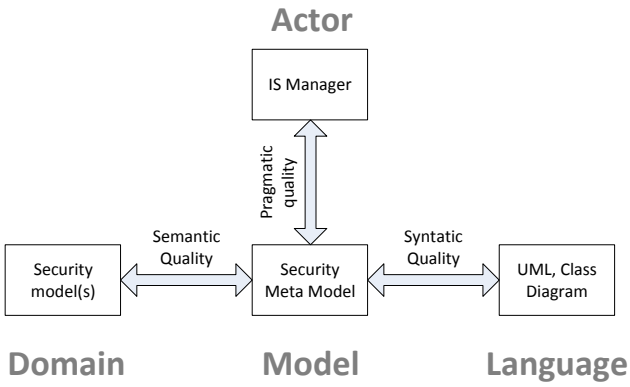

Figure 2. Quality criteria for the evaluation of conceptual models (see [12; 14])

So far the pragmatic quality of models has only scarcely been addressed in the research of conceptual modeling in general. To the best of our knowledge the pragmatic evaluation of ISS model quality has not been operationalized in literature. While [12] reduce the notion of pragmatics to the comprehension we assume that a broader conception would be fruitful, esp. in regards to adoption rate by ISS practitioners. In accordance to [15] we address aspects of 'pragmatics of action'. Hence, in section 6 we demonstrate the applicability by using the meta model within a management process of ISS. Drawing on an example we can reveal how the model can solve relevant problems.

The goal is not to evaluate whether or not the meta model is a good representation of the ISO 27001 standard, as we believe that the repetitive inductive categorization process used to derive the concepts of the meta model is rigorous and complete. Rather our goal is to determine how well the meta model (and as such the concrete ISO 27001 model) reflects the reality or subject matter.

ISO 27001 META MODEL

Based on an inductive categorization approach a meta model of the ISO 27001 ISS standard (Annex A specifically) has been developed. The meta model is shown in figure 3 and will be briefly discussed. See [16] for a full discussion. Additionally the semantic evaluation has been completed in [17] and been incorporated in the adapted ISS management meta model.

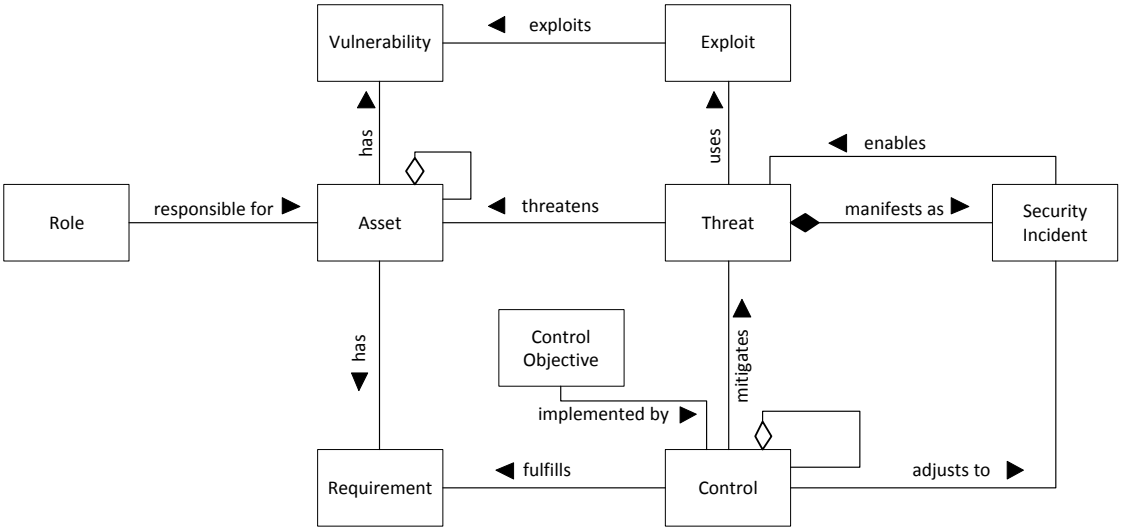

Figure 3. Adapted Information Systems Security Management Meta Model [17]

The concept 'requirements' is further specified by three sub concepts we identified in the standard: 1) security requirements, 2) legal requirements and 3) business requirements. These distinctions indicate potential aspects or layers of information security management, as suggested by other information security researchers (see [9]).

In comparison to the other core concepts 'role' has relatively weak grounding based on the in-vivo coding. However, with 10 quotations the code 'responsibility' is one of the more predominant ones and represents the relation between 'role' and other concepts, mainly 'asset'. To include this emphasis on an ownership-type paradigm we decided to include role as a (supporting) core concept. The relationships among the concepts have been derived by analyzing quotations.

After adding the five core concepts (asset, threat, control, requirement and role) and branching 'requirement' and 'control' into sub concepts, we re-evaluated prior excluded codes with singular occurrences. By doing so we identified three codes that had a semantic similarity: 'security event', 'security incident' and 'security breach'. While interpretation allows to distinct them by varying levels of severity, we decided to perform code merging and add them as one core concept ('security breach'), since an analysis of quotations for the 'control' and 'threat' concept showed that this element played an important part for the control objectives A.8.2, A.10.10 and A.13.2. Additionally we added 'control objective', as it is an important structural element in the standard which groups controls and elaborates on their common purpose.

It is notable, that in the part of the ISO 27001 standard we used for our analysis, measures were not included systematically, even though in the rest of the standard they are mentioned frequently. Furthermore, there is no strong evidence that roles and responsibilities might be assigned to controls or control objectives. From a governance point of view, it would be of central importance to define accountability and decision rights during the implementation of a security standard, not just for assets, but potentially controls and even threats.

During the semantic evaluation in [17] with other ontologies of the ISS domain twofurther concepts have been identified: vulnerabilities, most common as software flaws, and exploits, the specific attack vector of a threat, mostly tailored to benefit from a specific vulnerability. These two concepts are crucial for the day-to-day activities in ISS operations.

PRAGMATIC EVALUATION – APPLICATION OF THE META MODEL

For a pragmatic evaluation of the meta model, activities (as 'actions' in [15]'s proposed pragmatic evaluation) in information security management need to be identified. Most activities are linked to the three main questions information security managers are facing: 1) What needs to be protected? 2) Against what does it have to be protected? 3) How can it be protected? Answering these three fundamental questions leads to: 1) the identification of assets, 2) the identification of threats and 3) the identification and selection of appropriate countermeasures as main activities for ISS managers. However this way of ad-hoc reasoning does neither allows nor conforms with a rigorous evaluation, as such – similar to our semantic

evaluation – a comparison and consolidation of multiple ISS management processes would be preferable. Thankfully we can build upon work of other researchers in this respect.

A Process Model of Information Security

We base the pragmatic evaluation on activities that are found in the related ISO 27005 standard, which describes a so-called information security risk management (ISRM) process. [5] have analyzed said standard along with 4 other standards and derived generic phases of information security (risk) management, which are listed in table 2.

Generic phase	ISO 27005 phase	Output
System Characterization	Identification of Assets	Inventory list of assets to be protected, including their accepted risk level
Threat and Vulnerability Assessment	Identification of Threats, Identification of Vulnerabilities	List of threats and corresponding vulnerabilities endangering the identified assets
Risk Determination	Identification of Impact, Assessment of Threat Likelihood, Assessment of Vulnerability Likelihood, Risk Estimation	Quantitative or qualitative risk figures/levels for identified threats (input: threat probability and magnitude of impact)
Control Identification	Evaluation of Existing and Planned Controls	List of potential controls that can mitigate the risks to an acceptable level
Control Evaluation and Implementation	Information Security Risk Treatment (Risk Avoidance, Risk Reduction, or Risk Transfer)	List of cost-efficient controls that have to be implemented to reduce the risk to an acceptable level

Table 1. ISRM process model (extract from [5])

In the left column the generic phases coined by [5] are listed with their equivalent from the ISO 27005 standard in the middle column. [5] also listed expected outputs for each of the phases, which can be found in the right column. Based on this process model and outcomes we'll discuss possible applications of the meta model and demonstrate how it can support information security managers in their tasks. Due to page restrictions we cannot include concrete models for each phase. These can be obtained from the authors via request.

Mapping of Opportunities for Model-Support

System Characterization

In the system characterization phase the information security manager is tasked to identify and list all assets which require protection. What constitutes an asset is not necessarily clear. We argue that relevant assets need to be deserving of protection, meaning they must have a value attached to them which may be inflicted if an attack was to be successfully launched. [5] also add the inclusion of accepted risks levels in the asset identification process.

Models, and in this instance our meta model, can support this phase by instantiating the asset concept and providing the necessary attributes to detail things like the internal identification code (e.g. from the inventory system or a configuration item database like used in ITIL), vendor name or version. By adding a recursive reference to the asset concept in our meta model certain dependencies between assets could be modeled as well, e.g. the asset "web server" could be running on an asset "blade server". Were the web server to be compromised, other assets running on the same host machine would be exposed to threats that wouldn't be identified outside of this larger context.

Threat and Vulnerability Assessment

During the Threat and Vulnerability Assessment phase the identified assets are checked for known threats which could be realized via vulnerabilities within the asset. The terminology stems mainly from software security and refers to software vulnerabilities which can be exploited by so-called Exploits. However this setting can equally be transferred to non-software assets like confidential information known to management. The threat might be social engineering, where an attacker tries to persuade the victim to reveal information and grant him access without realizing the situation. The vulnerability in this scenario may be a lack of security awareness on the part of the manager who is targeted.

The assessment of threats and vulnerabilities has a certain temporal element to it, as the threat concept to our understanding represents a class of potential vulnerabilities. As such threats are often found in (hierarchical) taxonomies and special threat

ontologies, while new vulnerabilities quite literally appear every day and as such need to be assessed in a more continuous way.

The vulnerability concept is quite common in information security modeling, as our comparison with multiple ontologies showed. However the ISO 27001 standard did not explicitly mention it. In order to support this phase the vulnerability concept should be added to the meta model so it can be instantiated, just like the threat concept can be.

Risk Determination

The risk determination deals with qualitative or quantitative factors of risk, especially probabilities and the impact of a successful security breach. Typically various approaches suggest the formulation of varying degrees of risk in order to build a simple qualitative risk model based on e.g. three levels of probability and severity (like: "low", "medium", "high").

This is the phase where the described meta model has the least to offer in terms of support. Ultimately the determined risk levels can be added as attributed to various concepts, but those number cannot be analyzed or condensed without additional methodology that our model is not support as of right now. However it is desirable to find a solution to this as the quantification of information security is one of the grand challenges in this domain. The model of assets, threats and vulnerabilities could allow the description of propagation or sharing of risk amongst multiple assets or assets in a certain constellation.

Control Identification

Similar to the identification of assets and threats and vulnerabilities the possible controls must be identified, stored and analyzed in order to support information security managers in their decision making process.

Models in general and our meta model in this instance can help by instantiating the control concept as well as putting various controls in relation to each other using the recursive relation. This helps orchestrating a set of controls, e.g. to model a so-called "defense in-depth" approach, where multiple layers of protection are setup so potential attackers need to overcome various obstacles in their attempt to breach security.

Control Evaluation and Implementation

Like the risk determination phase the control evaluation and implementation activities revolve around quantifiable measures in order to select the most effective, most efficient control and reduce risk to an acceptable level.

Our meta model in its current state cannot sufficiently support this phase as additional data is necessary, especially regarding effectiveness of controls. We see the potential of future research in this field in order to determine how to add measurements to the model in order to guide information security managers to better decisions.

EXEMPLARY APPLICATION AND DISCUSSION

To further elaborate on the applicability of the meta model we'll describe a scenario where real world data can be used to instantiate the model and where the model offers substantial support to managerial tasks in information security. Based on the discussion in the prior section we're using the adapted meta model to match our requirements for this scenario of threat and vulnerability identification. In this scenario an enterprise IT infrastructure is comprised of merely a webserver and the IS managers' task is to identify potential threats to this asset.

For this scenario the asset, threat, vulnerability and exploit are required. While assets may have vulnerabilities, only once there is a way to exploit them those vulnerabilities become significant (of course vulnerabilities should be addressed evening without immediate threats). To test how said model could be used in a real life scenario we collected data from the information security mailing list "Bugtraq"[1] and modeled a web server as the asset of interest. By selecting a specific version, in our case version 2.2.12, we got access to a list of known (and reported) vulnerabilities assigned to this version of this particular web server software. Additionally we added the listed exploit along with a general class of threats to the modeled scenario. All information is available from a generated Bugtraq report. Figure 4 shows the instantiated concepts as UML classes.

[1] http://www.securityfocus.com/archive/1

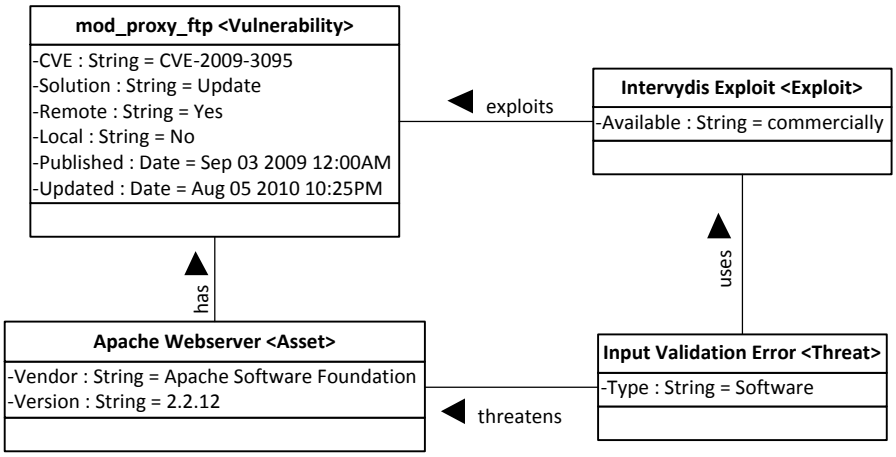

Figure 4. Instantiation of Core Concepts

As can be seen, by merely modeling his assets, the ISS manager can trace back (in this case via vulnerability reports) what types of threats exist and are an immediate danger and matter to be dealt with. This replaces theoretical threat models with actual threat models for a given asset; a webserver in our case. If this scenario was to be extended and additional complexity added (by adding dependencies to other assets, say a web-based content management system), further analyses can be performed, which would be both complex and tedious without model-support. For example side-effects of a software update (to mitigate the vulnerability and threat posed by the current version) may cause a conflict with business requirements, which rely on a web application that is only functional with the specific software version. Another possibility could be that an update would violate security requirements, if an ISS policy is in place that states which type of software releases are approved for deployment (many software products have releases for production environment and for testing environment, where full functionality is not guaranteed).

Upon further inspection of various archives similar to the Bugtraq mailing list, it is our belief that this process can be easily automated once an ISS manager has identified and modeled the assets in his enterprise. Transforming the meta model and this approach into a functional tool is part of future research endeavors as page restriction limit the amount of meta model transformations and model instantiations that can be shown in this paper.

CONCLUSION AND FUTURE RESEARCH

In this paper we presented a previously constructed meta model of the ISO 27001 standard. We presented an overview of existing modeling approaches in the ISS domain and proposed a model driven ISS management approach to remedy certain conflicts and problems. A semantically evaluated meta model has been the starting point (the adapted meta model) for our research.

In their comparison of security ontologies [3] focused on ontological metrics and evaluated essentially the structure of the proposed ontologies, not their content. Based on our comparison in [17] we could adapt the meta model for a pragmatic evaluation and add the necessary concepts to make the meta model applicable for a fictional scenario in the field of vulnerability identification, a crucial process for ISS managers. By instantiating the respective concepts of the meta model according to the ISRM process model by [5], an ISS manager can construct a consistent and extensive model representation of the key security concepts. Additionally the meta model approach allows for structured integration of multiple models in a multi-model-environment.

In an extension of the presented research the relationships between the concepts will be enriched with additional data, e.g. empirical data regarding the effectiveness of a given control regarding a set of threats. Additionally it would be desirable to empirically evaluate the application of the proposed model driven ISS management approach. Also a full set of derived concrete models for each ISS management activity will be published.

REFERENCES

1. Amaral, F. d. N., Bazilio, C., Silva, G. M. H. d., Rademaker, A., and Haeusler, E. H. 2006. An Ontology-based Approach to the Formalization of Information Security Policies. In Proceedings of the 10th IEEE on International Enterprise Distributed Object Computing Conference Workshops EDOCW '06, IEEE Computer Society.
2. Arief, B. and Besnard, D. 2003. Technical and human issues in computer-based systems security. Report No. CS-TR 790, University of Newcastle, UK.
3. Blanco, C., Lasheras, J., Valencia-Garcia, R., Fernandez-Medina, E., Toval, A., and Piattini, M. 2008. A Systematic Review and Comparison of Security Ontologies. In Proceedings of the Third International Conference on Availability, Reliability and Security, 813-819.
4. Donner, M. 2003. Toward a Security Ontology, IEEE Sec. & Privacy, 1, 3, 6-7.
5. Fenz, S. and Ekelhart, A. 2010. Verification, Validation, and Evaluation in Information Security Risk Management. IEEE Security & Privacy PrePrint. DOI=http://doi.ieeecomputersociety.org/10.1109/MSP.2010.117
6. Generally Accepted Information Security Principles (GAISP) Version 3.0, 2003; www.issa.org/gaisp/_pdfs/v30.pdf.
7. Goeken, M. and Alter, S. 2009. Towards Conceptual Metamodelling of IT Governance Frameworks. Approach - Use – Benefits. In Proceedings of the 42nd Annual Hawaii Int. Conference on System Sciences, Hawaii.
8. Grubner, T. R. 1995. Towards principles for the design of ontologies used for knowledge sharing. Int. Journal of Human-Computer Studies, 43, 5, 907-928.
9. Gurpreet, D. and Backhouse, J. 2001. Current directions in IS security research: towards socio-organizational perspectives. Information Systems Journal, Vol. 11, No. 2, 127-154.
10. International Organization for Standardization and International Electrotechnical Commission 2005. ISO/IEC 27001:2005, information technology - security techniques - information security management systems- requirements.
11. Jürjens J. 2002. UMLsec: Extending UML for Secure Systems Development. In: Lecture Notes In Computer Science Vol. 2460, Proceedings of the 5th International Conference on The Unified Modeling Language, Springer, S.412 – 425.
12. Lindland, O.I., Sindre, G., and Sølvberg, A. 1994. Understanding quality in conceptual modeling. In: IEEE Software 11(2), 42–49.
13. Lodderstedt T., Basin, D. and Doser, J. 2002. SecureUML: A UML-Based Modeling Language for Model-Driven Security, Lecture Notes In Computer Science Vol. 2460, Proceedings of the 5th International Conference on The Unified Modeling Language, Springer Verlag (Hrsg.), S.426-441
14. Maes, A. and Poels, G. 2007. Evaluating quality of conceptual modelling scripts based on user perceptions. Data Knowl. Eng. 63(3): 701-724.
15. Mendling, J., Recker, J. 2007. Extending the Discussion of Model Quality: Why Clarity and Completeness may not always be enough. In: B. Pernici and J. A. Gulla: CAiSE 2007 Workshop Proceedings Vol. 1. Tapir Academic Press, Trondheim, Norway, pp. 109-121.
16. Milicevic, D. and Goeken, M. 2010. Konzepte der Informationssicherheit in Informationssicherheitsstandards am Beispiel ISO 27001. In: Lecture Notes in Informatics (LNI) Band P-175, Springer, Leipzig.
17. Milicevic, D. and Goeen, M. 2010. Ontology-based evaluation of ISO 27001. In: Proceedings of the 10th IFIP Conference on e-Business, e-Services and e-Society, Buenos Aires.
18. Sandhu, R. S., Coyne, E. J., Feinstein, H. L., Youman, C. E. 1996. Role-based access control models. Computer 29(2), 38-47.
19. Schmidt, D. C. 2006. Guest Editor's Introduction: Model-Driven Engineering. Computer 39(2), 25-31.
20. Sindre, G. and Opdahl, A. L. 2005. Eliciting security requirements with misuse cases, Requirements Engineering, 10, 1, 34-44.
21. Solms, S.H. and Solms, R. 2009. Information Security Governance, New York.